Max-Min Greedy Matching

Alon Eden
Tel Aviv University, Israel
alonarden@gmail.com

Uriel Feige
Weizmann Institute of Science, Rehovot, Israel
uriel.feige@weizmann.ac.il

Michal Feldman
Tel Aviv University, Israel
Microsoft Research, Herzlyia, Israel
michal.feldman@cs.tau.ac.il

Abstract

A bipartite graph $G(U, V; E)$ that admits a perfect matching is given. One player imposes a permutation π over V, the other player imposes a permutation σ over U. In the greedy matching algorithm, vertices of U arrive in order σ and each vertex is matched to the highest (under π) yet unmatched neighbor in V (or left unmatched, if all its neighbors are already matched). The obtained matching is maximal, thus matches at least a half of the vertices. The max-min greedy matching problem asks: suppose the first (max) player reveals π, and the second (min) player responds with the worst possible σ for π, does there exist a permutation π ensuring to match strictly more than a half of the vertices? Can such a permutation be computed in polynomial time?

The main result of this paper is an affirmative answer for these questions: we show that there exists a polytime algorithm to compute π for which for every σ at least $\rho > 0.51$ fraction of the vertices of V are matched. We provide additional lower and upper bounds for special families of graphs, including regular and Hamiltonian graphs. Our solution solves an open problem regarding the welfare guarantees attainable by pricing in sequential markets with binary unit-demand valuations.

2012 ACM Subject Classification Theory of computation → Computational pricing and auctions; Mathematics of computing → Matchings and factors

Keywords and phrases Online matching, Pricing mechanism, Markets

Digital Object Identifier 10.4230/LIPIcs.APPROX-RANDOM.2019.7

Category APPROX

Related Version `https://arxiv.org/pdf/1803.05501.pdf`

Funding *Alon Eden*: Partially supported by the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) / ERC grant agreement number 337122, the Israel Science Foundation (grant number 317/17), and the Blavatnik family foundation.
Uriel Feige: Partially supported by the Israel Science Foundation (grant No. 1388/16).
Michal Feldman: Partially supported by the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) / ERC grant agreement number 337122, the Israel Science Foundation (grant number 317/17), and the Blavatnik family foundation.

Acknowledgements A substantial part of this work was conducted in Microsoft Research, Herzlyia. We are grateful to Amos Fiat and Sella Nevo for numerous discussions that contributed significantly to the ideas presented in this paper. We also thank Robert Kleinberg for helpful discussions.

Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2019).
Editors: Dimitris Achlioptas and László A. Végh; Article No. 7; pp. 7:1–7:23
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Given a bipartite graph $G(U, V; E)$, where U and V are the sets of vertices and $E \in U \times V$ is the set of edges, a matching $M \subset E$ is a set of edges such that every vertex is incident with at most one edge of M. For simplicity of notation, for every n we shall only consider the following class of bipartite graphs, that we shall refer to as G_n. For every $G(U, V; E) \in G_n$ it holds that $|U| = |V| = n$ and that E contains a matching of size n (and hence G has a perfect matching). All results that we will state for G_n hold without change for all bipartite graphs that have a matching of size n (and arbitrarily many vertices).

Karp, Vazirani and Vazirani [12] introduced the *online bipartite matching* problem. This setting can be viewed as a game between two players: a maximizing player who wishes the resulting matching to be as large as possible, and a minimizing player who wishes the matching to be as small as possible. First, the minimizing player chooses $G(U, V; E)$ in private (without the maximizing player seeing E), subject to $G \in G_n$. Thereafter, the structure of G is revealed to the maximizing player in n steps, where at step j (for $1 \leq j \leq n$) the set $N(u_j) \subset V$ of vertices adjacent to u_j is revealed. At every step j, upon seeing $N(u_j)$ (and based on all edges previously seen and all previous matching decisions made), the maximizing player needs to irrevocably either match u_j to a currently unmatched vertex in $N(u_j)$, or leave u_j unmatched.

There is much recent interest in the online bipartite matching problem and variations and generalizations of it, as such models have applications for allocation problems in certain economic settings, in which buyers (vertices of U) arrive online and are interested in purchasing various items (vertices of V). A prominent example of such an application is online advertising; for more details, see for example the survey by Mehta [17]. The new problems are both theoretically elegant and practically relevant.

Max-Min Greedy Matching

We study a setting related to online bipartite matching, that we call *Max-Min Greedy matching*. Our setting is also a game between a maximizing player and a minimizing player. The bipartite graph $G(U, V; E) \in G_n$ is given upfront. Upon seeing G the maximizing player chooses a permutation π over V. Upon seeing G and π, the minimizing player chooses a permutation σ over U. The combination of G, π and σ define a unique matching $M_G[\sigma, \pi]$ that we refer to as the greedy matching. It is the matching produced by the greedy matching algorithm in which vertices of U arrive in order σ and each vertex $u \in U$ is matched to the highest (under π) yet unmatched $v \in N(u)$ (or left unmatched, if all $N(u)$ is already matched).

The matching $M_G[\sigma, \pi]$ has several additional equivalent definitions. For example, $M_G[\sigma, \pi]$ is the matching produced by the greedy matching algorithm in which vertices of V arrive in order π and each vertex $v \in V$ is matched to the highest (earliest in arrival order under σ) yet unmatched $u \in N(v)$ (or left unmatched, if all $N(v)$ is already matched). Also, $M_G[\sigma, \pi]$ is the unique stable matching in G (in the sense of [9]), if the preference order of every vertex $u \in U$ over its neighbors is consistent with π, and the preference order of every vertex $v \in V$ over its neighbors is consistent with σ.

Let $\rho[G] = \frac{1}{n} \max_\pi \min_\sigma [|M_G[\sigma, \pi]|]$, and let $\rho = \min_{G \in G_n}[\rho[G]]$. It is easy to see that $\rho \geq \frac{1}{2}$. In fact, to ensure a matching of size $n/2$, the max player need not work hard. Since every greedy matching is a *maximal* matching, for *every* permutation π the obtained matching is of size at least $n/2$. The question we study in this work is whether the max player can ensure a matching of size strictly greater than $n/2$; that is, whether ρ is strictly greater than $\frac{1}{2}$.

For an upper bound on ρ, it was observed by Cohen Addad et al. [4] that $\rho \leq 2/3$. To show this, they observe that in the 6-cycle graph, depicted in Figure 1, no permutation π can guarantee to match more than two vertices in the worst case. Indeed, suppose (without loss of generality) that $\pi = (v_3, v_2, v_1)$. For $\sigma = (u_1, u_3, u_2)$, u_1 is matched to v_2, u_3 is matched to v_3, and u_2 is left unmatched, resulting in a matching of size 2.

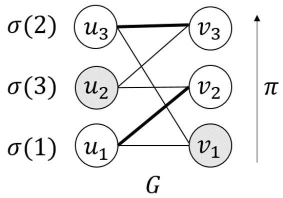

Figure 1 For every permutation π there exists a permutation σ that matches only 2 of the 3 vertices. Thick edges are in the matching; gray vertices are unmatched.

1.1 Our Results

Our main result resolves the open problem in the affirmative:

▶ **Theorem** (main theorem). *It holds that $\rho \geq \frac{1}{2} + \frac{1}{86} > 0.51$. Moreover, there is a polynomial time algorithm that given $G(U, V; E)$ produces a permutation π over V satisfying the above bound.*

The significance of this result is that 1/2 is not the optimal answer. We believe that further improvements are possible. In fact, for Hamiltonian graphs we show that $\rho \geq \frac{5}{9}$ (see Section 6).

The proof method is quite involved; it is natural to ask whether simpler approaches may work. In what follows we specify three natural attempts that all fail.

Failed attempt 1: random permutation

A first attempt would be to check whether a random permutation π obtains the desired result (in expectation). The performance of a random permutation is interesting for an additional reason: it is the performance in scenarios where the graph structure is unknown to the designer. Unfortunately, there exists a bipartite graph G, even one where all vertices have high degree, for which a random permutation matches no more than a fraction $1/2 + o(1)$ of the vertices (see Section 4).

In contrast, we show that in the case of Hamiltonian graphs a random permutation guarantees a competitive ratio strictly greater than 1/2 (Section 4). A similar proof approach applies to regular graphs as well.

▶ **Theorem** (random permutation). *There is some $\rho_0 > \frac{1}{2}$ such that for every Hamiltonian graph $G \in G_n$, regardless of n, a random permutation π results in $\rho \geq \rho_0$. Similarly, there is some constant $\rho_0 > \frac{1}{2}$ such that for every d-regular graph G, regardless of d, n, a random permutation π results in $\rho \geq \rho_0$.*

Failed attempt 2: iterative upgrading

A second attempt would be to iteratively "upgrade" unmatched vertices, with the hope that the iterative process will reach a state where many vertices will be matched. That is, in every iteration consider the worst order σ for the current permutation π and move all unmatched vertices (in the matching induced by (π, σ)) to be ranked higher in π. This algorithm is similar to the *k-pass Category Advice* algorithm of [1], but with the difference that in [1] σ remains unchanged throughout the k iterations. In [1] it was shown that in their setting, the fraction of matched vertices approaches $\frac{2}{1+\sqrt{5}} \simeq 0.619$ as k grows. In contrast, in Appendix B we show that in our setting this process can go on for $\log n$ iterations before reaching a permutation that matches more than a half of the vertices. This fact gives some indication that establishing a proof using this operator might be difficult.

Failed attempt 3: degree-based ranking

A third attempt would be to give preference to vertices with lower degrees, as they would have fewer opportunities to be matched to incoming vertices of U. Consider a graph with multiple copies of the subgraph $(u_1, v_1), (u_2, v_2), (u_1, v_2)$ along with two additional vertices u_a, u_b (and their partners v_a, v_b). If we connect all vertices of type v_1 to u_a and u_b, we get that their degree is 3, while the degree of vertices of type v_2 is 2. If π is chosen according to the degree, vertices of type v_2 will be ranked higher than vertices of type v_1. In this case, if σ orders the vertices of type u_1 first, they will be matched to vertices of type v_2, leaving the vertices of type v_1 unmatched. The resulting matching will therefore be of size $(1/2 + o(1))n$.

Why is this model interesting mathematically?

