Almost Envy-Free Repeated Matching in Two-Sided Markets

Sreenivas Gollapudi[1], Kostas Kollias[1], and Benjamin Plaut[2](✉)

[1] Google Research, Mountain View, USA
sgollapu@google.com, kostaskollias@google.com
[2] Stanford University, Stanford, USA
bplaut@stanford.edu

Abstract. A two-sided market consists of two sets of agents, each of whom have preferences over the other (Airbnb, Upwork, Lyft, Uber, etc.). We propose and analyze a repeated matching problem, where some set of matches occur on each time step, and our goal is to ensure fairness with respect to the cumulative allocations over an infinite time horizon. Our main result is a polynomial-time algorithm for additive, symmetric ($v_i(j) = v_j(i)$), and binary ($v_i(j) \in \{a, 1\}$) valuations that both (1) guarantees *envy-freeness up to a single match* (EF1) and (2) selects a maximum weight matching on each time step. Thus for this class of valuations, fairness can be achieved without sacrificing economic efficiency. This result holds even for *dynamic valuations*, i.e., valuations that change over time. Although symmetry is a strong assumption, we show that this result cannot be extended to asymmetric binary valuations: (1) and (2) together are impossible even when valuations do not change over time, and for dynamic valuations, even (1) alone is impossible. To our knowledge, this is the first analysis of envy-freeness in a repeated matching setting.

1 Introduction

Recent years have seen a dramatic increase in electronic marketplaces, both in quantity and scale. Many of these are *two-sided* markets, meaning that the market makes matches between two sets of agents (homeowners and guests for Airbnb, employers and workers for Upwork, drivers and riders for Lyft and Uber, etc.), each of whom has preferences over the other. This is in contrast to traditional resource allocation (cake cutting, Fisher markets, auctions, etc.) where only one side of the market has preferences. Although envy-freeness and relaxations thereof have been studied extensively in one-sided resource allocation (this research area is typically referred to as "fair division"), we are aware of just one paper considering envy-freeness for two-sided preferences [22].[1]

There are two primary motivations for our work. The first is to simply study fair division for two-sided preferences. The second is that in some ways, two-sided electronic marketplaces like Airbnb, Upwork, Lyft, and Uber are actually

[1] Although [22] is conceptually similar, it is in different setting of *recommendation algorithms*, and so it is technically quite different.

X. Chen et al. (Eds.): WINE 2020, LNCS 12495, pp. 3–16, 2020.
https://doi.org/10.1007/978-3-030-64946-3_1

in a better position to impose fairness than one-sided marketplaces. The reason is that most one-sided markets are *decentralized*, in the sense that a seller offers different goods at different prices, buyers peruse the wares at their leisure, and make individual decisions about what they wish to purchase. On the contrary, most two-sided markets operate by way of matches mediated by a centralized platform, giving the platform the ability to affect the outcomes of the system. Indeed, on Lyft and Uber, an automated central authority has almost complete control over the matches, giving the algorithm tremendous power over the outcomes for each individual agent. The power dynamic between the platform and the participating agents makes it even more important to ensure that the matching algorithms are fair to each agent.

1.1 Repeated Matching

A crucial element of marketplaces is *repeated* matching. Agents do not receive all of their matches at once; typically, an agent can only process a few matches at a time (a driver can only fit so many riders in the car, a worker can only handle so many contracts at once). Only after an agent completes some of her current matches can she be given new matches. This motivates a model where on each time step, an irrevocable matching decision must be made, and we expect fairness with respect to the cumulative matching at each time step. We consider an infinite time horizon and a finite set of agents, so we must allow the same pair of agents to be matched multiple times; thus each agent's cumulative set of matches will be a multiset.

A vital aspect of any repeated setting is that preferences can change. In some cases, preferences may change in direct response to the matches an agent receives: if a driver is matched with a rider who wishes to go to location X, once that ride is completed, the driver will prefer riders whose pickups are close to X. In other cases, agents may desire variety among matches: Airbnb guests may not wish to vacation in the same area every time. Additionally, preferences may simply drift over time. We refer to valuations that change over time as *dynamic* valuations.

1.2 Fairness Notions

For one-sided fair division with indivisible items, full envy-freeness is impossible: for two agents and a single item, one must receive the item and the other agent will be envious. The same issue applies in our setting: if every agent on one side of the market is interested in the same agent on the other side, no algorithm can guarantee envy-freeness.

