

Reachability of Fair Allocations via Sequential Exchanges

Ayumi Igarashi¹, Naoyuki Kamiyama², Warut Suksompong³, Sheung Man Yuen³

¹University of Tokyo

²Kyushu University

³National University of Singapore

Abstract

In the allocation of indivisible goods, a prominent fairness notion is envy-freeness up to one good (EF1). We initiate the study of reachability problems in fair division by investigating the problem of whether one EF1 allocation can be reached from another EF1 allocation via a sequence of exchanges such that every intermediate allocation is also EF1. We show that two EF1 allocations may not be reachable from each other even in the case of two agents, and deciding their reachability is PSPACE-complete in general. On the other hand, we prove that reachability is guaranteed for two agents with identical or binary utilities as well as for any number of agents with identical binary utilities. We also examine the complexity of deciding whether there is an EF1 exchange sequence that is optimal in the number of exchanges required.

1 Introduction

Fair division, the study of how to allocate resources fairly among competing agents, is a highly active research area at the intersection of mathematics, economics, and computer science. Recently, researchers have drawn connections between fair division and various other fields such as graph theory (Bei et al. 2022; Bilò et al. 2022), extremal combinatorics (Chaudhury et al. 2021; Berendsohn, Boyadzhyska, and Kozma 2022), two-sided matching (Freeman, Micha, and Shah 2021; Igarashi et al. 2023b), and differential privacy (Manurangsi and Suksompong 2023), to name a few.

In fair division, the goal is typically to find an allocation of the resource that is “fair” with respect to the agents’ preferences. When allocating indivisible goods—such as books, clothes, and office supplies—a prominent fairness notion in the literature is *envy-freeness up to one good (EF1)*. In an EF1 allocation of the goods, an agent is allowed to envy another agent only if there exists a good in the latter agent’s bundle whose removal would eliminate this envy. The “up to one good” relaxation is necessitated by the fact that full envy-freeness is sometimes infeasible, as can be seen, for instance, when two agents compete for a single valuable good. It is well-known that an EF1 allocation always exists regardless of the agents’ valuations for the goods (Lipton et al. 2004; Budish 2011).

In this work, we take a different perspective by initiating the study of *reachability* in fair division. Given two fair allocations—an initial allocation and a target allocation—we are interested in whether the target allocation can be reached from the initial allocation via a sequence of operations such that every intermediate allocation is also fair. As an application of our problem, consider a company that wants to redistribute some of its employees between its departments. Since performing the entire redistribution at once may excessively disrupt the operation of the departments, the company prefers to gradually adjust the distribution while maintaining fairness among the departments throughout the process. Another example is a museum that plans to reallocate certain exhibits among its branches—performing one small change at a time can help ensure a seamless transition for the visitors. In this paper, we shall use EF1 as our fairness benchmark and allow any two agents to *exchange* a pair of goods in an operation. The reachability between EF1 allocations, or lack thereof, is an interesting structural property in itself; similar properties have been studied in other collective decision-making scenarios such as voting (Obraztsova et al. 2013; Obraztsova, Elkind, and Faliszewski 2020).

Closest to our work is perhaps a line of work initiated by Gourvès, Lesca, and Wilczynski (2017). These authors considered the “housing market” setting, where the number of agents is the same as the number of goods and each agent receives exactly one good. In their model, a pair of agents is allowed to exchange goods if the two agents are neighbors in a given social network and the exchange benefits both agents. Their paper, along with a series of follow-up papers (Huang and Xiao 2020; Li, Plaxton, and Sinha 2021; Müller and Bentert 2021; Ito et al. 2023), explored the complexity of determining whether an allocation can be reached from another allocation in this model and its variants. More broadly, reachability problems are also known as *reconfiguration* problems (Nishimura 2018); examples of such problems that have been studied include minimum spanning tree (Ito et al. 2011), graph coloring (Johnson et al. 2016), and perfect matching (Bonamy et al. 2019).

1.1 Our Contributions

As is often done in fair division, we assume that every agent is equipped with an additive utility function. We consider

utilities		general	identical	binary	identical + binary
$n = 2$	connected?	✗ (Thm. 1)	✓ (Thm. 3)	✓ (Thm. 4)	✓ (Thm. 3/4)
	optimal?	✗ (Thm. 2)	✓ (Thm. 3)	✓ (Thm. 4)	✓ (Thm. 3/4)
$n \geq 3$	connected?	✗ (Thm. 1)	✗ (Thm. 13)	✗ (Thm. 12)	✓ (Thm. 10)
	optimal?	✗ (Thm. 2)	✗ (Thm. 11)	✗ (Thm. 11)	✗ (Thm. 11)

Table 1: Overview of our results. The top row indicates the class of utility functions considered, “connected?” refers to whether the EF1 exchange graph is always connected, and “optimal?” refers to whether there always exists an optimal EF1 exchange path between any two EF1 allocations provided that the EF1 exchange graph is connected.

an “exchange graph” with allocations as vertices. The first question we study is whether it is always possible for agents to reach a target EF1 allocation from an initial EF1 allocation by exchanging goods sequentially with each other while maintaining the EF1 property in all the intermediate allocations; in other words, we ask whether the subgraph of the exchange graph consisting of all EF1 allocations is *connected*. The second question is whether we could perform this exchange process using as few exchanges as would be required if the intermediate allocations need not be EF1; that is, whether there exists an EF1 exchange path which is *optimal* in terms of the number of exchanges required. Note that each agent’s bundle size remains unchanged throughout the process since every operation is an exchange of goods. Our formal model is described in Section 2.

In Section 3, we investigate the basic setting where there are only two agents. Perhaps surprisingly, we establish negative results even for this setting: the EF1 exchange graph may not be connected, and even for those instances in which it is connected, optimal EF1 exchange paths may not exist between EF1 allocations. Therefore, we consider restricted classes of utility functions, where the picture becomes much more positive. Specifically, we show that an optimal EF1 exchange path always exists between any two EF1 allocations if the utilities are identical *or* binary; this implies the connectivity of the EF1 exchange graph in these cases as well.