The setting of max-min greedy matching is easy to state. The problem of getting a ratio better than half turns out to be deceptively difficult. As discussed above, several natural approaches fail to achieve this. The problem remained open for quite some time, despite attempts to solve it. Indeed, the solution that we find is not simple; it involves taking the best of four algorithms. However, these algorithms are not unrelated. They all share a unifying theme that involves a clean combinatorial property, referred to as a maximal path cover (see Section 2). This theme enabled us to break the barrier of half, but interesting problems remain open, such as whether the bound of 2/3 can be achieved. We hope that the progress made in this work will motivate and enable further improvement in this interesting problem.

1.2 Additional Results

We further establish lower and upper bounds for regular graphs.

▶ **Theorem** (regular graphs)**.** *For d-regulars bipartite graphs, $\rho \geq \frac{5}{9} - O(\frac{1}{\sqrt{d}})$. On the other hand, for every integer $d \geq 1$, there is a regular graph G_d of even degree $2d$ such that $\rho(G_d) \leq \frac{8}{9}$.*

An additional natural problem is to find the best permutation π, given a graph G. We suspect that this is a difficult computational problem. However, the special case of determining whether there is a *perfect* π (a permutation on V that for every permutation σ leads to a perfect matching) does have a polynomial time algorithm (proof appears in Section 5).

▶ **Proposition 1.** *There is a polynomial time algorithm that given a graph $G \in G_n$ determines whether G has a* perfect *π, and if so, outputs a perfect π.*

1.3 Application to Resource Allocation and Pricing

Various problems related to online bipartite matching are closely related to problems that attract attention in the algorithms community. Gaining better understanding of the max-min greedy matching problem sheds light on more general problems, some of which are still open. In what follows we elaborate on an application to a pricing problem.

Feldman et al. [8] study the design of pricing mechanisms for allocation of items in markets. The basic setting is a matching market, where v_{ij} is the value of agent i for item j, and every agent can receive at most a single item. The seller assigns prices to items, and agents arrive in an adversarial order (after observing the prices), each purchasing an (arbitrary) item that maximizes their utility (defined as value minus price). It is shown that, given a weighted bipartite graph (with agents on one side, items of the other side, and weight v_{ij} for the edge between agent i and item j), one can set item prices that guarantee at least half of the optimal welfare for any arrival order. The last result holds in much more general settings, namely settings where buyers have submodular valuations over bundles of items [1], and even in a Bayesain settings, where the seller knows only the (product) distribution from which agent valuations are drawn, but not their realizations.

In the Bayesian setting no item prices can guarantee better than half of the optimal welfare in the worst case. A natural question is whether this ratio can be improved in scenarios where the designer knows the realized values of the buyers from the outset [2]. Concretely, do there exist item prices that guarantee strictly more than half the optimal welfare, for any arrival order σ? Not only has this question been open for general combinatorial auctions with submodular valuations, it has been open even for unit-demand buyers, and even if all individual values are in $\{0, 1\}$ (henceforth referred to as binary unit demand valuations). In the latter setting, pricing is equivalent to imposing a permutation over the items, hence the max-min greedy matching is a precise formulation for the pricing problem in binary unit-demand settings.

An equivalent scenario is one where in each step the "items player" offers an item, and the "buyers player", upon seeing the item, allocates the item to one of the buyers who wants the item (if there is any), and that buyer leaves. The items player is non-adaptive (plays blindfolded, without seeing which buyers remain[3]). The size of the matching that can be guaranteed by the items player is equivalent to the max-min greedy matching problem.

Yet an additional equivalent formulation of the problem is one where the permutation π is imposed over the buyers rather than over the items. The buyers then arrive in the order of π, each taking an arbitrary item she wants. One can verify that the size of the matching that can be guaranteed by an ordering over the buyers is equivalent to the max-min greedy matching problem.

[1] A valuation is said to be submodular if for every two sets S, T, $v(S) + v(T) \geq v(S \cup T) + v(S \cap T)$,.

[2] The full information assumption is sensible in repeated markets or in markets where the stakes are high and the designer may invest in learning the demand in the market before setting prices.

[3] We note that when the items player is adaptive (chooses the next item based on what happened in the past), the items player can ensure a perfect matching. This is done as follows: in each step, find a minimal tight set of items, and offer an arbitrary item from that set. Here, a set of items is tight if the number of buyers that want items in the set is equal to the size of the set.

1.4 Relation to Previous Work

Relation to online bipartite matching

The Max-Min Greedy Matching problem is a nonstandard version of online problems. In the standard online matching problem [12], the algorithm designer has control over the matching algorithm, but has no control over the arrival order of clients (vertices). Our setting can model a situation in which the designer (the maximizing player) has full control over the arrival order of clients (it knows which "items" in U each "client" in V wants, and it chooses π based on this knowledge), but no control over the matching algorithm (the minimizing player can choose the worst possible match in every step, effectively resulting in a permutation σ).

Karp et al. [12] introduced the *Ranking* algorithm which has a $1 - 1/e$ competitive ratio in the online bipartite matching setting. Translating this algorithm to our Max-Min Greedy setting, it amounts to simply selecting π at random, and then the minimizing player selects σ after seeing π. We show that there are bipartite graphs $G \in G_n$ for which with high probability over the random choice of π, there is a choice of σ resulting in $M_G[\sigma, \pi] \leq 1/2 + o(1)$. Karp et al. [12] also showed that no algorithm for online bipartite matching has a competitive ratio better than $1 - 1/e + o(1)$. This was shown by exhibiting a distribution over "difficult" graphs. Each graph in the support of this distribution has a unique perfect matching, and consequently (see Proposition 3), there is a permutation π in the Max-Min Greedy setting that ensures that all vertices are matched (regardless of σ). Hence neither the lower bounds nor the upper bounds known for the online matching model give useful bounds in the Max-Min Greedy model.

There are additional known results for online bipartite matching. For d-regular graphs, Cohen and Wajc [3] present a random algorithm that obtains $1-O(\sqrt{\log d}/\sqrt{d})$ in expectation, and a lower bound of $1 - O(1/\sqrt{d})$. This is in contrast to our Theorem 12 that shows that ρ is bounded away from 1 even when d is arbitrarily large. For general bipartite graphs, under random (rather than adversarial) arrival order, the deterministic greedy algorithm gives $1 - 1/e$, and no deterministic algorithm can obtain more than $3/4$ [10]. *Ranking* (which is a randomized algorithm) obtains at least 0.696 of the optimal matching [15] and at most 0.727 [11]. No random algorithm can obtain more than 0.823 [16].

Relation to pricing mechanisms

Our work is also related to the recent body of literature on pricing mechanisms. Motivated by the fact that in real-life situations one is often willing to trade optimality for simplicity, the study of simple mechanisms has gained a lot of interest in the literature on algorithmic mechanism design. One of the simplest forms of mechanisms is that of posted price mechanisms, where prices are associated with items and agents buy their most preferred bundles as they arrive. Pricing mechanisms have many advantages: they are simple, straightforward, and allow for asynchronous arrival and departure of buyers. Various forms of posted price mechanisms for welfare maximization have been proposed for various combinatorial settings [8, 5, 14, 6]. These mechanisms are divided along several axes, such as item vs. bundle pricing, static vs. dynamic pricing, and anonymous vs. personalized pricing. For any market with submodular valuations, one can obtain 1/2 of the optimal welfare by static, anonymous item prices [8]. Until the present paper, no better results than 1/2 were known even for markets with unit-demand valuations with $\{0, 1\}$ individual values. For a market with m identical items, there exists a pricing scheme that obtains at least $5/7 - 1/m$ of the optimal welfare for submodular valuations [6].

2 Proof of Main Result

The graph $G(U, V; E)$ with $|U| = |V| = n$ has a perfect matching M in which $u_i \in U$ is matched with $v_i \in V$ for every $1 \leq i \leq n$. For a given i, we refer to u_i and v_i as *partners* of each other. Given a set $S \subset V$, the set of neighbors of S is denoted as $N(S)$ (where necessarily $N(S) \subset U$). In this section we prove our main result.

▶ **Theorem 2.** *Given a bipartite graph $G(U, V; E)$ with a perfect matching $\{(u_i, v_i)\}$, there exists a permutation π that guarantees that the greedy matching will be of size at least $\frac{22}{43}n$, regardless of σ. Moreover, there is a polynomial time algorithm that chooses π with such a guarantee.*

Our proof approach is as follows. We shall first associate with G an auxiliary directed graph that we refer to as the spoiling graph $H(V, D)$. This notion by itself is not new – similar notions appeared also in previous related work. The new aspect related to the spoiling graph and the key to our approach is a notion of a *maximal path cover*. Given a maximal path cover of the spoiling graph (which as we show in Proposition 4, can be found in polynomial time), we partition the set V of vertices into four classes, depending on their roles in the maximal path cover. The classes are V_1 (singleton vertices), S (start vertices of paths), T (end vertices of paths), and I (intermediate vertices of paths). By considering several carefully chosen orders among the classes of vertices, and also of vertices within the classes, we obtain four possible candidate permutations for π, denoted $\pi_1, \pi_2, \pi_3, \pi_4$. We show that for every bipartite graph with a perfect matching, at least one of these permutations, if used as π, guarantees that the greedy matching will be of size at least $\frac{22}{43}n$, for every σ. Put in other words, if for each of $\{\pi_1, \pi_2, \pi_3, \pi_4\}$ there is a permutation over U for which the greedy matching is smaller than $\frac{22}{43}n$, this would imply (using properties listed in Lemma 5) that the path cover giving rise to these permutations was not maximal.

We now proceed to define the spoiling graph. Given $G(U, V; E)$, consider a directed graph $H(V, D)$ whose vertices are the set V, and whose set D of directed edges (arcs) is defined as follows: $(v_i, v_j) \in D$ iff $(u_j, v_i) \in E$. We refer to $H(V, D)$ as the *spoiling graph* for G, because arc $(v_i, v_j) \in D$ allows for the possibility that edge $(u_j, v_i) \in E$ is chosen into a matching M' in G, spoiling for v_j the possibility (offered by M) of being matched to u_j. Note that this spoiling effect may materialize in a (σ, π) matching only if v_i is ranked higher than v_j in π. Hence the spoiling graph conveys information that may be relevant to the choice of π.

As an example of the information that can be derived from the spoiling graph, consider the following proposition (whose proof can be also obtained as a special case of a result given in [4] and [13] for the more general case of Gross Substitutes valuations).