One solution is to consider relaxations of envy-freeness, such as *envy-freeness up to one good* (EF1). An outcome is EF1 if whenever agent i envies agent j, there exists a good in j's bundle such that i would not envy j after removing that good.[2] This property has been studied widely for one-sided preferences, but

[2] Note that this good is not actually removed: this is simply a thought experiment used in the definition of EF1.

to our knowledge has not been considered for two-sided markets. We can define EF1 equivalently for two-sided preferences: simply replace "there exists a good" with "there exists a match", etc.

The less obvious question is how to adapt EF1 to the repeated setting. In this paper, we assume time is divided into discrete steps, where on each step, some set of matches occur. Each match consists of two agents, one from each side of the market. We would like the cumulative matching after each time step to be EF1. We will also assume that each side of the market has the same number of agents (if not, add "dummy" agents to the smaller side).

We consider two different versions of this model. In the first version, each time step consists of just a single match, so we are effectively requiring the cumulative matching to be EF1 at every point in time. We call this *EF1-over-time*.

However, asking for fairness at every point in time may be too strong. Furthermore, in most real-world applications, matches would be happening in parallel anyway. Conversely, EF1-over-time poses no restrictions on how many matches agents receive. In real life, agents often have similar "capacities" (e.g., most cars have a similar number of seats) and thus should arguably receive matches at similar rates. These concerns motivate a second version of the model, where on each time step, each agent is matched exactly once (i.e., we select a perfect matching). Thus each time step represents a "round" of matches (which may happen in parallel). We still require that the cumulative matching is EF1 after time step, and we call this *EF1-over-rounds*.

1.3 Our Results

We use $v_i(j)$ to denote agent i's value for agent j, and assume valuations are additive. We say that valuations are *symmetric* if $v_i(j) = v_j(i)$ for all agents i, j, and *binary* if there exists $a \in [0, 1)$ such that $v_i(j) \in \{a, 1\}$ for all i, j.[3] It is worth noting that for symmetric valuations, negative values for $v_i(j)$ are subsumed in the following sense: if the algorithm ever says to match agents i and j where $v_i(j) < 0$, we simply ignore this and never make the match, which gives both agents value 0 for the "match".[4]

We now describe our results. Due to space constraints, all proofs are deferred to the full version of the paper [16].

EF1-Over-Rounds for Dynamic Symmetric, and Binary Valuations. Our main result is that for dynamic, symmetric, and binary valuations, we give an algorithm which both satisfies EF1-over-rounds, and selects a maximum weight matching on each time step (Theorem 3.1) This holds even when valuations are dynamic.[5] This

[3] If we wish to add dummy agents to one side of the market, the most natural case would be $a = 0$, in order to express that dummy agents have no value.

[4] In order for this argument to be complete when considering EF1-over-rounds, this "match" must still count as part of the perfect matching for that time step.

[5] We allow valuations to change arbitrarily between time steps. Furthermore, our algorithm does not need to know how valuations will change in response to a given match.

shows that for this class of valuations, fairness can be achieved without sacrificing economic efficiency. Our algorithm runs in time $O(n^{2.5})$ per time step.

The class of symmetric and binary valuations is somewhat restricted, but is important to keep several things in mind. First, it is often hard to elicit more complex valuations. Agents can easily answer binary questions such as "Would you be happy with this match?", but may not be able to provide a real number value for potential matches. Second, the best interpretation of our result (in our opinion) is that agents' preferences are not truly binary, but that our algorithm is guaranteeing EF1 with respect to a *binary projection of the preferences.* That is, ask each agent to label each possible match as "good" or "bad", and guarantee EF1 with respect to those preferences. This interpretation is reinforced by the fact that the cumulative matchings computed by our algorithm will be EF1 uniformly across all possible values a, and agents can even have different values of a. See Sect. 3.1 for a discussion of this.

Symmetry, however, is a significant assumption on the agents' preferences. There are reasons to believe real-world preferences are largely symmetric: a rider is likely to prefer a driver closer to her, and vice versa. However, a natural question is whether this assumption is necessary.

Counterexamples. Our next set of results shows that the symmetry assumption is in fact necessary. First, we show that for dynamic binary valuations, EF1-over-rounds alone is impossible (Theorem 4.1). Second, for non-dynamic (i.e., valuations do not change over time) binary valuations, it is impossible to satisfy EF1-over-rounds while guaranteeing a maximum weight matching for each time step (Theorem 4.2). These impossibility results suggest that EF1-over-rounds may be too much to ask for in the setting of two-sided repeated matching. However, our counterexamples do not rule out the possibility of EF1-over-time, even for general additive valuations. We leave this as our primary open question.