In Section 4, we explore the general setting of three or more agents. Interestingly, we show that finding the smallest number of exchanges between two allocations is NP-hard in this setting even if we disregard the EF1 restriction. In addition, we establish that deciding whether an EF1 exchange path exists between two allocations is PSPACE-complete, and deciding whether an optimal such path exists is NP-hard even for four agents with identical utilities. We also examine restricted utility functions in more detail. In particular, we show that while connectivity of the EF1 exchange graph is guaranteed for identical binary utilities, the same holds neither for identical utilities nor for binary utilities separately. Furthermore, the optimality of EF1 exchange paths cannot be guaranteed even for identical binary utilities. Overall, our findings demonstrate that the case of three or more agents is much less tractable than that of two agents in our setting.

With the exception of hardness results (Theorems 8, 9, and 14), our results are summarized in Table 1. For the positive results, we also show that the corresponding exchange paths can be found in polynomial time. All omitted proofs can be found in the full version of our paper (Igarashi et al. 2023a).

2 Preliminaries

Let N be a set of $n \geq 2$ agents, and M be a set of $m \geq 1$ goods. We typically denote the agents by $1, \dots, n$ and the goods by g_1, \dots, g_m . A *bundle* is a (possibly empty) subset of goods. An *allocation* $\mathcal{A} = (A_1, \dots, A_n)$ is an ordered partition of M into n bundles such that bundle A_i is allocated to agent $i \in N$. An (*allocation*) *size vector* $\vec{s} = (s_1, \dots, s_n)$ is a vector of non-negative integers such that $\sum_{i \in N} s_i = m$ and $s_i = |A_i|$ for all $i \in N$.

Given N , M , and \vec{s} , define the *exchange graph* $G = G(N, M, \vec{s})$ as a simple undirected graph with the following properties: the set of vertices consists of all allocations \mathcal{A} with size vector \vec{s} , and the set of edges consists of all pairs $\{\mathcal{A}, \mathcal{B}\}$ such that $\mathcal{B} = (B_1, \dots, B_n)$ can be obtained from $\mathcal{A} = (A_1, \dots, A_n)$ by having two agents exchange one pair of goods with each other—that is, there exist distinct agents $i, j \in N$ and goods $g \in A_i$ and $g' \in A_j$ such that $B_i = (A_i \cup \{g'\}) \setminus \{g\}$, $B_j = (A_j \cup \{g\}) \setminus \{g'\}$, and $B_k = A_k$ for all $k \in N \setminus \{i, j\}$. Note that the exchange graph is a non-empty connected graph. A path from one allocation to another on the graph is called an *exchange path*. The *distance* between two allocations is the length of a shortest exchange path between them.

Each agent $i \in N$ has a *utility function* $u_i : 2^M \rightarrow \mathbb{R}_{\geq 0}$ that maps bundles to non-negative real numbers. We write $u_i(g)$ instead of $u_i(\{g\})$ for a single good $g \in M$, and assume that the utility functions are additive, i.e., $u_i(M') = \sum_{g \in M'} u_i(g)$ for all $i \in N$ and $M' \subseteq M$. The utility functions are *identical* if $u_i = u_j$ for all $i, j \in N$ —we shall use u to denote the common utility function in this case. The utility functions are *binary* if $u_i(g) \in \{0, 1\}$ for all $i \in N$ and $g \in M$. An allocation \mathcal{A} is *envy-free up to one good (EF1)* if for all pairs $i, j \in N$ such that $A_j \neq \emptyset$, there exists a good $g \in A_j$ such that $u_i(A_i) \geq u_i(A_j \setminus \{g\})$.

Given N , M , \vec{s} , and $(u_i)_{i \in N}$, define the *EF1 exchange graph* $H = H(N, M, \vec{s}, (u_i)_{i \in N})$ as the subgraph of the exchange graph G induced by all EF1 allocations, i.e., H contains all vertices in G that correspond to EF1 allocations and all edges in G incident to two EF1 allocations. As we shall see later, EF1 exchange graphs are not always connected, unlike exchange graphs. An exchange path using only the edges in H is called an *EF1 exchange path*. An EF1 exchange path is *optimal* if its length is equal to the distance between the two corresponding allocations (in G).

An *instance* consists of a set of agents N , a set of goods M , a size vector \vec{s} , and agents’ utility functions $(u_i)_{i \in N}$.

3 Two Agents

In this section, we examine properties of the EF1 exchange graph when there are only two agents. We remark that this is an important special case in fair division and has been the focus of several prior papers in the area.¹

We first consider the question of whether the EF1 exchange graph is necessarily connected. One may intuitively think that with only two agents, an EF1 exchange path is guaranteed between any two EF1 allocations because the two agents only need to consider the envy between themselves. An agent may then carefully select a good from her bundle to exchange with the other agent so as to ensure that the subsequent allocation is also EF1. However, this in fact cannot always be done, as our first result shows.

Theorem 1. *There exists an instance with $n = 2$ agents with the same ordinal preferences over the goods such that the EF1 exchange graph is disconnected.*

Proof. Consider the utility of the goods as follows:

g	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8
$u_1(g)$	3	3	2	2	2	2	0	0
$u_2(g)$	3	3	1	1	1	1	0	0

Let \mathcal{A} and \mathcal{B} be allocations such that $A_1 = B_2 = \{g_1, g_2, g_7, g_8\}$ and $A_2 = B_1 = \{g_3, g_4, g_5, g_6\}$ —it can be verified that both \mathcal{A} and \mathcal{B} are EF1. If there exists an EF1 exchange path between \mathcal{A} and \mathcal{B} , then there exists an EF1 allocation \mathcal{A}' adjacent to \mathcal{A} on the exchange path. Without loss of generality, \mathcal{A}' can be reached from \mathcal{A} by exchanging g_3 with either g_1 or g_7 . If g_3 is exchanged with g_1 , then agent 1 envies agent 2 by more than one good. If g_3 is exchanged with g_7 , then agent 2 envies agent 1 by more than one good. Therefore, neither of these exchanges leads to an EF1 allocation, so \mathcal{A}' cannot be EF1. Hence, no EF1 exchange path exists between \mathcal{A} and \mathcal{B} . \square

Next, we consider the question of whether an *optimal* EF1 exchange path always exists between two EF1 allocations. By Theorem 1, even an EF1 exchange path may not exist, so an optimal such path does not necessarily exist either. We therefore focus on instances in which the EF1 exchange graph is *connected*. It turns out that even for such instances, an optimal EF1 exchange path still might not exist.