▶ **Proposition 3.** *If G has a unique perfect matching, then $\rho(G) = 1$.*

Proof. Let $u_i \in U$ and $v_i \in V$ be partners in the unique perfect matching M of G. We claim that the spoiling graph H of G is a *directed acyclic graph* (DAG). Suppose toward contradiction that H contains a simple directed cycle $v_{i_1}, v_{i_2}, \ldots, v_{i_\ell}, v_{i_1}$. This directed cycle corresponds to the cycle $u_{i_1}, v_{i_1}, u_{i_2}, v_{i_2}, \ldots, u_{i_\ell}, v_{i_\ell}, u_{i_1}$ in G. But removing the edges (u_{i_j}, v_{i_j}), $1 \leq j \leq \ell$ from M and adding the edges $(u_{i_j}, v_{i_{j+1}})$ to M (where $\ell + 1 = 1$) yields a different perfect matching, contradicting the uniqueness of M.

Since H is a DAG, we can topologically sort its vertices and choose a permutation π such that earlier vertices in the topological order have a lower rank in π. This ensures that for every directed edge (v, w) in H, v's partner will never prefer w over v. Thus, every vertex chooses its partner in M upon arrival. ◀

We now proceed to define the notion of a maximal path cover. A *directed path* P (whose length is denoted by $|P|$) in H is a sequence of $|P|$ vertices (say, $v_1, \ldots, v_{|P|}$) such that $(v_i, v_{i+1}) \in D$ for all $1 \leq i \leq |P| - 1$. A single vertex is a path of length 1. A *path cover* of H is a collection of vertex disjoint directed paths that covers all vertices in V. We consider the following operations that can transform a given path cover to a different one:

1. *Path merging*: Two paths can be merged into one longer path if $H(V, D)$ has an arc from the end of one path to the start of the other path.
2. *Path unbalancing*: Consider any two paths P and P' with $1 < |P| \leq |P'|$, let v_s and v_t denote the first and last vertices of P, and let v'_s and v'_t denote the first and last vertices of P'. If $(v_s, v'_s) \in D$ we may remove v_s from P and append it at the beginning of P'. Likewise, if $(v'_t, v_t) \in D$ we may remove v_t from P and append it at the end of P'.
3. *Rotation*: if there is a path P (say, $v_s, \ldots, v_t$) such that $(v_t, v_s) \in D$, we may add the arc (v_t, v_s) to P (obtaining a cycle), and then remove any other single arc from the resulting cycle to get a path P'. Observe that P' and P have the same vertex set, but they differ in their starting vertex along the cycle $v_s, \ldots, v_t, v_s$.

A path cover is *maximal* if no path merging operation and no path unbalancing can be applied to it, and furthermore, this continues to hold even after performing any single rotation operation.

▶ **Proposition 4.** *Given a bipartite graph $G(U, V; E)$ with a perfect matching $\{(u_i, v_i)\}$, a maximal path cover in the associated spoiling graph $H(V, D)$ can be found in $O(n^2)$ time.*

Proof. Start with the trivial path cover in which each vertex of V forms a path of length 1, and perform arbitrary path merging and path unbalancing operations (some of which are preceded by a single rotation operation) until no longer possible. The process must end within $O(n^2)$ operations, because each path merging and each path unbalancing operation increases the sum of squares of the lengths of the paths, and the sum of squares of the lengths is at most n^2. ◀

Given a maximal path cover of H (where p denotes the number of paths in the path cover), sort the paths in order of increasing lengths, breaking ties arbitrarily. Hence $1 \leq |P_1| \leq |P_2| \leq \ldots \leq |P_p|$. We consider the following classes of vertices of V:

1. *Singleton vertices* V_1. These are the vertices that belong to paths of length 1 in the given maximal path cover. Let $k = |V_1|$ denote the number of singleton vertices. Observe that $|P_k| = 1$ and $|P_{k+1}| > 1$.
2. *Other vertices* $V_2 = V \setminus V_1$. We partition V_2 into three subclasses of vertices:

   a. *Start vertices* S. These are the starting vertices of those paths that have length larger than 1. The start vertex of path j, for $k < j \leq p$, is denoted by s_j.
   b. *End vertices* T. These are the end vertices of those paths that have length larger than 1. The end vertex of path j, for $k < j \leq p$, is denoted by t_j.
   c. *Intermediate vertices* $I = V_2 \setminus (S \cup T)$.

▶ **Lemma 5.** *The classes of vertices listed above have the following properties:*
1. *There are no arcs in H between vertices of V_1. Hence no vertex of V_1 can be a spoiler vertex for another vertex in V_1.*
2. *There are no arcs in H from vertices of V_1 to vertices in S. Hence no vertex of V_1 can be a spoiler vertex for a vertex in S.*

3. *There are no arcs in H from vertices of T to vertices in V_1. Hence no vertex of T can be a spoiler vertex for a vertex in V_1.*
4. *For $i \neq j$, there are no arcs in H from any vertex $t_i \in T$ to any vertex $s_j \in S$. Hence no vertex of T can be a spoiler vertex for a vertex in S, unless they both belong to the same path in the given maximal path cover.*
5. *$(s_i, s_j) \notin D$ for any $s_i, s_j \in S$ with $i < j$. Hence s_i cannot be a spoiler vertex for s_j if $i < j$.*
6. *$(t_j, t_i) \notin D$ for any $t_i, t_j \in S$ with $i < j$. Hence t_j cannot be a spoiler vertex for t_i if $i < j$.*
7. *If for some $s_j \in S$ and $t_j \in T$ (where s_j and t_j are start and end vertices of the same path P_j) it holds that $(t_j, s_j) \in D$, then there are no arcs from $(T \setminus \{t_j\}) \cup V_1$ to any of the vertices of P_j, and likewise, no arcs from $s_{k+1}, \dots, s_{j-1}$ to any of the vertices of P_j.*

Proof. All properties follow from the maximality of the path cover. Properties 1,2,3 and 4 hold because otherwise one could perform a path merging operation. Properties 5 and 6 hold because otherwise one could perform a path unbalancing operation. Property 7 holds because otherwise one could perform a rotation operation for path P_j, followed either by a path merging operation (if there is an arc from $(T \setminus \{t_j\}) \cup V_1$ to any of the vertices of P_j) or a path unbalancing operation (if there is an arc from $s_{k+1}, \dots, s_{j-1}$ to any of the vertices of P_j). ◀

We now introduce additional notation. Considering only the arcs in D leading from V_2 to V_1, we let M_{21} denote the maximum matching among these arcs. In our analysis we shall consider three parameters:

1. ϵ_1: its value is such that $k = \left(\frac{1}{2} - \epsilon_1\right) n$ (recall that $k = |V_1|$ is the number of singleton paths in the maximal path cover). Observe that ϵ_1 might be negative.
2. ϵ_2: its value is such that $p = k + \epsilon_2 n = \left(\frac{1}{2} - \epsilon_1 + \epsilon_2\right) n$ (recall that p is the total number of paths in the maximal path cover). Necessarily, $\epsilon_2 \geq 0$.
3. ϵ_3: its value is such that $|M_{21}| = \left(\frac{1}{2} - \epsilon_3\right) n$. Necessarily, $\epsilon_3 \geq 0$.

Given the above classes of vertices, we consider four possible candidate permutations for π (denoted $\pi_1, \pi_2, \pi_3, \pi_4$, see below for details). Given some permutation π, we shall use the notation $\rho(\pi)$ to denote the fraction of vertices guaranteed to be matched under π. This fraction will be expressed as a function of the parameters ϵ_1, ϵ_2 and ϵ_3, and we will show that regardless of the value of these parameters, there must be some π with $\rho(\pi) \geq \frac{22}{43}$.

The following four lemmas present the four candidate permutations for π along with their corresponding guarantees. Whenever unspecified, the order within a set of vertices can be arbitrary; e.g., for two sets of vertices X, Y, $\pi = X, Y$ means that the set X precedes Y and the order within X, as well as the order within Y, is arbitrary.

▶ **Lemma 6.** *For G and $\pi_1 = V_2, V_1$,*

$$\rho(\pi_1) \geq \frac{1}{n}\left(|V_1| + \frac{|V_2|}{2} - \frac{|M_{21}|}{2}\right) = \frac{1}{2} - \frac{\epsilon_1}{2} + \frac{\epsilon_3}{2}.$$

Proof. Let σ be an arbitrary permutation over U. Let m denote the number of vertices in V_2 that are matched to vertices in U_1, where U_1 is the set of partners of V_1. Then $m \leq |M_{21}|$. Of the $|V_2| - m$ vertices of V_2 not matched to vertices in U_1, at least half are matched (because for every unmatched vertex from this set, its partner must be matched to a different vertex from this set). In addition, all those vertices of V_1 whose partner is not matched to V_2 are matched, because of property 1 of Lemma 5. Hence the total number of vertices matched is at least $m + \frac{|V_2| - m}{2} + |V_1| - m \geq |V_1| + \frac{|V_2|}{2} - \frac{|M_{21}|}{2}$, as desired. ◀

▶ **Lemma 7.** *For G and $\pi_2 = V_1, V_2$,*

$$\rho(\pi_2) \geq \frac{2}{3} - \frac{1}{3}(\epsilon_1 + \epsilon_3).$$

Proof. Let σ be an arbitrary permutation over U. All vertices in V_1 are matched because of property 1 of Lemma 5. As to the vertices in V_2, observe that $|N(V_2)| \geq |V_2| + |M_{21}|$, as the set $N(V_2)$ includes the $|V_2|$ partners of V_2, plus at least $|M_{21}|$ additional neighbors in U_1 (due to the matching M_{21}). Moreover, if x vertices are removed from V_2, the number of remaining neighbors is at least $|V_2| + |M_{21}| - 2x$, because each vertex of V_2 contributed at most two neighbors to the lower bound that we provided on the number of neighbors.

Let x denote the number of vertices in V_2 matched under (π, σ). Then the size of the matching is $|V_1| + x$, the number of unmatched vertices in V_2 is $|V_2| - x$, and they have at least $|V_2| + |M_{21}| - 2x$ neighbors which have to be matched. Since the number of matched vertices at each side is the same, we have that $|V_1| + x \geq |V_2| + |M_{21}| - 2x$.