Beyond Symmetric Valuations. Despite this negative result, we show that it is possible to relax the symmetry assumption, at least in the context of EF1-over-time. We show that for $\{0, 1\}$ binary valuations[6] with an assumption that we call "only symmetric cycles", EF1-over-time can be guaranteed. We formally define "only symmetric cycles" in the full version of the paper [16], but this assumption is strictly weaker than full symmetry of valuations.

Beyond Binary Valuations. In a similar vein, we show that when one side of the market has two agents, the binary assumption can also be relaxed. Specifically, we give an algorithm which is EF1-over-time for any additive valuations.

1.4 Related Work

There are two primary bodies of related work: (one-sided) fair division, and matching markets.

[6] For this result, we assume that $v_i(j) \in \{0, 1\}$ for each agent, as opposed $v_i(j) \in \{a, 1\}$ for any $a \in [0, 1)$.

Fair Division. Fair division has a long history. In fact, the Bible documents Abraham and Lot's use of the cut-and-choose protocol to fairly divide land. The formal study of fair division was started by [27] in 1948, and envy-freeness was proposed in 1958 [15] and further developed by [13]. A full overview of the fair division literature is outside the scope of this paper (we refer the interested reader to [5, 21]), and we discuss only the work most relevant to our own.

There are two main differences between our work and that of traditional fair division. First, we study two-sided preferences instead of one-sided preferences. Second, we study a repeated setting, where we must make an irrevocable decision on each time step; most fair division research considers a "one-shot" model where all the goods are allocated at once.

We briefly overview some key results in the one-sided one-shot model of fair division for indivisible items[7]. The EF1 property was proposed for this model by [6]. EF1 allocations always exist, and can be computed in polynomial time, even for general combinatorial valuations [20][8]. This sweeping positive result for the one-sided one-shot model lies in stark contrast to our negative results for the two-sided repeated model. It was later shown that for additive valuations, maximizing the product of valuations yields an allocation that is both EF1 and Pareto optimal [8].

Envy-Freeness up to Any Good (EFX). The same paper suggested a new fairness notion that is strictly stronger than EF1, which they called EFX[9]. The first formal results regarding EFX allocations were given by [23]. A major breakthrough recently proved the existence of EFX allocations for additive valuations and three agents [9], but despite ongoing effort, the question of existence remains unsolved for more than three agents (or more complex valuations). This is perhaps the most significant open problem in the fair division of indivisible items.

In many contexts (especially for additive valuations), it is common to modify the requirement to be that whenever i envies j, removing any good which i values positively from j's bundle is sufficient to eliminate the envy [8].[10] Under the latter definition, for $\{0, 1\}$ binary valuations, EF1 and EFX coincide. This is because the only positively valued goods are the maximum value goods. In this sense, our positive results for $\{0, 1\}$ binary valuations immediately extend to EFX as well.

Repeated Fair Division. There are a smattering of recent works studying envy-freeness for one-sided preferences in a repeated (i.e., not one-shot) setting; see [1]

[7] *Indivisible* items, such as cars, must each go entirely to a single agent. In contrast, *divisible* items, such as a cakes, can be split between multiple players. Fair division studies both of these settings, but the indivisible case is more relevant to our work.

[8] The algorithm of [20] was originally developed with a different property in mind.

[9] An allocation is *envy-free up to any good* (EFX) if whenever i envies j, removing *any* good from j's bundle eliminates the envy.

[10] The reason this is less common when considering non-additive valuations is that for general combinatorial valuations, it is less clear what "values positively" means.

for a short survey. One example is [2], which focuses on minimizing the maximum envy (i.e., the maximum difference between an agent's value for her own bundle and her value for another agent's bundle) at each time step. Despite the growing interest in repeated one-sided fair allocation, the literature on the analogous two-sided problem remains sparse.

Matching Markets. The other relevant field is (bipartite) matching markets[11]. In a *one-to-one* matching market, each agent receives exactly one match. Perhaps the most famous result for one-to-one matching is that of Gale and Shapley, whose algorithm finds a stable matching [14]. More relevant to us is the model of *many-to-many matching markets*, where each agent can receive multiple matches; stability has often been the primary criterion in this model as well [11,19,25,26]. There is also some work on stability in dynamic matching markets [10,18].

In contrast, fairness in many-to-many matching has received considerably less attention. In fact, Gale and Shapley's algorithm for one-to-one matching is known to compute the stable matching which is the worst possible for one side of the market, and best possible for the other.