Theorem 2. *There exists an instance with $n = 2$ agents satisfying the following properties: the EF1 exchange graph is connected, but for some pair of EF1 allocations, no optimal EF1 exchange path exists between them.*

Proof. Consider $\vec{s} = (3, 3)$ and the utility of the goods as follows:

g	g_1	g_2	g_3	g_4	g_5	g_6
$u_1(g)$	5	3	1	0	2	2
$u_2(g)$	0	3	1	5	2	2

¹Plaut and Roughgarden (2020, Sec. 1.1.1) discussed the significance of the two-agent setting in detail.

Let \mathcal{B} be the allocation such that $B_1 = \{g_1, g_2, g_3\}$ and $B_2 = \{g_4, g_5, g_6\}$ —it can be verified that \mathcal{B} is EF1. We first prove that the EF1 exchange graph is connected by constructing an EF1 exchange path between any EF1 allocation \mathcal{A} and the EF1 allocation \mathcal{B} . If g_1 is not with agent 1 or g_4 is not with agent 2 in \mathcal{A} , perform any exchange involving g_1 and/or g_4 so that g_1 is now with agent 1 and g_4 is now with agent 2. After the exchange, for each $i \in \{1, 2\}$, agent i 's bundle is worth at least 5 to her, while any two goods in agent $(3 - i)$'s bundle are worth at most 5 to agent i , so the allocation is EF1. Now, we can exchange the goods in $\{g_2, g_3, g_5, g_6\}$ in an arbitrary order to reach \mathcal{B} after at most two more exchanges.

We next prove that an optimal EF1 exchange path between allocations \mathcal{C} and \mathcal{D} does not exist, where $C_1 = \{g_2, g_3, g_4\}$, $C_2 = \{g_1, g_5, g_6\}$, $D_1 = \{g_4, g_5, g_6\}$, and $D_2 = \{g_1, g_2, g_3\}$; it can be verified that both \mathcal{C} and \mathcal{D} are EF1, and the distance between \mathcal{C} and \mathcal{D} is 2 (through exchanging $g_2 \leftrightarrow g_5$ and $g_3 \leftrightarrow g_6$). Suppose there exists an optimal EF1 exchange path between \mathcal{C} and \mathcal{D} , and let \mathcal{C}' be the EF1 allocation between \mathcal{C} and \mathcal{D} on the exchange path. Since \mathcal{C} and \mathcal{C}' are adjacent, one good from $\{g_2, g_3\}$ must be exchanged with one good from $\{g_5, g_6\}$ in \mathcal{C} to reach \mathcal{C}' . However, no matter which goods are exchanged with this restriction, there exists $i \in \{1, 2\}$ such that agent i 's bundle is worth 3 to her and agent $(3 - i)$'s bundle is worth $5 + 5$ to agent i , contradicting the EF1 property of \mathcal{C}' . Therefore, no optimal EF1 exchange path exists between \mathcal{C} and \mathcal{D} . \square

In light of these negative results, we turn our attention to special classes of utility functions: identical utilities and binary utilities. We prove that for these two classes of utility functions, the EF1 exchange graph is always connected, and moreover, an optimal EF1 exchange path exists between every pair of EF1 allocations.

Theorem 3. *Let an instance with $n = 2$ agents and identical utilities be given. Then, the EF1 exchange graph is connected. Moreover, there exists an optimal EF1 exchange path between any two EF1 allocations, and this path can be computed in polynomial time.*

Theorem 4. *Let an instance with $n = 2$ agents and binary utilities be given. Then, the EF1 exchange graph is connected. Moreover, there exists an optimal EF1 exchange path between any two EF1 allocations, and this path can be computed in polynomial time.*

To establish these results, we shall prove by induction on t that two EF1 allocations with distance t have an optimal EF1 exchange path between them. For the base case $t = 0$, an optimal EF1 exchange path trivially exists. For the inductive step, let $t \geq 1$ be given, and assume the inductive hypothesis that any two EF1 allocations with distance $t - 1$ have an EF1 exchange path of length $t - 1$. Now, let $\mathcal{A} = (A_1, A_2)$ and $\mathcal{B} = (B_1, B_2)$ be any two EF1 allocations with distance t ; this means that $|A_1 \setminus B_1| = |A_2 \setminus B_2| = t$. Define $X = A_1 \setminus B_1 = \{x_1, \dots, x_t\}$ and $Y = A_2 \setminus B_2 = \{y_1, \dots, y_t\}$. We show that there exist goods $x_k \in X$ and $y_\ell \in Y$ such that exchanging them in \mathcal{A} leads to an EF1 allocation $\mathcal{A}' = (A'_1, A'_2)$. If this is possible, then $|A'_1 \setminus B_1| = |A'_2 \setminus B_2| =$

$t - 1$, which implies that the distance between \mathcal{A}' and \mathcal{B} is $t - 1$. By the inductive hypothesis, there exists an EF1 exchange path between \mathcal{A}' and \mathcal{B} of length $t - 1$. This means that there exists an EF1 exchange path between \mathcal{A} and \mathcal{B} via \mathcal{A}' of length t , which is optimal, hence completing the proof.

For the time complexity, for each pair of goods from $X \times Y$, one can check in polynomial time whether exchanging them leads to an EF1 allocation. Since there are at most t^2 pairs of goods to check at each step, and there are t steps in the path, the running time claim follows.

We show the proof of Theorem 3, and refer the reader to the full version of our paper (Igarashi et al. 2023a) for the proof of Theorem 4.

Proof of Theorem 3. We follow the notation and inductive outline described before this proof. Assume that the goods in X and Y are arranged in descending order of preference, i.e., $u(x_i) \geq u(x_j)$ and $u(y_i) \geq u(y_j)$ whenever $i < j$. Denote $\Delta_k = u(y_k) - u(x_k)$ for all $k \in \{1, \dots, t\}$. Define $A'_1 = (A_1 \cup \{y_1\}) \setminus \{x_1\}$ and $A'_2 = (A_2 \cup \{x_1\}) \setminus \{y_1\}$ to be the bundles after exchanging x_1 and y_1 . If (A'_1, A'_2) is EF1, we are done by induction. Otherwise, we assume without loss of generality that in the allocation (A'_1, A'_2) , agent 2 envies agent 1 by more than one good. Let x be a highest-utility good in A_1 —we may assume that $x \neq x_k$ for all $k \geq 2$. Since (A_1, A_2) is an EF1 allocation, we have $u(x) \geq \gamma := u(A_1) - u(A_2)$.