We get that

$$\begin{aligned} 3x \;\geq\; |V_2| + |M_{21}| - |V_1| \;&=\; n\left(\frac{1}{2} + \epsilon_1\right) + n\left(\frac{1}{2} - \epsilon_3\right) - n\left(\frac{1}{2} - \epsilon_1\right) \\ &=\; \left(\frac{1}{2} + 2\epsilon_1 - \epsilon_3\right) n. \end{aligned}$$

Therefore, the size of the matching is at least

$$|V_1| + x \geq \left(\frac{1}{2} - \epsilon_1\right) n + \left(\frac{1}{6} + \frac{2\epsilon_1}{3} - \frac{\epsilon_3}{3}\right) n = \left(\frac{2}{3} - \frac{1}{3}(\epsilon_1 + \epsilon_3)\right) n.$$ ◀

▶ **Lemma 8.** *For G and $\pi_3 = t_p, \ldots, t_{k+1}, V_1, s_{k+1}, \ldots, s_p, I$,*

$$\rho(\pi_3) \geq \frac{2p - k}{n} = \frac{1}{2} - \epsilon_1 + 2\epsilon_2.$$

Proof. In π_3, we refer to the vertices of $T \cup V_1 \cup S$ as the *prefix* of π_3, and to the vertices of I as the *suffix*. Lemma 5 implies that within the prefix, the only arcs of H that go from an earlier vertex to a later vertex are of the form (t_j, s_j) (for a path P_j that can undergo a rotation). We claim that regardless of σ, all the prefix will be matched. As the length of this prefix is $2p - k$, this will prove the lemma.

It remains to prove the claim. Suppose first that in the above prefix there are no arcs of H that go from an earlier vertex to a later vertex. Then earlier vertices in this prefix cannot be spoiling vertices for later vertices. Hence indeed, regardless of σ, all the prefix will be matched.

Suppose now that in the prefix of π_3 there are arcs of H that go from an earlier vertex to a later vertex. As noted above, such an arc would be of the form (t_j, s_j). We need to show that even if t_j acts as a spoiling vertex for s_j under π_3 and σ, the vertex s_j will still be matched. Consider the path P_j, and let us rename its vertices as $x_1, \ldots, x_\ell$ (where previously we used $s_j = x_1$ and $t_j = x_\ell$). We wish to show the x_1 would be matched even if x_ℓ is matched to the partner of x_1. The path P_j implies that the partner of x_2 is a neighbor of x_1 in G. Hence x_1 will be matched if no vertex preceding $x_1 = s_j$ in π_3 is matched to the partner of x_2. By property 7 of Lemma 5, there is no arc in H from any of the vertices $T \cup V_1 \cap \{s_{k+1}, \ldots, s_{j-1}\} \setminus \{t_j\}$ to x_2, and consequently none of them can be matched to the partner of x_2. As to $t_j = x_\ell$, it might be a neighbor of the partner of x_2 (in fact, it could be that $\ell = 2$), but we already assumed that t_j is matched to the partner of x_1, and hence it is not matched to the partner of x_2. Hence no vertex preceding $x_1 = s_j$ in π_3 is matched to the partner of x_2, and hence s_j will be matched. ◀

Let V_e (V_o, respectively) denote those vertices of $S \cup I$ that are at even (odd, respectively) distance from the beginning of their respective path. Observe that $S \subset V_e$.

▶ **Lemma 9.** *For G and $\pi_4 = t_p, \ldots, t_{k+1}, V_1, V_o, V_e$,*

$$\rho(\pi_4) \geq \frac{5}{9} - \frac{p}{9n} = \frac{1}{2} + \frac{\epsilon_1}{9} - \frac{\epsilon_2}{9}.$$

Proof. Observe that $|V_e| \geq |V_o|$, because in every path (of length above 1) the vertices alternate in entering V_e and V_o, and start with V_e. Observe also that every vertex $v \in V_e$ contributes two distinct neighbors to $N(V_e)$: the partner of v, and the partner of the vertex that follows v on its path (note that the vertex that follows v is not in V_e). Likewise, every vertex $v \in V_o$ contributes two distinct neighbors to $N(V_o)$.

Regardless of σ, all p vertices of T and V_1 are matched, as in Lemma 8. For a given σ, let n_o be the number of vertices matched in V_o and let n_e be the number of vertices matched in V_e. Then, $|V_o| - n_o$, the number of unmatched vertices in V_o, satisfies $2(|V_o| - n_o) \leq p + n_o$, because the neighbors of the unmatched vertices in V_o need to be matched to earlier vertices in $T \cup V_1 \cup V_o$. Likewise, $|V_e| - n_e$, the number of unmatched vertices in V_e, satisfies $2(|V_e| - n_e) \leq p + n_o + n_e$. Adding two times the first inequality and three times the second we get that $4|V_o| + 6|V_e| - 4n_o - 6n_e \leq 5p + 5n_o + 3n_e$. Using $|V_o| + |V_e| = n - p$ and $|V_e| \geq |V_o|$, we can replace $4|V_o| + 6|V_e|$ by $5(n-p)$, implying that $9(p + n_o + n_e) \geq 5n - p$, as desired. ◀

We can now prove Theorem 2.

Proof. Observe that $\rho(G) \geq \max_{i \in [1,4]}[\rho(\pi_i)]$.

By Lemma 6 we have: $\rho(\pi_1) \geq \frac{1}{2} - \frac{\epsilon_1}{2} + \frac{\epsilon_3}{2}$.

By Lemma 7 we have: $\rho(\pi_2) \geq \frac{2}{3} - \frac{1}{3}(\epsilon_1 + \epsilon_3)$.

By Lemma 8 we have: $\rho(\pi_3) \geq \frac{1}{2} - \epsilon_1 + 2\epsilon_2$.

By Lemma 9 we have: $\rho(\pi_4) \geq \frac{1}{2} + \frac{\epsilon_1}{9} - \frac{\epsilon_2}{9}$.

Taking a weighted average of the lower bounds provided by the four lemmas, with weights $\frac{2}{43}$, $\frac{3}{43}$, $\frac{2}{43}$, $\frac{36}{43}$, respectively, results in a weighted average of $\frac{22}{43}$. Hence regardless, of the values of ϵ_1, ϵ_2 and ϵ_3, at least one of the lemmas gives $\rho(G) \geq \frac{22}{43}$. For $\epsilon_1 = \frac{19}{86}$, $\epsilon_2 = \frac{10}{86}$ and $\epsilon_3 = \frac{21}{86}$, none of the lemmas implies a bound better than $\frac{1}{2} + \frac{1}{86} = \frac{22}{43}$.

The above analysis leads to a polynomial time algorithm for finding π that ensures $\rho(G) \geq \frac{22}{43}$. A maximal path cover of $H(V, D)$ can be found in polynomial time by Proposition 4. Thereafter, the sets V_1, S, T, V_e and V_o can easily be determined, and likewise, the values of ϵ_1 and ϵ_2 can be easily computed. A maximum matching M_{21} (from V_2 to V_1 in H) can be computed in polynomial time using any standard algorithm for maximum bipartite matching. Thereafter, ϵ_3 can be easily computed. Given the values ϵ_1, ϵ_2 and ϵ_3, one can determine which of π_1, π_2, π_3 or π_4 provides a higher lower bound on ρ, and use that permutation as π. ◀

3 Regular Graphs

In this section we consider the case where $G(U, V; E)$ is a d-regular bipartite graph with $2n$ vertices. Given that such graphs have d edge disjoint perfect matchings, one can hope to achieve higher values for ρ for these graphs.

3.1 Positive Result

The following known proposition (see for example [18]) establishes a lower bound on ρ, as a function of d. A proof is provided for completeness.

▶ **Proposition 10.** *For every d-regular graph $G \in G_n$, it holds that $\rho[G] \geq \frac{d}{2d-1}$.*

Proof. Since the greedy algorithm produces a maximal matching, it suffices to show that every maximal matching in a d-regular graph has size at least $\frac{d}{2d-1}n$. To see this, let $S \subset U$ and $T \subset V$ be the sets of unmatched nodes in an arbitrary maximal matching, and suppose $|S| = |T| = (1-\alpha)n$. The nodes in S, T must form an independent set. Consider the size of the edge set connecting S and $V \setminus T$. On the one hand, this size equals $(1-\alpha)nd$ (since all edges from S go to $V \setminus T$); on the other hand, this size is at most $\alpha n(d-1)$ (since at least one edge from each node in $V \setminus T$ goes to $U \setminus S$). Thus, $(1-\alpha)nd \leq \alpha n(d-1)$, implying that $\alpha \geq d/(2d-1)$. Hence we have that $|M_G[\sigma, \pi]| \geq \frac{d}{2d-1}n$, for every π. ◀

Remark: For every d there exists a d-regular graph with a perfect matching that admits a maximal matching of size $\frac{d}{2d-1}n$. Suppose that $n = 2d-1$, and consider a d-regular graph where $|S| = |T| = d-1$ for some $S \subset U, T \subset V$, every node in $U \setminus S$ is connected to a single, different node in $V \setminus T$, and to all $d-1$ nodes in T, and every node in $V \setminus T$ is connected to a single, different node in $U \setminus S$, and to all $d-1$ nodes in S. The perfect matching between $U \setminus S$ and $V \setminus T$ is a maximal matching of size $\frac{d}{2d-1}n$.

The lower bound of Proposition 10 approaches $\frac{1}{2}$ from above as d grows. The following theorem shows that there exists some permutation π that ensures that the fraction of matched vertices approaches 5/9. This is a direct corollary from Lemma 9 and a theorem in [7].

▶ **Corollary 11.** *For d-regulars bipartite graphs, $\rho \geq \frac{5}{9} - O(\frac{1}{\sqrt{d}})$.*

Proof. Theorem 3 in [7] shows that every n-vertex d-regular graph has a path cover (referred to as a *linear forest*) with $p = O(\frac{n}{\sqrt{d}})$ paths. By Lemma 9, $\rho(G) \geq \frac{5}{9} - O(\frac{1}{\sqrt{d}})$. ◀

Remarks.

1. For small d, the bound of $\rho \geq \frac{d}{2d-1}$ which holds for every maximal matching is stronger than the bound in Corollary 11.
2. The proof of Corollary 11 extends to graphs that are nearly d-regular, by using Theorem 5 from [7].
3. For d-regular graphs, conjectures mentioned in [7] combined with our proof approach suggest that $\rho \geq \frac{5}{9} - O(\frac{1}{d})$.