Fair Ride-Hailing. There is a growing body of work surrounding the ethics of crowdsourced two-sided markets, especially ride-hailing (e.g., Lyft and Uber) [3,4,7,12,17]. We are aware of just two works studying fairness for two-sided markets from an algorithmic perspective: [29] and [28], both of which focus on ride-hailing. The former paper considers ride-*sharing*, where multiple passengers are matched with a single driver. The authors focus on fairness with respect to the savings achieved by each passenger. This paper is primarily theoretical, like ours, but is specific to ride-hailing, unlike ours. The latter paper studies a fairness notion based on the idea that "spread over time, all drivers should receive benefits proportional to the amount of time they are active in the platform". The model considered in this paper is more general than just ride-hailing, however the paper is primarily experimental, and the experiments are in the ride-hailing setting.

Consequently, we are not aware of any prior work studying algorithms with provable fairness guarantees for repeated two-sided matching markets. In this way, our work can be viewed as simultaneously building on the fair division literature (by considering two-sided preferences) and building on the matching market literature (by studying envy-freeness for repeated two-sided markets).

The paper proceeds as follows. Section 2 describes the formal model. Section 3 presents our main result: an algorithm for symmetric and binary valuations such that (1) the sequence of cumulative matchings is EF1-over-rounds, (2) a maximum weight matching is chosen for each time step, and (3) this holds even for dynamic valuations (Theorem 3.1). Section 4 presents our counterexamples: Theorem 3.1 cannot be extended to non-symmetric binary valuations. Specifically, without symmetry, (1) and (3) together and (1) and (2) together are both

[11] For a broad overview of this topic, see [24].

impossible (Theorems 4.1 and 4.2, respectively). The rest of the results (and all of the proofs) can be found in the full version of the paper [16].

2 Model

Let N and M be two sets of agents. We assume that $|N| = |M| = n$; if this is not the case, we can add "dummy" agents (i.e., agents i such that $v_i(j) = v_j(i) = 0$ for all j) to the smaller side of the market until both sides have the same number of agents. We will typically use odd numbers for the elements of N and even numbers for the elements of M, i.e., $N = \{1, 3, \dots, 2n-1\}$ and $M = \{2, 4, \dots, 2n\}$. A *matching* X assigns a multiset of agents in N to each agent in M, and a multiset of agents in M to each agent in N. For each $i \in N \cup M$, we will use X_i to denote agent i's *bundle*, i.e., the multiset of agents she is matched to. Throughout the paper, we will use standard set notation for operations on the multisets X_i. For example, $X_i \cup \{j\}$ increments the *multiplicity* of j in X_i, i.e., the number of times j occurs in X_i. In order for X to be a valid matching, the multiplicity of j in X_i must be equal to the multiplicity of i in X_j for each $i \in N$, $j \in M$.

Each agent i also has a *valuation function* v_i, which assigns a real number to each possible bundle she might receive. We will use $\mathbf{v}$ to denote the *valuation profile* which assigns valuation v_i to agent i. We say that v_i is *additive* if for any bundle X_i,

$$v_i(X) = \sum_{j \in X} v_i(\{j\})$$

Since X is a multiset, the sum over $j \in X$ includes each j a number of times equal to its multiplicity. For example, if $X = \{j, j\}$ and $v_i(\{j\}) = 1$, then $v_i(X) = 2$. With slight abuse of notation, we will write $v_i(\{j\}) = v_i(j)$. We say that a valuation v_i is *binary* if there exists $a \in [0, 1)$ such that $v_i(j) \in \{a, 1\}$ for all $i, j \in N$ or $i, j \in M$. We say that a valuation profile $\mathbf{v}$ is *symmetric* if $v_i(j) = v_j(i)$ for all $i \in N, j \in M$.

We say that i *envies* j under X if $v_i(X_i) < v_i(X_j)$. We only consider envy within the same side of the market: it is unclear what it would mean for some $i \in N$ to envy $j \in M$. We can express this by setting $v_i(j) = 0$ for $i, j \in N$ or $i, j \in M$.

Definition 2.1. *A matching X is* envy-free up to one match *(EF1) if whenever i envies j, there exists $k \in X_j$ such that $v_i(X_i) \geq v_i(X_j \setminus \{k\})$.*

Note that the set subtraction $X_j \setminus \{k\}$ decreases the multiplicity of k by 1; it does not remove k altogether. Also, we will say that X is EF1 with respect to N (resp., M) if the above holds for every pair $i, j \in N$ (resp., $i, j \in M$).

One important tool we will use is the *envy graph*:

Definition 2.2. *The* envy graph *of a matching X is a graph with a vertex for each agent, and a directed edge from agent i to agent j if i envies j under X.*

We will especially be interested in cycles in the envy graph, and will use the terms "cycle in the envy graph" and "envy cycle" interchangeably.

2.1 Repeated Matching

We consider a repeated setting, where on each time step t, some set of matches occur. Each "match" (alternatively, *pairing*) consists of one agent in N and one agent in M.