If both $x = x_1$ and $\Delta_1 < 0$ are true, then

$$\begin{aligned} u(A'_2) &= u(A_2) - \Delta_1 \\ &> u(A_2) \\ &\geq u(A_1 \setminus \{x\}) \\ &= u(A_1 \setminus \{x_1\}) \\ &= u(A'_1 \setminus \{y_1\}), \end{aligned}$$

which shows that agent 2 does not envy agent 1 by more than one good in (A'_1, A'_2) —a contradiction. Therefore, we must have $x \neq x_1$ or $\Delta_1 \geq 0$. If $x \neq x_1$, then both x and y_1 belong to A'_1 . If $x = x_1$ and $\Delta_1 \geq 0$, then y_1 belongs to A'_1 and $u(y_1) \geq u(x)$. Hence, in either case, we have

$$\begin{aligned} \max\{u(x), u(y_1)\} &< u(A'_1) - u(A'_2) \\ &= u(A_1) - u(A_2) + 2\Delta_1, \end{aligned}$$

which implies

$$\gamma + 2\Delta_1 > \max\{u(x), u(y_1)\}. \quad (1)$$

We claim that there exists $k \in \{2, \dots, t\}$ such that

$$2\Delta_k \leq u(x) - \gamma. \quad (2)$$

Suppose on the contrary that $2\Delta_k > u(x) - \gamma$ for all $k \in \{2, \dots, t\}$. Since every good in A_1 has value at most $u(x)$ and every good in $B_1 \setminus A_1$ has value at most $u(y_1)$, it holds that every good in B_1 has value at most $\max\{u(x), u(y_1)\}$. As (B_1, B_2) is an EF1 allocation, we have

$$\begin{aligned} \max\{u(x), u(y_1)\} &\geq u(B_1) - u(B_2) \\ &= (u(A_1) - u(A_2)) + \sum_{k=1}^t 2\Delta_k \end{aligned}$$

$$\begin{aligned} &= \gamma + 2\Delta_1 + \sum_{k=2}^t 2\Delta_k \\ &\geq \gamma + 2\Delta_1 + \sum_{k=2}^t (u(x) - \gamma) \\ &\geq \gamma + 2\Delta_1, \end{aligned}$$

where the last inequality holds because $u(x) \geq \gamma$ and $t \geq 1$. This contradicts (1). Therefore, let $k \in \{2, \dots, t\}$ be an index that satisfies (2). We now claim that we must have

$$2\Delta_k \geq \max\{u(x), u(y_1)\} - 2u(y_1) - \gamma. \quad (3)$$

Suppose on the contrary that $2\Delta_k < \max\{u(x), u(y_1)\} - 2u(y_1) - \gamma$. Then we have

$$\begin{aligned} &\max\{u(x), u(y_1)\} - 2u(y_1) - \gamma \\ &> 2\Delta_k && \text{(by assumption)} \\ &\geq -2u(x_k) && \text{(since } u(y_k) \geq 0) \\ &\geq -2u(x_1), && \text{(since } u(x_k) \leq u(x_1)) \end{aligned}$$

which implies

$$\gamma + 2\Delta_1 < \max\{u(x), u(y_1)\},$$

contradicting (1). This establishes (3).

Combining inequalities (2) and (3), we have

$$\begin{aligned} -u(y_1) &\leq \max\{u(x) - u(y_1), 0\} - u(y_1) \\ &= \max\{u(x), u(y_1)\} - 2u(y_1) \\ &\leq \gamma + 2\Delta_k && \text{(by (3))} \\ &\leq u(x), && \text{(by (2))} \end{aligned}$$

which implies $\gamma + 2\Delta_k \in [-u(y_1), u(x)]$. We claim that exchanging x_k and y_k results in an EF1 allocation, i.e., the allocation comprising $A''_1 = (A_1 \cup \{y_k\}) \setminus \{x_k\}$ and $A''_2 = (A_2 \cup \{x_k\}) \setminus \{y_k\}$ is EF1. This is because

$$\begin{aligned} u(A''_1) - u(A''_2) &= u(A_1) - u(A_2) + 2\Delta_k \\ &= \gamma + 2\Delta_k \in [-u(y_1), u(x)], \end{aligned}$$

where $x \in A''_1$ and $y_1 \in A''_2$ —note that $x (\neq x_k)$ and y_1 were not exchanged going from \mathcal{A} to \mathcal{A}'' . This completes the induction and therefore the proof. \square

Since the EF1 exchange graph H is a subgraph of the exchange graph G , the distance between two allocations (in G) cannot be greater than the length of the shortest EF1 exchange path between the two allocations in H . In Theorems 3 and 4, the polynomial-time algorithms find EF1 exchange paths in H that are optimal in the exchange graph G ; such exchange paths must also be the shortest possible ones in H .

4 Three or More Agents

In this section, we address the general case where there are more than two agents. We shall see that this case is less tractable, both existentially and computationally.

Before we present our results on the EF1 exchange graph, we provide some insights on finding the distance between two allocations regardless of EF1 considerations. Observe that finding this distance for two agents is simple, as the

distance equals the number of goods from each of the two bundles that need to be exchanged. However, this task is not so trivial for more agents—in fact, we shall show that it is NP-hard. To this end, we draw an interesting connection between this distance and the maximum number of disjoint cycles in a graph constructed based on the allocations. We start off by detailing how to construct such a graph.