3.2 Negative Result

The following example shows that even in a regular graph with arbitrarily high degree, there may be no permutation π that ensures to match more than a fraction 8/9 of the vertices.

▶ **Theorem 12.** *For every integers $d, t \geq 1$, there is a regular bipartite graph $G_{d,t}$ of even degree $2d$ and $n = 3dt$ vertices on each side such that $\rho(G_{d,t}) \leq \frac{8}{9}$.*

Proof. Consider a regular bipartite graph $G(U, V; E)$ with even degree $2d$, and $3d$ vertices on each side. To define the edge set, let $U = U_1 \cup U_2 \cup U_3$ with each U_i of cardinality d, and similarly $V = V_1 \cup V_2 \cup V_3$ with each V_i of cardinality d. For every $i \neq j$, we have a complete bipartite graph between U_i and V_j, and for every i, there are no edges between U_i and V_i.

Let π be an arbitrary permutation over V, let S be the first $2d$ vertices in π, and let T be the last d vertices. Let i be such that $|V_i \cap T|$ is largest (breaking ties arbitrarily). Without loss of generality we may assume that $i = 3$, and then $|V_3 \cap T| \geq d/3$. Hall's condition implies that there is a perfect matching between $U_1 \cup U_2$ and S (and more generally, between $U_1 \cup U_2$ and any $2d$ vertices from V). Hence one can choose a permutation σ over U whose first $2d$ vertices are $U_1 \cup U_2$ that will match the vertices of S one by one. Thereafter, the vertices of $T \cap V_3$ will remain unmatched.

To get the graph $G_{d,t}$ claimed in the theorem, take t disjoint copies of $G(U, V; E)$ above. ◀

4 Random Permutation

In this section we consider scenarios in which the maximizing player is unaware of the graph structure. In such scenarios, the best she can do is impose a random permutation over the vertices in V.

We first show that there exists a graph $G \in G_n$ for which a random permutation does not match significantly more than a half of the vertices, even if every vertex has a high degree.

▶ **Proposition 13.** *There exists a bipartite graph $G(U, V; E) \in G_n$ such that a random permutation gets $\rho(G) = \frac{1}{2} + o(1)$ almost surely.*

Proof. Consider the graph $G(U, V; E)$, where $U = (U_1, U_2)$, $V = (V_1, V_2)$, and each of U_1, U_2, V_1, V_2 is of size $n/2$. The set of edges constitutes of a perfect matching between U_1 and V_1, a perfect matching between U_2 and V_2, and a bi-clique between U_1 and V_2. Let π be a random permutation. With high probability, for each vertex $v_1 \in V_1$, except for $\sim \sqrt{n}$ such vertices, we can associate a unique vertex $v_2 \in V_2$ that precedes v_1 in π. Let $S \subseteq V_1$ denote this set. Consider an arrival order σ in which agents in U_1 arrive first, with a vertex u_{1j} preceding a vertex $u_{1j'}$ if $\pi(v_{2j}) < \pi(v_{2j'})$. Every vertex in U_1 such that its neighbor in V_1 (according to the perfect matching) belongs to S will be matched to the corresponding vertex in V_2. Therefore, all but $\sim \sqrt{n}$ vertices of V_1 remain unmatched, and the size of the matching is $n(1/2 + o(1))$, whereas $OPT = n$. ◀

In the above example, if the vertices of V with degree 1 are placed in the prefix of π, then the obtained matching is optimal. This might suggest that prioritizing low degree vertices in π (and randomizing within sets of vertices of comparable degrees) leads to good performance. However, the example above can be transformed into one where all vertices in V have the same degree. To see this, consider a graph where vertices are partitioned into sets of perfect matchings of size $\sqrt{n}$, $\{(U_{11}, V_{11}), \ldots, (U_{1\sqrt{n}}, V_{1\sqrt{n}}), (U_{21}, V_{21}), \ldots, (U_{2\sqrt{n}}, V_{2\sqrt{n}})\}$. Each V_{1i} is also connected in a bi-clique to U_{2i}, and in addition, there are sets U', V' of size $\sqrt{n}$ each connected to the vertices of the other side to balance out the degrees. A similar argument shows that in this graph, a random permutation performs badly as well.

In contrast to the last examples, in some classes of graphs, a random permutation guarantees to match a fraction of the vertices that is bounded away from a half. This is the case, for example, in Hamiltonian graphs. The formal statement and proof are deferred to Section 6.

5 Finding a perfect π

A permutation π over V is said to be *perfect* if for every permutation σ over U, $|M_G[\sigma, \pi]| = n$. A vertex $v \in V$ is *good* with respect to a set of vertices $S \subset V$ if there is no matching between $N(v)$ and S. Given permutation π over V, let $\overline{\pi}(v)$ be the set of vertices preceding v in π.

▶ **Observation 14.** *π is perfect if and only if every vertex $v \in V$ is good with respect to $\overline{\pi}(v)$.*

Proof. Suppose there exists a vertex $v \in V$ that is not good. Then, consider a permutation σ where the vertices in $N(v)$ are placed first, in an order corresponding to the rank (according to π) of their partners in the perfect matching between $N(v)$ and $\overline{\pi}(v)$. In such a σ, v will not be matched. Now suppose that all vertices in V are good. Then, for every σ, for every $v \in V$, there exists a vertex $u \in U$ that is not matched to a vertex in $\overline{\pi}(v)$; therefore v will surely be matched. ◀

We present an algorithm that finds a perfect π if one exists, and claims that no such π exists otherwise.

Checking whether v_i is good can be done in polynomial time by running a maximum matching algorithm on $N(v_i)$ and S_i.

An algorithm for finding a perfect π.

1. Let $S_1 = V$.
2. In iteration $i = 1, \dots, n$:
   a. Find an arbitrary good vertex $v_i \in S_i$ with respect to $S_i \setminus \{v_i\}$, and place it in rank $n - i + 1$ in π.
   b. Set $S_{i+1} = S_i \setminus \{v_i\}$.

▶ **Lemma 15.** *If there exists a perfect π, then the algorithm is guaranteed to find a good v_i in every iteration i.*

Proof. Consider some perfect permutation π (not necessarily the one produced by our algorithm), and the suffix $V_{i-1} = v_{i-1}, v_{i-2}, \dots, v_1$ of vertices chosen in the first $i-1$ iterations of the algorithm (of course, there must be a good v_1 at the first iteration, otherwise there is no perfect π). Let π' be the permutation that places $v_{i-1}, v_{i-2}, \dots, v_1$ as the lowest ranked vertices in the same order as the algorithm picked them, and places all other vertices of $V \setminus V_{i-1}$ in a higher rank than V_{i-1} according to their internal order in π.

Since every v_j, $1 \leq j \leq i-1$, is good with respect to $V \setminus \{v_1, \dots, v_j\}$, then clearly v_j is good with respect to $\overline{\pi'}(v_j)$ (since $\overline{\pi'}(v_j) = V \setminus \{v_1, \dots, v_j\}$). Now consider a vertex $v \in V \setminus V_{i-1}$. This vertex is good with respect to $\overline{\pi}(v)$, and since $\overline{\pi'}(v) \subseteq \overline{\pi}(v)$, it is clear that v is good with respect to $\overline{\pi'}(v)$. It follows that π' is perfect as well.

Let v'_i be the vertex ranked in position $n - i + 1$ in π'. Since π' is perfect, this vertex is good with respect to $\overline{\pi'}(v'_i)$. But since $\overline{\pi'}(v'_i) \cup \{v'_i\}$ is exactly the set S_i in iteration i, it is guaranteed that the algorithm can find a good v_i in this iteration. ◀

Let π be the permutation computed by the algorithm. Since every vertex v is good with respect to $\overline{\pi}(v)$, it follows from Observation 14 that π is perfect, and Proposition 1 follows.

6 Hamiltonian Bipartite Graphs

In this section we establish two results about Hamiltonian graphs. First, we show that $\rho \geq \frac{5}{9}$. Note that, since for the case of a Hamiltonian graph there exists a path cover using only a single path (*i.e.*, $p = 1$), Lemma 9 directly implies that $\rho \geq \frac{5}{9} - \frac{1}{9n}$. Theorem 16 improves this bound to 5/9. Second, we show that for Hamiltonian graphs, even a random permutation π ensures a ratio that is bounded away from $\frac{1}{2}$ (this is in contrast to general graphs, see Section 4).

▶ **Theorem 16.** *For every Hamiltonian graph G, it holds that $\rho \geq \frac{5}{9}$.*

Proof. Consider a Hamiltonian graph G and a Hamiltonian cycle $u_1, v_1, u_2, v_2, \dots, u_n, v_n, u_1$ that traverses through its vertices. Let $V_o = \{v_i \;:\; i = 2\ell+1,\; \ell \in \mathbb{N},\; i \le n\}$ be the set of odd labeled vertices of the cycle, and $V_e = V \setminus V_o$.

We first claim that if the number of vertices is even ($|V_o| = |V_e| = \frac{n}{2}$), then $\pi = V_e, V_o$ (and in fact, also $\pi = V_o, V_e$) ensures that $\rho \ge 5/9$. Let n_e (respectively, n_o) be the number of vertices of V_e (respectively, V_o) matched in $M_G[\sigma, \pi]$ defined using $\pi = (V_e, V_o)$ and an arbitrary σ. Similar to the proof of Lemma 9, it is easy to see that each vertex in V_e contributes two distinct neighbors to $N(V_e)$, and each vertex in V_o contributes two distinct neighbors to $N(V_o)$ (the difference from the proof of Lemma 9 is that this property also holds for $v_1 \in V_o$ and $v_n \in V_e$, and this follows because v_1 and v_n contribute u_1 to $N(V_o)$ and $N(V_e)$, respectively). The number of unmatched vertices in V_o, namely $|V_o| - n_o$, satisfies

$$2(|V_o| - n_o) \le n_o,$$

because the neighbors of the unmatched vertices in V_o must be matched to vertices in V_o, as they precede the vertices in V_e in π. Likewise, the number of unmatched vertices in N_e, namely $|V_e| - n_e$, satisfies

$$2(|V_e| - n_e) \le n_e + n_o.$$

Adding two times the first inequality and three times the second, we get

$$4|V_o| + 6|V_e| \le 9(n_o + n_e) \;\Rightarrow\; \frac{5}{9} \cdot n \le n_o + n_e.$$

As $|V| = n$ and $|M_G[\sigma, \pi]| = n_o + n_e$, this implies that $\rho \ge \frac{5}{9}$.