Let x^t denote the set of matches which occur at time t. Each agent will receive at most one match per time step. If agent i is matched to an agent j at time t, let $x_i^t = \{j\}$; otherwise, let $x_i^t = \emptyset$. For an infinite sequence $x^1, x^2, x^3 \dots$, let X^t denote the cumulative matching up to and including time t. Formally, for each $i \in N \cup M$,

$$X_i^t = \begin{cases} \emptyset & \text{if } t = 0 \\ X_i^{t-1} \cup x_i^t & \text{if } t > 0 \end{cases}$$

In words, X_i^t is the set of matches i has received up to and including time t.

Our main result holds even when valuations are allowed to vary over time. Specifically, a *dynamic* valuation v_i will have a value $v_i^t(j)$ for each agent j on each time step t (as before, we write $v_i^t(j) = 0$ for $i, j \in N$ or $i, j \in M$). A profile of dynamic valuations is symmetric if $v_i^t(j) = v_j^t(i)$ for all i, j, t. For a pair of agents i, j (with $i = j$ allowed), $v_i(X_j^t)$ is given by

$$v_i(X_j^t) = \sum_{t'=1}^{t} v_i^{t'}(x_j^{t'})$$

where $v_i^{t'}(\emptyset) = 0$. That is, i's value for a bundle X_j is as if i had received exactly those matches at exactly those times. It is important for this definition to include both $i = j$ and $i \neq j$, so that we can evaluate envy between agents.

We make no assumptions on how valuations change between time steps: they can even change adversarially, since our algorithm will not use any knowledge about future valuations when making matching decisions.

We consider two definitions of EF1 in the repeated matching setting. In both cases, we require the cumulative matching at the end of each time step to be EF1. The difference is that for EF1-over-time, each time step consists a single match, and for EF1-over-rounds, each time step consists of a "round" of matches where all agents receive exactly one match (i.e., a perfect matching between N and M).

Definition 2.3. *The sequence $\mathcal{X} = X^0, X^1, X^2 \dots$ is* EF1-over-time *if for all $t \geq 0$, each x^t contains a single match, and X^t is EF1.*

Definition 2.4. *The sequence $\mathcal{X} = X^0, X^1, X^2 \dots$ is* EF1-over-rounds *if for all $t \geq 0$, x^t is a perfect matching, and X^t is EF1.*

Formally, these notions are incomparable: EF1-over-time has a stronger fairness requirement (the cumulative matching should be EF1 after every match, not just after every round of matches), but does not require agents to receive the same number of matches. However, EF1-over-rounds does imply EF2-over-time (where we may remove two matches in order to eliminate the envy): expand each "round" into n time steps, each containing one match, in an arbitrary order. We know that at the end of each round of n time steps, the cumulative matching is EF1. Within each round, each agent only gains one additional match, and we can always remove that match to return to an EF1 state.

Our goal will be to show the existence of (and efficiently compute) a sequence $x^1, x^2, x^3 \dots$ such that the induced sequence $\mathcal{X}$ is EF1-over-time and/or EF1-over-rounds. For brevity, if an algorithm is guaranteed to produce a sequence $\mathcal{X}$ that is EF1-over-time (resp., EF1-over-rounds), we simply say that the algorithm is EF1-over-time (resp., EF1-over-rounds).

3 EF1 for Dynamic, Symmetric, and Binary Valuations

In this section, we consider binary and symmetric valuations that may change over time. For this class of valuations, we give a polynomial-time algorithm that produces a sequence which is EF1-over-rounds, and chooses a maximum weight matching for each time step. This leads to the following theorem:

Theorem 3.1. *For dynamic, binary, and symmetric valuations, Algorithm 1 is EF1-over-rounds, and the matching x^t for each time step t is a maximum weight matching (with respect to the valuations on that time step). Furthermore, the algorithm runs in time $O(n^{2.5})$ per time step.*

3.1 Algorithm Setup

Before we discuss the algorithm, we need the following definition, which will imply EF1 (Lemma 3.2):

Definition 3.1. *We say that a pair of agents (i, j) is* c-envy-bounded *if $v_i(X_j) - v_i(X_i) \leq c$, and we say a matching X is c-envy-bounded if every pair (i, j) is c-envy-bounded.*

A quick note: recall that our goal is to choose a sequence of pairings $x^1, x^2 \dots$, and that these pairings fully specify the sequence of cumulative matchings $\mathcal{X}$. Consequently, when giving pseudocode for our algorithms (throughout the paper), we do not explicitly update $\mathcal{X}$: we assume that whenever some x^t is changed, every $X^{t'}$ for $t' \geq t$ is automatically updated. We feel that this leads to more concise and intuitive pseudocode.