Let N , M , and \vec{s} be given, and let $\mathcal{A} = (A_1, \dots, A_n)$ and $\mathcal{B} = (B_1, \dots, B_n)$ be two allocations with size vector \vec{s} . Define $G_{\mathcal{A},\mathcal{B}} = (V_{\mathcal{A},\mathcal{B}}, E_{\mathcal{A},\mathcal{B}})$ as a directed multigraph consisting of a set of vertices $V_{\mathcal{A},\mathcal{B}} = N$ and a set of (directed) edges $E_{\mathcal{A},\mathcal{B}} = \{e_1, \dots, e_m\}$. For each $k \in \{1, \dots, m\}$, the edge e_k represents the good g_k , and $e_k = (i, j)$ if and only if $g_k \in A_i \cap B_j$, i.e., g_k is in agent i 's bundle in \mathcal{A} and in agent j 's bundle in \mathcal{B} (possibly $i = j$). Note that for every vertex i , its indegree is equal to its outdegree, which is equal to s_i , the number of goods in agent i 's bundle. Let $\mathcal{C}_{\mathcal{A},\mathcal{B}}$ be the collection of partitions of $E_{\mathcal{A},\mathcal{B}}$ into directed circuits.² Note that $\mathcal{C}_{\mathcal{A},\mathcal{B}}$ is non-empty—for example, a partition of $E_{\mathcal{A},\mathcal{B}}$ into directed circuits can be constructed in the following way: start with any vertex with an outdegree of at least 1, traverse a path until some vertex v is encountered for the second time, remove the resulting directed cycle from v to itself, and repeat on the remaining graph; the remaining graph still satisfies the condition that every vertex has its indegree equal to its outdegree. Let $c_{\mathcal{A},\mathcal{B}}^* = \max_{C \in \mathcal{C}_{\mathcal{A},\mathcal{B}}} |C|$ be the maximum cardinality of such a partition. Note that a partition with the maximum cardinality must consist only of directed *cycles*; otherwise, if it contains a circuit that passes through a vertex more than once, we can break the circuit into two smaller circuits, contradicting the fact that this partition has the maximum cardinality. We claim that the distance between allocations \mathcal{A} and \mathcal{B} is $m - c_{\mathcal{A},\mathcal{B}}^*$.

Proposition 5. *Let N , M , and \vec{s} be given, and let \mathcal{A} and \mathcal{B} be two allocations with size vector \vec{s} . Then, the distance between \mathcal{A} and \mathcal{B} is $m - c_{\mathcal{A},\mathcal{B}}^*$.*

Having found a correspondence for the distance between two allocations, a natural question is whether there exists an efficient algorithm to compute this distance. It turns out that computing this distance is an NP-hard problem, so no polynomial-time algorithm exists unless $P = NP$. We show this via a series of reductions.

We start by considering the decision problem DIRECTED TRIANGLE PARTITION: given a directed graph with no directed cycles of length 1 or 2, determine whether there is a partition of edges into triangles (i.e., directed cycles of length 3). This decision problem is NP-hard via a reduction from 3SAT similar to that used by Holyer (1981) in his proof of the corresponding result for *undirected* graphs.

Lemma 6. DIRECTED TRIANGLE PARTITION is NP-hard.

We now use Lemma 6 to show that computing $c_{\mathcal{A},\mathcal{B}}^*$ is NP-hard.

Lemma 7. *Given a directed graph such that for each vertex, its indegree and outdegree are equal, computing the maximum*

²Recall that a directed circuit is a non-empty walk such that the first vertex and the last vertex coincide; we consider a self-loop to be a directed circuit as well.

cardinality of a partition of the edges into directed cycles is an NP-hard problem.

Proof. The result follows from reducing DIRECTED TRIANGLE PARTITION to the problem of deciding whether there exists a partition of the edges of a directed graph into $|E|/3$ directed cycles. Let $G = (V, E)$ be an instance of DIRECTED TRIANGLE PARTITION. If there is some vertex with unequal indegree and outdegree, then G cannot be edge-partitioned into triangles. Otherwise, since G does not have cycles of length 1 or 2 (by definition of DIRECTED TRIANGLE PARTITION), the edges of G can be partitioned into triangles if and only if the maximum cardinality of a partition of the edges into directed cycles is $|E|/3$. Since DIRECTED TRIANGLE PARTITION is NP-hard by Lemma 6, so is the problem of finding the maximum cardinality of a partition of the edges into directed cycles. \square

Proposition 5 and Lemma 7 imply the following theorem.

Theorem 8. *Finding the distance between two allocations is an NP-hard problem.*

Proof. Start with an instance $\mathcal{G} = (V, E)$ of the problem described in Lemma 7, and denote $V = \{v_1, \dots, v_n\}$. We shall construct, in polynomial time, an instance of the problem of finding the distance between two allocations. Let $N = \{1, \dots, n\}$ be the set of agents, $M = \{g_e\}_{e \in E}$ be the set of goods, and $s_i = |\{e \in E \mid \exists j \in N, e = (v_i, v_j)\}|$ be the size of agent i 's bundle for each $i \in N$. The initial allocation $\mathcal{A} = (A_1, \dots, A_n)$ and target allocation $\mathcal{B} = (B_1, \dots, B_n)$ are such that $A_i = \{g_e \in M \mid \exists j \in N, e = (v_i, v_j) \in E\}$ and $B_i = \{g_e \in M \mid \exists j \in N, e = (v_j, v_i) \in E\}$ for each $i \in N$. Note that this induces the graph $G_{\mathcal{A},\mathcal{B}}$ isomorphic to \mathcal{G} . The distance between \mathcal{A} and \mathcal{B} is $|E| - c_{\mathcal{A},\mathcal{B}}^*$, by Proposition 5. Therefore, if we can find this distance, then we can find $c_{\mathcal{A},\mathcal{B}}^*$, solving the problem instance from Lemma 7. \square

4.1 General Utilities

We now discuss properties of the EF1 exchange graph. The non-connectivity and non-optimality results for three or more agents can be derived from Theorems 1 and 2 for two agents by adding agents with no good in both the initial and target allocations and having zero utility for every good. Moreover, we show next that deciding whether an EF1 exchange path exists is a PSPACE-complete problem.

Theorem 9. *Deciding the existence of an EF1 exchange path between two EF1 allocations is PSPACE-complete.*

Proof. First, we show membership in PSPACE—recall that PSPACE is the set of all decision problems that can be solved by a deterministic polynomial-space Turing machine. We can solve the problem non-deterministically by simply guessing an EF1 exchange path between the two EF1 allocations. Since the total number of allocations is at most n^m , if there exists an EF1 exchange path between the two allocations, then there exists one with length at most n^m ; such a path can be represented in polynomial space (i.e., using a polynomial number of bits). This shows that the problem is in NPSpace, the set of all decision problems that can be

solved by a non-deterministic polynomial-space Turing machine. By Savitch’s Theorem, $\text{NPSpace} \subseteq \text{PSPACE}$ (Savitch 1970), which implies that this problem is in PSPACE .