We now handle the case where n is odd. Lemma 9 ensures that $\frac{5}{9} \cdot n - \frac{1}{9}$ of the vertices are matched by π_4 when the path cover is of a single path. If $\frac{5}{9} \cdot n - \frac{1}{9}$ is not integral, then $\lceil \frac{5}{9} \cdot n - \frac{1}{9} \rceil$ is at least $\frac{5}{9} \cdot n$, thus $\rho \ge \frac{5}{9}$. Therefore, it only remains to handle the case where $\frac{5}{9} \cdot n - \frac{1}{9}$ is integral; namely where $n = 18\ell + 11$ for some integer ℓ. In this case, we show that $\pi = V_e, V_o$ ensures that $|M[\sigma, \pi]| > \frac{5}{9} \cdot n$ for every σ. Since $n = 18\ell + 11$, it holds that $|V_o| = 9\ell + 6$ and $|V_e| = 9\ell + 5$. As in the case where n is even, every vertex in V_e contributes two distinct neighbors to $N(V_e)$. As for V_o, every vertex in $V_o \setminus \{v_1, v_n\}$ also contributes two distinct neighbors to $N(V_o)$, and v_1 and v_n contribute (together) to $N(V_o)$ three additional distinct vertices (since they share a vertex along the Hamiltonian cycles). Using the same reasoning as before, it follows that

$$2(|V_e| - n_e) \le n_e \Rightarrow 18\ell + 10 \le 3 \cdot n_e \Rightarrow n_e \ge 6\ell + 3\frac{1}{3}.$$

Since n_e is integral, this implies that

$$n_e \ge 6\ell + 4. \tag{1}$$

Again, for $|V_o| - n_o$ we have $2(|V_o| - n_o) - 1 \le n_o + n_e$. Rearranging gives us

$$n_e + 3n_o \ge 18\ell + 11.$$

Adding twice Inequality (1) to the last inequality yields

$$n_e + n_o \ge 10\ell + 6\frac{1}{3} > 10\ell + 6\frac{1}{9} = \frac{5}{9} \cdot |V|,$$

which implies $\rho > \frac{5}{9}$. This concludes the proof. ◀

Next, we show that for the case of a Hamiltonian graph, a random permutation yields a $\rho > 1/2$.

▶ **Theorem 17.** *Consider choosing a permutation π uniformly at random. For every Hamiltonian graph G, it holds that $E_\pi[\min_\sigma[|M_G[\sigma, \pi]|]] > 0.5012$.*

We explain the proof approach here, and present the full details in Appendix A.

Proof Approach

We first provide a high level overview of our proof approach.

A permutation π (over V) is said to be *safe* for a set $S \subset V$ if for every permutation σ (over U) the greedy process matches at least one vertex in S (i.e., no σ leaves all vertices in S unmatched). Fix some constant ϵ. In order to establish that $\rho \geq (1/2 + \epsilon)$, we need to show that there exists a permutation π that is safe for every set S of size $(1/2 - \epsilon)n$. Our proof approach is the following: we show that for a permutation π chosen uniformly at random, the expected number (expectation taken over choice of π) of sets of size $(1/2 - \epsilon)n$ for which *pi* is unsafe is smaller than 1. This implies that there exists a permutation π that is safe for all sets of size $(1/2 - \epsilon)n$, as desired.

First, we define a collection of sets that can potentially remain unmatched ("bad" sets). Let B_ϵ denote the set of all sets $S \subset U$ of size $(1/2 - \epsilon)n$ such that there exists a permutation π that is unsafe for S.

Second, for a given set S and permutation π we identify a sufficient condition for π to be safe for S. Let $S' \subset S$ be the lowest αn vertices in S (according to π), let v' be the last vertex in S' (i.e., the vertex with rank αn in S'), and let P be the set of vertices in $V - S'$ that precede v' in π. We claim that if the size of P is smaller than the size of $N(S')$ (the neighbors of S'), then π is safe for S. To see this, assume by way of contradiction that π is unsafe for S'. This implies that every vertex in $N(S')$ is matched to a vertex in $V - S'$. Since there are strictly less than $|N(S')|$ vertices in $V - S'$ that precede v', at least one of the vertices in $N(S')$ must be matched to a vertex higher than v'. But, this vertex has a neighbor in S' with lower rank, contradicting the greedy process.

We now proceed by establishing the following three lemmas:

- Few bad sets lemma: the size of B_ϵ is at most $n_B = n_B(\epsilon)$.
- *Expansion lemma:* given a set $S \subset V$ and parameters α, β, the probability (over a random choice of π) that the lowest αn vertices in S have less than βn neighbors is at most $p = p(\alpha, \beta)$.
- *Good order lemma:* given a set $S \subset V$ and parameters α, β, the probability (over a random choice of π) that the $(\alpha n)^{th}$ lowest vertex in S is higher than βn vertices in $V \setminus S$ is at most $q = q(\alpha, \beta)$.

The three lemmas are combined as follows. For a given set S, due to the sufficient condition identified above, it follows from the union bound that the probability that a uniformly random permutation π is unsafe for S is at most $p + q$. Applying the union bound once more over all bad sets (at most n_B sets, as implied by the few bad sets lemma), implies that the probability that a uniformly random permutation π is unsafe for some set of size $(1/2 - \epsilon)n$ is at most $n_B(p + q)$. Thus, to conclude the proof, it remains to find parameters such that $n_B(p + q) < 1$.

The good order lemma is independent of the graph structure. In contrast, the expansion lemma and the few bad sets lemma rely heavily on the structure of the graph. As it turns out, Hamiltonian graphs have properties that enable us to establish the two lemmas with good parameters.

References

1 Allan Borodin, Denis Pankratov, and Amirali Salehi-Abari. On Conceptually Simple Algorithms for Variants of Online Bipartite Matching. In *Approximation and Online Algorithms - 15th International Workshop, WAOA 2017, Vienna, Austria, September 7-8, 2017, Revised Selected Papers*, pages 253–268, 2017. doi:10.1007/978-3-319-89441-6_19.

2 Miroslav Chlebík and Janka Chlebíková. Approximation hardness of edge dominating set problems. *J. Comb. Optim.*, 11(3):279–290, 2006. doi:10.1007/s10878-006-7908-0.

3 Ilan Reuven Cohen and David Wajc. Randomized Online Matching in Regular Graphs. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pages 960–979, 2018. doi:10.1137/1.9781611975031.62.

4 Vincent Cohen-Addad, Alon Eden, Michal Feldman, and Amos Fiat. The Invisible Hand of Dynamic Market Pricing. In *Proceedings of the 2016 ACM Conference on Economics and Computation, EC '16, Maastricht, The Netherlands, July 24-28, 2016*, pages 383–400, 2016. doi:10.1145/2940716.2940730.

5 Paul Dütting, Michal Feldman, Thomas Kesselheim, and Brendan Lucier. Posted Prices, Smoothness, and Combinatorial Prophet Inequalities. *CoRR*, abs/1612.03161, 2016. arXiv:1612.03161.

6 Tomer Ezra, Michal Feldman, Tim Roughgarden, and Warut Suksompong. Pricing Identical Items. *CoRR*, abs/1705.06623, 2017. arXiv:1705.06623.

7 Uriel Feige, R. Ravi, and Mohit Singh. Short Tours through Large Linear Forests. In *Integer Programming and Combinatorial Optimization - 17th International Conference, IPCO 2014, Bonn, Germany, June 23-25, 2014. Proceedings*, pages 273–284, 2014. doi:10.1007/978-3-319-07557-0_23.

8 Michal Feldman, Nick Gravin, and Brendan Lucier. Combinatorial Auctions via Posted Prices. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, pages 123–135, 2015. doi:10.1137/1.9781611973730.10.

9 David Gale and L. S. Shapley. College Admissions and the Stability of Marriage. *American Math. Monthly*, 69:9–15, 1962.

10 Gagan Goel and Aranyak Mehta. Online Budgeted Matching in Random Input Models with Applications to Adwords. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '08, pages 982–991, Philadelphia, PA, USA, 2008. Society for Industrial and Applied Mathematics. URL: http://dl.acm.org/citation.cfm?id=1347082.1347189.

11 Chinmay Karande, Aranyak Mehta, and Pushkar Tripathi. Online Bipartite Matching with Unknown Distributions. In *Proceedings of the Forty-third Annual ACM Symposium on Theory of Computing*, STOC '11, pages 587–596, New York, NY, USA, 2011. ACM. doi:10.1145/1993636.1993715.

12 R. M. Karp, U. V. Vazirani, and V. V. Vazirani. An Optimal Algorithm for On-line Bipartite Matching. In *Proceedings of the Twenty-second Annual ACM Symposium on Theory of Computing*, STOC '90, pages 352–358, New York, NY, USA, 1990. ACM. doi:10.1145/100216.100262.

13 Renato Paes Leme and Sam Chiu-wai Wong. Computing Walrasian Equilibria: Fast Algorithms and Structural Properties. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 632–651, 2017. doi:10.1137/1.9781611974782.42.

14 Brendan Lucier. An economic view of prophet inequalities. *SIGecom Exchanges*, 16(1):24–47, 2017. doi:10.1145/3144722.3144725.

15 Mohammad Mahdian and Qiqi Yan. Online Bipartite Matching with Random Arrivals: An Approach Based on Strongly Factor-revealing LPs. In *Proceedings of the Forty-third Annual ACM Symposium on Theory of Computing*, STOC '11, pages 597–606, New York, NY, USA, 2011. ACM. doi:10.1145/1993636.1993716.

16 Vahideh H. Manshadi, Shayan Oveis Gharan, and Amin Saberi. Online Stochastic Matching: Online Actions Based on Offline Statistics. *Mathematics of Operations Research*, 37(4):559–573, 2012. URL: `http://www.jstor.org/stable/23358636`.

17 Aranyak Mehta. Online Matching and Ad Allocation. *Foundations and Trends in Theoretical Computer Science*, 8(4):265–368, 2013. `doi:10.1561/0400000057`.

18 Joseph Naor and David Wajc. Near-Optimum Online Ad Allocation for Targeted Advertising. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation, EC '15, Portland, OR, USA, June 15-19, 2015*, pages 131–148, 2015. `doi:10.1145/2764468.2764482`.