Algorithm 1 is very simple. For each time step t, we initialize x^t to be an arbitrary maximum weight matching for the current valuations, and make changes to this matching until we are satisfied. Specifically, while there exist agents i, j such that (i, j) is not $(1-a)$-envy-bounded in the cumulative matching, we swap their matches in x^t. When no such pair of agents exists, we exit the while loop and confirm the matches. Throughout all of our algorithms, we will use the function `MakeMatch` to indicate that we are confirming the matches in x^t.

It is important to note that the algorithm is *not* going back in time and changing pairings already made: once a pairing is confirmed with `MakeMatch`, it

Algorithm 1. An EF1-over-rounds algorithm for agents with dynamic, symmetric, and binary valuations.

```
function EF1Matching($N, M, \mathbf{v}$)
    for each $t \in \mathbb{N}_{\geq 0}$ do
        $\{x^t\} \leftarrow$ MaxWeightMatching($N, M, \mathbf{v}$)
        while $\exists$ agents $i, j$ s.t. $v_i(X_j^t) - v_i(X_i^t) > 1 - a$ do
            $(x_i^t, x_j^t) \leftarrow (x_j^t, x_i^t)$
        MakeMatch($x^t$)
```

is never changed. The algorithm starts with a tentative matching, and changes *tentative* matches until it is satisfied for the current time step (see Fig. 1), at which point the matches are confirmed with `MakeMatch`. The algorithm then proceeds to the next time step and never changes pairings from previous time steps. Note also that the algorithm uses no information about valuations for future time steps.

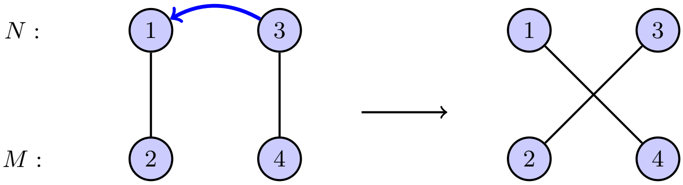

Fig. 1. A hypothetical swap performed by Algorithm 1. On the left we see a tentative perfect matching: $(\{1, 2\}, \{3, 4\})$. The blue arrow indicates that if this matching were to be chosen, the pair $(3, 1)$ would not be $(1 - a)$-envy-bounded. Thus agents 1 and 3 swap their (tentative) matches, and the new tentative matching is $(\{1, 4\}, \{3, 2\})$. The matching $(\{1, 2\}, \{3, 4\})$ is never confirmed by `MakeMatch`: it is merely a stepping stone in the process of computing the eventual matches to be chosen for this time step. For the case of more than four agents, this process could repeat (although not indefinitely; see the full version of the paper for this proof [16]. (Color figure online)

Our central correctness lemma will be the following:

Lemma 3.1. *Let $t \geq 1$ be any time step, and suppose that X^{t-1} is $(1 - a)$-envy-bounded and has no envy cycles. Then X^t is $(1 - a)$-envy-bounded and has no envy cycles. Furthermore, the chosen matching x^t is a maximum weight matching (with respect to the valuations on that time step).*

Before diving into the proof of Lemma 3.1 (and the runtime analysis), we briefly show that $(1 - a)$-envy-boundedness will actually give us the result we want:

Lemma 3.2. *Suppose valuations are binary, and suppose X is $(1 - a)$-envy-bounded. Then X is EF1.*

Proof. Suppose i envies j under X^t. If $v_i(X_j^t) = a|X_j^t|$, then $v_i(X_i^t) \geq v_i(X_j^t)$, which contradicts i envying j. Thus $v_i(X_j^t) \geq 1 + a(|X_j^t| - 1)$. Thus there exists $k \in X_j^t$ such that $v_i(X_j^t \setminus \{k\}) = v_i(X_j^t) - 1$. Therefore $v_i(X_i) - v_i(X_j^t \setminus \{k\}) \geq 1 + (a-1) = a \geq 0$, which proves the claim.

The role of a. Before diving into the main proof, we briefly discuss the role of a. For Theorem 3.1, we assume that there exists $a \in [0,1)$ such that $v_i(j) \in \{a, 1\}$ for all $i, j \in N$ or $i, j \in M$. For ease of notation, we assume that all agents have the same value of a, but this is in fact not necessary. In fact, Algorithm 1 will be EF1-over-rounds simultaneously for all values of a.