To prove that our problem is PSPACE-hard , we shall reduce the PERFECT MATCHING RECONFIGURATION problem for a balanced (undirected) bipartite graph to our problem. Recall that the PERFECT MATCHING RECONFIGURATION problem is the task of deciding if two perfect matchings of a balanced bipartite graph can be reached from each other via a sequence of *flips*, i.e., given perfect matchings \widetilde{W}_0 and \widetilde{W} of a balanced bipartite graph $\widetilde{G} = (\widetilde{V}, \widetilde{E})$, whether there exists a sequence of perfect matchings $\widetilde{W}_0, \widetilde{W}_1, \dots, \widetilde{W}_t$ such that

- $\widetilde{W}_t = \widetilde{W}$, and
- for each $z \in \{1, \dots, t\}$, there exist edges $\widetilde{e}_z^1, \widetilde{e}_z^2, \widetilde{e}_z^3, \widetilde{e}_z^4$ of \widetilde{G} such that $\widetilde{W}_{z-1} \setminus \widetilde{W}_z = \{\widetilde{e}_z^1, \widetilde{e}_z^3\}$, $\widetilde{W}_z \setminus \widetilde{W}_{z-1} = \{\widetilde{e}_z^2, \widetilde{e}_z^4\}$, and $\widetilde{e}_z^1 \widetilde{e}_z^2 \widetilde{e}_z^3 \widetilde{e}_z^4$ forms a cycle.

The operation from \widetilde{W}_{z-1} to \widetilde{W}_z is called a *flip*, and we say that \widetilde{W}_{z-1} and \widetilde{W}_z are *adjacent* to each other. PERFECT MATCHING RECONFIGURATION is known to be PSPACE-hard (Bonamy et al. 2019). Let $|\widetilde{V}| = 2v$, and let the two independent sets of \widetilde{G} be $\widetilde{P} = \{\widetilde{p}_1, \dots, \widetilde{p}_v\}$ and $\widetilde{Q} = \{\widetilde{q}_1, \dots, \widetilde{q}_v\}$. For each $i \in \{1, \dots, v\}$, let $\widetilde{q}_{k_i} \in \widetilde{Q}$ be the vertex adjacent to \widetilde{p}_i in \widetilde{W}_0 , and let $\widetilde{q}_{\ell_i} \in \widetilde{Q}$ be the vertex adjacent to \widetilde{p}_i in \widetilde{W} . We shall show that this problem instance can be reduced to an instance of deciding the existence of an EF1 exchange path between two EF1 allocations.

Define an instance of the EF1 exchange path problem as follows: let $N = \{0, 1, \dots, v\}$ be the set of agents, $M = \{p_1, \dots, p_v, q_1, \dots, q_v, r_1, r_2, r_3, r_4\}$ be the set of goods, and the utility function of each agent be

- $u_0(g) = 0$ for all $g \in M$, and
- for $i \in \{1, \dots, v\}$,

$$u_i(g) = \begin{cases} 3 & \text{if } g \in \{p_i\} \cup \{q_k \mid \{\widetilde{p}_i, \widetilde{q}_k\} \in \widetilde{E}\}; \\ 2 & \text{if } g \in \{r_1, r_2, r_3, r_4\}; \\ 0 & \text{otherwise.} \end{cases}$$

In the initial allocation \mathcal{A}_0 , agent 0 has the bundle $\{r_1, r_2, r_3, r_4\}$ and agent i has the bundle $\{p_i, q_{k_i}\}$ for each $i \in \{1, \dots, v\}$. In the target allocation \mathcal{A} , agent 0 again has the bundle $\{r_1, r_2, r_3, r_4\}$ and agent i has the bundle $\{p_i, q_{\ell_i}\}$ for each $i \in \{1, \dots, v\}$. Observe that both allocations are EF1—agent 0 assigns zero utility to every bundle, while each agent $i \in \{1, \dots, v\}$ assigns a utility of 6 to her own bundle, a utility of at most 6 to the bundle of every agent in $\{1, \dots, v\} \setminus \{i\}$, and a utility of $6 + 2$ to agent 0’s bundle. Clearly, this instance can be constructed in polynomial time.

Suppose first that there exists a sequence of adjacent perfect matchings from \widetilde{W}_0 to \widetilde{W} . Then each flip from \widetilde{W}_{z-1} to \widetilde{W}_z corresponds to an exchange in the EF1 exchange path problem: if $\widetilde{W}_{z-1} \setminus \widetilde{W}_z = \{\{\widetilde{p}_i, \widetilde{q}_k\}, \{\widetilde{p}_j, \widetilde{q}_\ell\}\}$ and $\widetilde{W}_z \setminus \widetilde{W}_{z-1} = \{\{\widetilde{p}_i, \widetilde{q}_\ell\}, \{\widetilde{p}_j, \widetilde{q}_k\}\}$, then exchange q_k in agent i ’s bundle with q_ℓ in agent j ’s bundle. The new allocation is also EF1—as before, agent 0 assigns zero utility

to every bundle, while each agent $i' \in \{1, \dots, v\}$ assigns a utility of 6 to her own bundle, a utility of at most 6 to the bundle of every agent in $\{1, \dots, v\} \setminus \{i'\}$, and a utility of $6 + 2$ to agent 0’s bundle. By performing the exchanges according to the flips in sequence, we reach the target allocation. Therefore, an EF1 exchange path exists.