A Full Proof of Theorem 17

Throughout this section we use $H(\cdot)$ to denote the binary entropy function; i.e., given a constant $p \in (0,1)$, $H(p) = -p \log_2 p - (1-p)\log_2(1-p)$.

▶ **Fact 18** (Stirling's Approximation)**.** *As $n \to \infty$,*

$$n! = (1+o(1))\sqrt{2\pi n}\left(\frac{n}{e}\right)^n.$$

Using Stirling's Approximation, one can derive the following bound.

▶ **Fact 19.** *For n and $k = pn$ for some constant $p \in (0,1)$,*

$$\binom{n}{k} = 2^{(H(p)+o(1))n}, \tag{2}$$

where $H(p) = -p\log_2 p - (1-p)\log_2(1-p)$ is the binary entropy function.

We first establish the good order lemma, which is independent of the graph structure.

▶ **Lemma 20** (Good order lemma)**.** *Let $\alpha < \beta < 1$, $\rho = \frac{1}{2} + \epsilon$ for some $\epsilon > 0$ and $\bar{\rho} = 1 - \rho$ such that $\frac{\beta}{\alpha} > \frac{\rho}{\bar{\rho}}$. Let $S \subset V$ be a set of size $\bar{\rho}n$. The probability that in a random permutation π there are at least βn vertices of $V \setminus S$ before αn vertices from S is at most $2^{-(H(\alpha+\beta)-H(\frac{\alpha}{\bar{\rho}})\bar{\rho}-H(\frac{\beta}{\rho})\rho-o(1))n}$.*

Proof. We first analyze the case that in the first $(\alpha+\beta)n$ vertices in π there are *exactly* αn vertices from S. The number of possibilities for this case is $\binom{\bar{\rho}n}{\alpha n}\binom{\rho n}{\beta n}$.

Let $\beta' = \beta + x$ and $\alpha' = \alpha - x$. By the conditions on α, β and ϵ, we have that $\frac{\beta'}{\alpha'} \geq \frac{\beta}{\alpha} \geq \frac{\rho}{\bar{\rho}}$. Therefore,

$$\beta'\bar{\rho} \geq \alpha'\rho \Rightarrow \beta'\bar{\rho} - \alpha'\beta' \geq \alpha'\rho - \alpha'\beta' \Rightarrow \frac{(\bar{\rho}-\alpha')}{\alpha'} \cdot \frac{\beta'}{(\rho - \beta')} \geq 1$$

$$\Rightarrow \frac{(\bar{\rho}n - \alpha'n + 1)}{\alpha'n} \cdot \frac{\beta'n+1}{(\rho n - \beta'n)} \geq 1 \iff \frac{\binom{\bar{\rho}n}{\alpha'n}\binom{\rho n}{\beta'n}}{\binom{\bar{\rho}n}{\alpha'n-1}\binom{\rho n}{\beta'n+1}} > 1.$$

It follows that $\binom{\bar{\rho}n}{\alpha n}\binom{\rho n}{\beta n} > \binom{\bar{\rho}n}{\alpha' n}\binom{\rho n}{\beta' n}$ for every $\alpha' < \alpha$ and $\beta' > \beta$ such that $\alpha + \beta = \alpha' + \beta'$. Therefore, the probability to have at most αn vertices from S in the first $(\alpha+\beta)n$ vertices in π is at most

$$\begin{aligned}\frac{\alpha n \cdot \binom{\bar{\rho}n}{\alpha n}\binom{\rho n}{\beta n}}{\binom{n}{(\alpha+\beta)n}} &= \frac{2^{(H(\frac{\alpha}{\bar{\rho}})\bar{\rho}+H(\frac{\beta}{\rho})\rho+o(1))n}}{2^{(H(\alpha+\beta)+o(1))n}} \\ &= 2^{-(H(\alpha+\beta)-H(\frac{\alpha}{\bar{\rho}})\bar{\rho}-H(\frac{\beta}{\rho})\rho-o(1))n},\end{aligned}$$

where the first equality follows Fact 19. ◀

Let $\rho = \frac{1}{2} + \epsilon$ for some constant $\epsilon > 0$, $\bar{\rho} = 1 - \rho = \frac{1}{2} - \epsilon$. The next lemma will be used in order to prove the few bad sets lemma and the expansion lemma. It uses the existence a Hamiltonian cycle in the graph in order to claim that most sets will have a large number of neighbors. Therefore, a random set will have a large expansion. In addition, there will be few sets of size $(\frac{1}{2} - \epsilon)n$ with less than $(\frac{1}{2} + \epsilon)n$ neighbors (i.e., a few bad sets).

▶ **Lemma 21.** *Let $\alpha \in (0, 1/2)$ and $\beta \in (\alpha, 1)$ be two constants such that $\delta = \beta - \alpha < \alpha/2$. The number of sets S of size αn where $|N(S)| \leq \beta n$ is at most $2^{(\alpha H(\frac{\delta}{\alpha}) + (1-\alpha)H(\frac{\delta}{(1-\alpha)}) + o(1))n}$.*

Proof. Consider a Hamiltonian cycle that traverses through the graph's vertices $H = (v_1, u_1, v_2, u_2, \ldots, v_n, u_n, v_1)$, where $\{v_i\}_{i \in [n]} = V$ and $\{u_i\}_{i \in [n]} = U$. Let S be some set of vertices from V of cardinality ρn. Note that in the cycle H, each vertex v of S has two neighbors, where one of these neighbors is joined with an adjacent vertex from V in the cycle. Therefore, the number of neighbors of a sequence of k consecutive vertices of V in H is $k + 1$. Thus, the set $N(S)$ is of size αn plus the number of consecutive blocks of vertices from V chosen.

We bound the number of ways to pick at most δn consecutive blocks of vertices from V. We first bound the number of ways to pick *exactly* δn such blocks. In this case, the αn chosen elements have to be within δn blocks. The number of ways to partition αn elements to δn noen empty blocks is $\binom{\alpha n - 1}{\delta n - 1}$. After deciding the number of elements in each block, we need to figure out their location along the Hamiltonian cycle. $(1 - \alpha)n$ elements reside outside of the blocks of chosen αn elements. We need to chose the location of the first block in H (for which there are n possibilities), and then the number of element between each block, where two blocks are separated by at least one element. The latter is equivalent to splitting $(1-\alpha)n$ elements into δn non empty bins, for which there are $\binom{(1-\alpha)n-1}{\delta n - 1}$ possibilities. Overall, there are $n\binom{\alpha n - 1}{\delta n - 1}\binom{(1-\alpha)n-1}{\delta n - 1}$ such possibilities[4].

For $\delta' < \delta$, one can similarly devise the bound of $n\binom{\alpha n - 1}{\delta' n - 1}\binom{(1-\alpha)n-1}{\delta' n - 1}$ which is smaller than $n\binom{\alpha n - 1}{\delta n - 1}\binom{(1-\alpha)n-1}{\delta n - 1}$ by our conditions on α and δ. Overall, we can bound the number of ways to pick at most δn consecutive blocks of vertices from V by

$$\begin{aligned}
\delta n^2 \binom{\alpha n - 1}{\delta n - 1}\binom{(1-\alpha)n - 1}{\delta n - 1} &< \delta n^2 \binom{\alpha n}{\delta n}\binom{(1-\alpha)n}{\delta n} \\
&= 2^{o(1)n} \cdot 2^{(H(\frac{\delta}{\alpha}) + o(1))\alpha n} \cdot 2^{(H(\frac{\delta}{(1-\alpha)}) + o(1))(1-\alpha)n} \\
&= 2^{(\alpha H(\frac{\delta}{\alpha}) + (1-\alpha)H(\frac{\delta}{(1-\alpha)}) + o(1))n},
\end{aligned}$$

where the first equality follows Fact 19. ◀

The expansion and few bad sets lemmas are obtained as direct corollaries of Lemma 21.

▶ **Lemma 22** (Few bad sets Lemma for Hamiltonian graphs)**.** *Let ϵ be a constant such that $\epsilon < 0.1$. The number of bad sets in any Hamiltonian graph is at most*

$$|B_\epsilon| \leq 2^{\left(\bar{\rho} H(\frac{2\epsilon}{\bar{\rho}}) + \rho H(\frac{2\epsilon}{\rho}) + o(1)\right)n}.$$

[4] Notice there's some over-counting in this argument, but this bound suffices for our purpose.

Proof. Notice that if a set S of size $\bar{\rho}n = (\frac{1}{2} - \epsilon)n$ has more than ρn neighbors, it cannot be left unmatched, since at least one of it's neighbors will could not be matched to $V \setminus S$. A direct application of Lemma 21 yields that the number of such sets is at most $2^{\left(\bar{\rho}H(\frac{2\epsilon}{\bar{\rho}})+\rho H(\frac{2\epsilon}{\rho})+o(1)\right)n}$. ◀

We note that this lemma is not true for general graphs. An example of a graph that admits $2^{n/4}$ bad sets is given in the full version.

▶ **Lemma 23** (Expansion Lemma for Hamiltonian graphs). *Consider a set $S \subset V$ of size $\bar{\rho}n$ and parameters α, β. The probability that the lowest αn vertices in S have less than βn neighbors is at most*

$$2^{\left(-H(\frac{\alpha}{\bar{\rho}})+\alpha H(\frac{\delta}{\alpha})+(1-\alpha)H(\frac{\delta}{(1-\alpha)})+o(1)\right)\bar{\rho}n}.$$

Proof. Consider a set S of size $\bar{\rho}n$, and the first αn vertices in S in a random permutation. This set is just a random subset of S of size αn. The number of choices of such subset is

$$\binom{\bar{\rho}n}{\alpha n} = 2^{(H(\frac{\alpha}{\bar{\rho}})+o(1))\bar{\rho}n}.$$

Notice that we can apply Lemma 21 for with set S, even though S is just a subset of V, because the same proof applies only with respect to a subset of vertices in one side of a Hamiltonian graph. Therefore, the number of subsets of size αn of S with at most βn neighbors is at most

$$2^{(\alpha H(\frac{\delta}{\alpha})+(1-\alpha)H(\frac{\delta}{(1-\alpha)})+o(1))\bar{\rho}n}.$$

Combining the above, we get that the probability that a random set of αn vertices of S have at most βn neighbors is at most

$$2^{(-H(\frac{\alpha}{\bar{\rho}})+\alpha H(\frac{\delta}{\alpha})+(1-\alpha)H(\frac{\delta}{(1-\alpha)})+o(1))\bar{\rho}n}.$$ ◀

Now that we have established the three lemmas we are ready to prove Theorem 17.