Lemma 3.3. *Assume (i,j) is $(1-a)$-envy-bounded and that $|X_i^t| = |X_j^t|$. Let $a' \in [0,1)$, and define a new valuation v_i' such that $v_i'(k) = a'$ whenever $v_i(k) = a$, and $v_i'(k) = 1$ otherwise. Then (i,j) is $(1-a')$-envy-bounded with respect to v_i'.*

Note that the assumption of $|X_i^t| = |X_j^t|$ is always satisfied when working with EF1-over-rounds, since we will match every agent once on each time step. Therefore we can actually just choose an arbitrary value of $a \in [0,1)$ and run Algorithm 1. Lemma 3.3 implies that the resulting sequence of matchings will be EF1-over-rounds simultaneously for all values of a, even if different agents have different values of a. That said, if we need to include dummy agents in order to equalize the sizes of N and M, $a = 0$ probably makes the most sense.

4 Counterexamples

A natural question is whether Theorem 3.1 can be extended to all dynamic binary valuations (i.e., not necessarily symmetric). The answer is unfortunately no, which we show in two different ways. First, for dynamic binary valuations, EF1-over-rounds alone is impossible (Theorem 4.1). Second, for non-dynamic binary valuations, it is impossible to guarantee both EF1-over-rounds and maximum weight matching for each time step (Theorem 4.2).

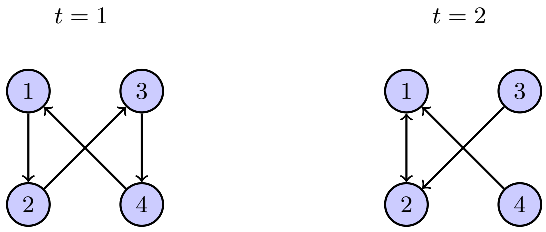

Fig. 2. An instance with dynamic and binary valuations where EF1-over-rounds is impossible.

Theorem 4.1 uses the instance in Fig. 2. Essentially, after the first round, either agents 1 and 3 form an envy cycle, or agents 2 and 4 form an envy cycle.

After the second round of matching, one of the agents in the envy cycle will become even more envious, violating EF1.

Theorem 4.1. *For dynamic and binary valuations, there is no algorithm which is EF1-over-rounds.*

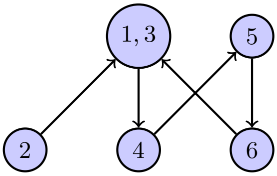

Fig. 3. An instance with binary valuations where guaranteeing both EF1-over-rounds and maximum weight matching is impossible.

Theorem 4.2 uses the instance in Fig. 3. For some intuition, note that are two cycles of desire: $(1, 4, 5, 6)$ and $(3, 4, 5, 6)$. Like in the previous counterexample, these cycles will cause problems, but here we have the additional consideration that agents 1 and 3 are competing for agent 4. We show that the frequency with which agents 4 and 5 are matched is at least the frequency with which *either* agent 1 or agent 3 is matched with agent 4. For example, if agents 1 and 3 have each been matched to agent 4 twice, then agents 4 and 5 will have been matched 4 times. This leads to agents 1 and 3 increasingly envying agent 5, until EF1 is violated.

The assumption of maximum weight is necessary only to prevent agents 2 and 5 from ever being matched: if agents 2 and 5 can be matched, the above argument can be circumvented.

Theorem 4.2. *For binary valuations, there is no algorithm which is EF1-over-rounds and also chooses a maximum weight matching for each time step, even for non-dynamic valuations.*

5 Conclusion

In this paper, we proposed a model of envy-freeness for repeated two-sided matching. For binary and symmetric valuations, we gave an algorithm that (1) satisfies EF1-over-rounds, (2) chooses a maximum weight matching for each time step, and (3) works even for dynamic valuations (Sect. 3). Furthermore, without symmetry, (1) + (2) together and (1) + (3) together are each impossible. All proofs can be found in the full version of the paper, along with several additional results [16].

Our negative results for even binary valuations suggest that EF1-over-rounds may be too much to ask for. However, our results do not rule out the possibility of EF1-over-time, even for general additive valuations. More broadly, future work could investigate other possible fairness notions for this setting.

Another possible future direction concerns more general study of two-sided preferences. Envy-freeness is an example of a topic that has been widely studied for one-sided resource allocation, but not for two-sided markets. We wonder if there are other such topics that are worthy of study for two-sided preferences.