Conversely, assume that an EF1 exchange path exists between the initial allocation \mathcal{A}_0 and the target allocation \mathcal{A} . Consider the sequence of EF1 allocations $\mathcal{A}_0, \mathcal{A}_1, \dots, \mathcal{A}_t = \mathcal{A}$. We show by induction that for every intermediate allocation \mathcal{A}_z , every agent $i \in \{1, \dots, v\}$ assigns a utility of 6 to her own bundle (consisting of p_i and q_k for some k), and agent 0 retains $\{r_1, r_2, r_3, r_4\}$. The base case $z = 0$ is trivial. For the inductive case, suppose that the statement is true for $z - 1$. If some agent $i \in \{1, \dots, v\}$ attempts to exchange one of her goods with a good from agent 0, then agent i ’s new bundle has utility 5 but agent 0’s new bundle has utility $6 + 3$ for agent i , which violates EF1. Therefore, agent i must exchange goods with agent j for some $j \in \{1, \dots, v\}$. Note that agent 0’s bundle is worth $6 + 2$ to agent i and j , so agent i ’s and j ’s own bundles must be worth at least 6 to i and j respectively. If agent i gives p_i to agent j , then agent j ’s new bundle consists of p_i (worth zero to her) and some q_ℓ , which is worth at most 3 to her—this violates EF1. By the same reasoning, agent j cannot give p_j to agent i . Therefore, they must exchange q_k in agent i ’s bundle with q_ℓ in agent j ’s bundle. As agent i and j must have bundles worth at least 6 to each of them, q_ℓ must be worth 3 to agent i and q_k must be worth 3 to agent j . This completes the induction.

As a result, the perfect matchings \widetilde{W}_{z-1} (corresponding to \mathcal{A}_{z-1}) and \widetilde{W}_z (corresponding to \mathcal{A}_z) must be adjacent to each other for all z , where $\widetilde{W}_{z-1} \setminus \widetilde{W}_z = \{\{\widetilde{p}_i, \widetilde{q}_k\}, \{\widetilde{p}_j, \widetilde{q}_\ell\}\}$ and $\widetilde{W}_z \setminus \widetilde{W}_{z-1} = \{\{\widetilde{p}_i, \widetilde{q}_\ell\}, \{\widetilde{p}_j, \widetilde{q}_k\}\}$. It follows that a sequence of adjacent perfect matchings $\widetilde{W}_0, \widetilde{W}_1, \dots, \widetilde{W}_t = \widetilde{W}$ indeed exists. This completes the proof. \square

Regarding the existence of *optimal* EF1 exchange paths, we shall show later in Theorem 14 that the corresponding decision problem is NP-hard even for four agents with identical utilities.

4.2 Identical Binary Utilities

We now consider the most restrictive class of utility functions in our paper: those that are identical *and* binary. We show that the EF1 exchange graph is connected for any number of agents with such utility functions.

Theorem 10. *Let an instance with $n \geq 2$ agents and identical binary utility functions be given. Then, the EF1 exchange graph is connected. Moreover, an EF1 exchange path between any two allocations can be found in polynomial time.*

Proof. Let \mathcal{A} and \mathcal{B} be two EF1 allocations. Since every good is worth either 1 or 0 to every agent, every agent’s bundle in \mathcal{A} and \mathcal{B} must have a utility of either $\lfloor u(M)/n \rfloor$ or $\lfloor u(M)/n \rfloor + 1$ (otherwise, the gap between the utilities of some two agents’ bundles is at least 2, and the corresponding allocation is not EF1). Let N' be the set of agents whose bundles in \mathcal{A} and \mathcal{B} have different utilities. Note that half

of the agents in N' have bundles worth $\lfloor u(M)/n \rfloor$ in \mathcal{A} and $\lfloor u(M)/n \rfloor + 1$ in \mathcal{B} ; the other half have bundles worth $\lfloor u(M)/n \rfloor + 1$ in \mathcal{A} and $\lfloor u(M)/n \rfloor$ in \mathcal{B} . If $N' \neq \emptyset$, let agent $i \in N'$ be an agent with a bundle worth $\lfloor u(M)/n \rfloor$ in \mathcal{A} , and let g_i be a good with utility 0 in A_i —this good exists because agent i has at least $\lfloor u(M)/n \rfloor + 1$ goods in her bundle (due to B_i 's utility of $\lfloor u(M)/n \rfloor + 1$) but only has utility $\lfloor u(M)/n \rfloor$ in A_i . Let agent $j \in N'$ be an agent with a bundle worth $\lfloor u(M)/n \rfloor + 1$ in \mathcal{A} , and let g_j be a good with utility 1 in A_j —this good exists because A_j has utility at least 1. Exchange g_i with g_j ; it can be verified that the resulting allocation is EF1. As this exchange reduces the size of the set N' by two, we can repeatedly make such exchanges between two agents in N' until $N' = \emptyset$. Note that such exchanges can be performed in polynomial time.

At this point, we have shown that there exists an EF1 allocation \mathcal{A}' such that an EF1 exchange path exists between \mathcal{A} and \mathcal{A}' , and for every agent i , her bundles in \mathcal{A}' and \mathcal{B} have the same utility. Define the item graph $G_{\mathcal{A}',\mathcal{B}}$ as in the beginning of Section 4, and consider its subgraph with only the edges representing the goods with utility 1. For each agent, the indegree and the outdegree of the corresponding vertex in this subgraph are equal, so we can perform exchanges to ‘resolve’ these edges. Specifically, suppose there is an edge $e_x = (i, j)$ corresponding to a good g_x , where $i \neq j$. By the degree condition, there must exist another edge $e_y = (j, k)$ corresponding to a good g_y , where $j \neq k$ but possibly $k = i$. We let agents i and j exchange g_x and g_y , so g_x is now with its correct owner, agent j . Hence, at least one new good goes to the correct agent after the exchange. This exchange process can be performed in polynomial time, and no agent’s utility changes during the process, which means that the intermediate allocations are all EF1. Similarly, if we consider the subgraph with only the edges representing the goods with utility 0, we can perform exchanges to resolve these edges as well. Therefore, there exists an EF1 exchange path from \mathcal{A}' to \mathcal{B} , and thus an EF1 exchange path from \mathcal{A} to \mathcal{B} , and this path can be found in polynomial time. \square

In spite of this positive result, the polynomial-time algorithm described in Theorem 10 does not necessarily find an optimal EF1 exchange path between the two allocations. In fact, even for the special case where the EF1 exchange graph H and the exchange graph G coincide (e.g., when every agent assigns zero utility to every good, so every allocation is EF1), it is NP-hard to compute an optimal EF1 exchange path by Theorem 8, regardless of whether optimality refers to the length of the shortest path in G or in H . Hence, a polynomial-time algorithm for this task does not exist unless $P = NP$. Moreover, we show next that, somewhat surprisingly, an optimal EF1 exchange path (with respect to G) is not guaranteed to exist even for identical binary utilities.