Proof of Theorem 17. Setting $\epsilon = 0.0012$, $\alpha = 0.245$ and $\beta = 0.3675$ (and $\rho = \frac{1}{2} + \epsilon$, $\bar{\rho} = 1 - \rho$), we get that these parameters satisfy the conditions for Lemmas 20, 23 and 22.

Applying Lemma 22, we get that the size of B_ϵ is at most $n_B \leq 2^{0.044n}$. Applying Lemma 23, we get that the probability that the lowest αn vertices of a set of size $\bar{\rho}n$ have less than βn neighbors is at most $p \leq 2^{-0.86n}$. Applying Lemma 20, we get that the probability that for a set S of size $\bar{\rho}n$ the αnth vertex in a random π comes after βn vertices of $V - S$ is at most $q \leq 2^{-0.45n}$. Combining these three, we get that the probability there exists a set of size $\bar{\rho}n$ unmatched by a random π is at most $n_B(p+q) < 1$, therefore, there must be a π that matches at least one vertex in each set of size $\bar{\rho}n$, and the proof follows. ◀

This proof approach can be also used to show that a random permutation guarantees to match more than a half of the vertices in every regular graph. On the other hand, Theorem 24 in Section C shows that one cannot hope to get $\rho > 3/4$ with a random permutation in regular graphs.

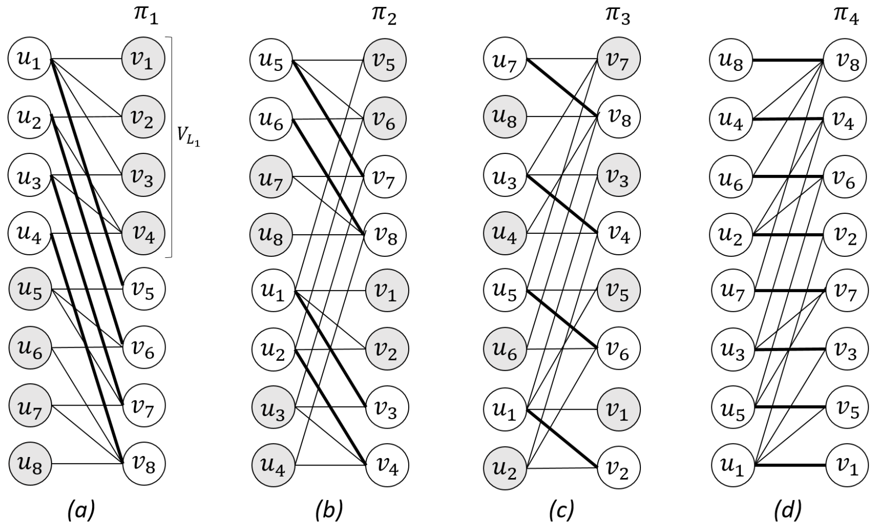

Figure 2 An iterative process where unmatched vertices are given priority. In every iteration thick edges are in the matching; gray vertices are unmatched.

B Iterative Process

A natural approach for establishing the existence of a good permutation π is the following iterative process of "upgrading" unmatched vertices. Given a permutation $\pi : V \to [n]$ and a permutation $\sigma : U \to [n]$, let $M[\pi, \sigma]$ be the result of the greedy matching where vertices in U arrive in order σ (from low to high) and each vertex $u \in U$ is matched to its lowest (under π) neighbor (or left unmatched if all its neighbors are already matched).

Fix an arbitrary permutation π_1 on V, and let σ_1 be a permutation on U minimizing the greedy matching[5]. Let $M_1 = M[\pi, \sigma]$ be the result of the greedy matching under permutations σ and π. If $|M_1|/n$ is some constant greater than $1/2$, then terminate with permutation π_1. Otherwise, partition V into the set V_{L_1} of unmatched vertices (L for *low*, as they will be placed low in the next iteration, and also for *losers*, or *leftovers*) and the set V_{H_1} of matched vertices (H for *high*, as they will be placed high in the next iteration, and also for *hitters*, or *happy*).

Consider now a permutation π_2 in which V_{L_1} precedes V_{H_1} (preserving the internal order between vertices in V_{L_1} and similarly between vertices in V_{H_1}), and let σ_2 be a permutation on U minimizing the resulting greedy matching. Let $M_2 = M[\pi_2, \sigma_2]$. If $|M_2|/n$ is some constant greater than $1/2$, then terminate with permutation π_2. Else, partition V into the set V_{L_2} of unmatched vertices and the set V_{H_2} of matched vertices, and consider a permutation π_3 in which V_{L_2} precedes V_{H_2} (preserving internal orders). Continue this iterative process until the obtained permutation π_k ensures a matching greater than a half.

The intuition behind this approach is that the unmatched vertices need some "help" in order to be matched, and we provide this help in the form of prioritizing them over their mates. One might hope that this process will reach a good permutation within a constant number of iterations. Unfortunately, we show an example where the process goes through $\log n$ iterations before it first obtains a permutation ensuring a matching that exceeds $n/2$.

[5] It is unclear whether σ_1 can be computed in polynomial time. The related problem of computing a minimum maximal matching in bipartite graphs is known to be NP-hard [2]. However, here we consider the existential problem.

The construction of the graph is inductive. The base is $G_0(U_0, V_0; E_0)$, with two vertices u, v and a single edge between them. For every $i = 1, 2, \ldots$, $G_i(U_i, V_i; E_i)$ is such that $|U_i| = |V_i| = 2^i$; it is obtained by taking two (disjoint) copies of G_{i-1}, with additional edges of the form (u_j, v_j) for every u_j from one copy of G_{i-1} to v_j in the second copy of G_{i-1}. An example of G_3 is presented in Figure 2(a). The iterative process is depicted in Figure 2(a)-(d). In all iterations preceding the last one, exactly $n/2$ vertices are matched in the worst σ.

C Additional Results

The following theorem shows that one cannot hope to get $\rho > 3/4$ with a random permutation in regular graphs.

▶ **Theorem 24.** *For every $\epsilon > 0$ and sufficiently large d, there are d-regular graphs G for which a random permutation π results in $\rho \leq \frac{3}{4} + \epsilon$.*

Proof. Consider a d-regular bipartite graph $G(U, V; E)$, where d is very large, there is a balanced bipartite independent set (S, T) of size $\frac{1-\epsilon}{2}n$, and conditioned on that, G is random. Let Q (a random variable) be the set of first $\frac{1+\epsilon}{2}n$ vertices under the random permutation π. Then, $E[|T \cap (V \setminus Q)|] = (\frac{1-\epsilon}{2})^2 n \simeq \frac{1}{4}n$. W.h.p. there will be a perfect matching between Q and $U \setminus S$. Hence one can choose a permutation σ over U that matches all of $U \setminus S$ to Q. But then the vertices $T \cap (V \setminus Q)$ will remain unmatched. ◀

We also establish a few impossibility results for regular graphs of low degree.

▶ **Theorem 25.** *The following hold:*
- *There exists a 3-regular bipartite graph G for which $\rho(G) = \frac{5}{7}$.*
- *There exists a 4-regular bipartite graph G for which $\rho(G) = \frac{10}{13}$.*

The proof relies on graphs induced by projective planes. A projective plane consists of a set of lines and a set of points, where (among other properties) every two lines intersect in a single point and every two points are incident to a single line. A projective plane induces a bipartite graph $G(U, V; E)$, where every vertex $u \in U$ corresponds to a point in the plane, every vertex $v \in V$ corresponds to a line, and there exists an edge between u and v if the point corresponding to u is incident to the line corresponding to v.

Proof. For the first result, we show that $\rho = \frac{5}{7}$ for the bipartite graph induced by the Fano plane. The Fano plane is a projective plane consisting of 7 points and 7 lines, with 3 points on every line and 3 lines through every point. Consider the 3-regular bipartite graph $G(U, V; E)$ induced by the Fano plane. Let $N(V')$ denote the neighbors of a set $V' \in V$. For every set $V' \in V$ such that $|V'| = 2$, it holds that $|N(V')| = 5$. We show below that for every such V' there exists a perfect matching between $N(V')$ and $V \setminus V'$. Hence one can choose a permutation σ over U whose first 5 vertices are $N(V')$ that will match the vertices of $V \setminus V'$ one by one. Thereafter, the vertices of V' will remain unmatched. By Hall's condition, it suffices to show that for every set $U' \subset N(V')$ such that $|U'| \leq 5$ it holds that $|N(U')| \geq |U'| + 2$ (so that Hall's condition applies with respect to the set $V \setminus V'$). Indeed, for every set U' of size 1, $|N(U')| = 3$, for every set U' of size ≥ 2, $|N(U')| \geq 6$, and for every set U' of size 5, $|N(U')| = 7$. It follows that $\rho(G) = 5/7$.

The second result follows a similar argument. It is known that there exists a projective plane consisting of 13 points and 13 lines, with 4 points on every line and 4 lines through every point. We claim that $\rho = \frac{10}{13}$ for the bipartite graph $G(U, V; E)$ induced by this projective plane. By the properties of a projective plane, for every set $V' \in V$ such that $|V'| = 3$, it holds that $|N(V')| \in \{9, 10\}$. We show below that for every such V' there exists a perfect matching between $N(V')$ (and possibly an additional vertex u in case $|N(V')| = 9$) and $V \setminus V'$. Hence one can choose a permutation σ over U whose first 10 vertices are $N(V')$ (possibly with the additional vertex) that will match the vertices of $V \setminus V'$ one by one. Thereafter, the vertices of V' will remain unmatched. By Hall's condition, it suffices to show that for every set $U' \subset N(V')$ such that $|U'| \leq 10$ it holds that $|N(U')| \geq |U'| + 3$ (so that Hall's condition applies with respect to the set $V \setminus V'$). Indeed, for every set U' of size 1, $|N(U')| = 4$, for every set U' of size ≥ 2, $|N(U')| \geq 7$, for every set U' of size ≥ 5, $|N(U')| \geq 11$, and for every set U' of size ≥ 9, $|N(U')| = 13$, It follows that $\rho(G) = 10/13$. ◀