References

1. Aleksandrov, M., Walsh, T.: Online Fair Division: A Survey. arXiv preprint arXiv:1911.09488 (2019)
2. Benade, G., Kazachkov, A.M., Procaccia, A.D., Psomas, C.A.: How to make envy vanish over time. In: Proceedings of the 2018 ACM Conference on Economics and Computation, EC 2018, pp. 593–610. ACM, New York (2018)
3. Bokányi, E., Hannák, A.: Evaluating Algorithm Fairness in an Agent-based Taxi Simulation (2018, working paper)
4. Bokányi, E., Hannák, A.: Ride-share Matching Algorithms Generate Income Inequality (2019, working paper)
5. Brandt, F., Conitzer, V., Endriss, U., Lang, J., Procaccia, A.D.: Handbook of Computational Social Choice. Cambridge University Press, Cambridge (2016)
6. Budish, E.: The combinatorial assignment problem: approximate competitive equilibrium from equal incomes. J. Polit. Econ. **119**(6), 1061–1103 (2011)
7. Calo, R., Rosenblat, A.: The taking economy: uber, information, and power. Columbia Law Rev. (2017)
8. Caragiannis, I., Kurokawa, D., Moulin, H., Procaccia, A.D., Shah, N., Wang, J.: The unreasonable fairness of maximum Nash welfare. In: Proceedings of the 2016 ACM Conference on Economics and Computation, pp. 305–322. ACM (2016)
9. Chaudhury, B.R., Garg, J., Mehlhorn, K.: EFX exists for three agents. arXiv preprint arXiv:2002.05119 (2020)
10. Damiano, E., Lam, R.: Stability in dynamic matching markets. Games Econ. Behav. **52**, 34–53 (2005)
11. Echenique, F., Oviedo, J.: A theory of stability in many-to-many matching markets. Theor. Econ. **1**(2), 233–273 (2006)
12. Fieseler, C., Bucher, E., Hoffmann, C.P.: Unfairness by design? The perceived fairness of digital labor on crowdworking platforms. J. Bus. Ethics **156**(4), 987–1005 (2019)
13. Foley, D.K.: Resource allocation and the public sector. Yale Econ. Essays **7**(1), 45–98 (1967)
14. Gale, D., Shapley, L.S.: College admissions and the stability of marriage. Am. Math. Mon. **69**(1), 9–15 (1962)
15. Gamow, G., Stern, M.: Puzzle-math. Viking Adult (1958). http://www.worldcat.org/isbn/0670583359
16. Gollapudi, S., Kollias, K., Plaut, B.: Almost Envy-free Repeated Matching in Two-sided Markets. arXiv preprint arXiv:2009.09336 (2020)
17. Hannák, A., Wagner, C., Garcia, D., Mislove, A., Strohmaier, M., Wilson, C.: Bias in online freelance marketplaces: evidence from Taskrabbit and Fiverr. In: Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing, CSCW 2017, pp. 1914–1933. ACM, New York (2017)

18. Haruvy, E., Ünver, M.U.: Equilibrium selection and the role of information in repeated matching markets. Econ. Lett. **94**(2), 284–289 (2007)
19. Konishi, H., Ünver, M.U.: Credible group stability in many-to-many matching problems. J. Econ. Theor. **129**(1), 57–80 (2006)
20. Lipton, R.J., Markakis, E., Mossel, E., Saberi, A.: On approximately fair allocations of indivisible goods. In: Proceedings of the 5th ACM conference on Electronic commerce, pp. 125–131. ACM (2004)
21. Moulin, H.: Fair Division and Collective Welfare. MIT Press, Cambridge (2004)
22. Patro, G.K., Biswas, A., Ganguly, N., Gummadi, K.P., Chakraborty, A.: FairRec: two-sided fairness for personalized recommendations in two-sided platforms. Proc. Web Conf. **2020**, 1194–1204 (2020)
23. Plaut, B., Roughgarden, T.: Almost envy-freeness with general valuations. In: Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 2584–2603. Society for Industrial and Applied Mathematics (2018)
24. Roth, A.E., Sotomayor, M.: Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis. Cambridge University Press, Cambridge (1990)
25. Sotomayor, M.: Three remarks on the many-to-many stable matching problem. Math. Soc. Sci. **38**(1), 55–70 (1999)
26. Sotomayor, M.: Implementation in the many-to-many matching market. Games Econ. Behav. **46**(1), 199–212 (2004)
27. Steinhaus, H.: The problem of fair division. Econometrica **16**(1), 101–104 (1948)
28. Sühr, T., Biega, A.J., Zehlike, M., Gummadi, K.P., Chakraborty, A.: Two-sided fairness for repeated matchings in two-sided markets: a case study of a ride-hailing platform. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, pp. 3082–3092. ACM, New York (2019)
29. Wolfson, O., Lin, J.: Fairness versus optimality in ridesharing. In: 2017 18th IEEE International Conference on Mobile Data Management (MDM), pp. 118–123. IEEE (2017)