Theorem 11. *For each $n \geq 3$, there exists an instance with n agents with identical binary utility functions satisfying the following properties: the EF1 exchange graph is connected, but for some pair of EF1 allocations, no optimal EF1 exchange path exists between them.*

Proof sketch (for three agents). For $n = 3$ agents, consider

$\vec{s} = (2, 2, 2)$ and the utility of the goods as follows:

g	g_1	g_2	g_3	g_4	g_5	g_6
$u(g)$	1	1	1	0	0	0

Note that the EF1 exchange graph is connected by Theorem 10. However, an optimal EF1 exchange path between \mathcal{A} and \mathcal{B} does not exist, where $A_1 = \{g_2, g_6\}$, $A_2 = \{g_3, g_4\}$, $A_3 = \{g_1, g_5\}$, and $B_i = \{g_i, g_{i+3}\}$ for $i \in \{1, 2, 3\}$. \square

4.3 Binary Utilities

We saw in Theorem 10 that the EF1 exchange graph is always connected for any number of agents with identical binary utilities. Now, we consider the case where the agents have binary utilities which may differ between agents. Interestingly, it turns out that the EF1 exchange graph is not necessarily connected in this case, even when there are three agents. This also provides a contrast to the case of two agents (Theorem 4).

Theorem 12. *For each $n \geq 3$, there exists an instance with n agents with binary utility functions such that the EF1 exchange graph is disconnected.*

4.4 Identical Utilities

Let us now consider the case where the utilities are identical across agents, though they need not be binary. As with the case of binary utilities, there are instances in which the EF1 exchange graph is not connected even for three agents.

Theorem 13. *For each $n \geq 3$, there exists an instance with n agents with identical utility functions such that the EF1 exchange graph is disconnected.*

We end this section with a result that determining whether an optimal EF1 exchange path exists between two allocations is NP-hard even for four agents with identical valuations. This can be shown via a reduction from the NP-hard problem PARTITION.

Theorem 14. *Deciding the existence of an optimal EF1 exchange path between two EF1 allocations is NP-hard, even for $n = 4$ agents with identical utility functions.*

5 Conclusion and Future Work

In this paper, we have initiated the study of reachability problems in fair division by investigating the connectivity of the EF1 exchange graph and the optimality of EF1 exchange paths. While determining the existence of an EF1 exchange path between two given allocations is PSPACE-complete in general, an intriguing question is whether this can be done in polynomial time for two agents. For the negative results obtained in our paper, one could ask whether an (optimal or otherwise) exchange path between EF1 allocations exists if we allow the intermediate allocations to be *envy-free up to k goods (EF k)* for some small $k > 1$. Extending our results to fairness notions other than EF1 is also a meaningful direction. Finally, instead of exchanging goods between agents, one may also consider the setting where an agent *transfers* one good to another agent in each operation—in this case, the size of the allocation does not need to be fixed.

Acknowledgments

This work was partially supported by the Singapore Ministry of Education under grant number MOE-T2EP20221-0001, by JST PRESTO under grant number JPMJPR20C1, by JSPS KAKENHI under grant number JP20H05795, by JST ERATO under grant number JPMJER2301, and by an NUS Start-up Grant. We thank the anonymous reviewers for their valuable comments.

References

- Bei, X.; Igarashi, A.; Lu, X.; and Suksompong, W. 2022. The price of connectivity in fair division. *SIAM Journal on Discrete Mathematics*, 36(2): 1156–1186.
- Berendsohn, B. A.; Boyadzhyska, S.; and Kozma, L. 2022. Fixed-point cycles and approximate EFX allocations. In *Proceedings of the 47th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, 17:1–17:13.
- Bilò, V.; Caragiannis, I.; Flammini, M.; Igarashi, A.; Monaco, G.; Peters, D.; Vinci, C.; and Zwicker, W. S. 2022. Almost envy-free allocations with connected bundles. *Games and Economic Behavior*, 131: 197–221.
- Bonamy, M.; Bousquet, N.; Heinrich, M.; Ito, T.; Kobayashi, Y.; Mary, A.; Mühlenthaler, M.; and Wasa, K. 2019. The perfect matching reconfiguration problem. In *Proceedings of the 44th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, 80:1–80:14.
- Budish, E. 2011. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6): 1061–1103.
- Chaudhury, B. R.; Garg, J.; Mehlhorn, K.; Mehta, R.; and Misra, P. 2021. Improving EFX guarantees through rainbow cycle number. In *Proceedings of the 22nd ACM Conference on Economics and Computation (EC)*, 310–311.
- Freeman, R.; Micha, E.; and Shah, N. 2021. Two-sided matching meets fair division. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)*, 203–209.
- Gourvès, L.; Lesca, J.; and Wilczynski, A. 2017. Object allocation via swaps along a social network. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, 213–219.
- Holyer, I. 1981. The NP-completeness of some edge-partition problems. *SIAM Journal on Computing*, 10(4): 713–717.
- Huang, S.; and Xiao, M. 2020. Object reachability via swaps under strict and weak preferences. *Autonomous Agents and Multi-Agent Systems*, 34(2): 51:1–51:33.
- Igarashi, A.; Kamiyama, N.; Suksompong, W.; and Yuen, S. M. 2023a. Reachability of fair allocations via sequential exchanges. *CoRR*, abs/2312.07241.
- Igarashi, A.; Kawase, Y.; Suksompong, W.; and Sumita, H. 2023b. Fair division with two-sided preferences. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence (IJCAI)*, 2756–2764.
- Ito, T.; Demaine, E. D.; Harvey, N. J. A.; Papadimitriou, C. H.; Sideri, M.; Uehara, R.; and Uno, Y. 2011. On the complexity of reconfiguration problems. *Theoretical Computer Science*, 412(12–14): 1054–1065.
- Ito, T.; Kakimura, N.; Kamiyama, N.; Kobayashi, Y.; Nozaki, Y.; Okamoto, Y.; and Ozeki, K. 2023. On reachable assignments under dichotomous preferences. *Theoretical Computer Science*, 979: 114196.
- Johnson, M.; Kratsch, D.; Kratsch, S.; Patel, V.; and Paulusma, D. 2016. Finding shortest paths between graph colourings. *Algorithmica*, 75(2): 295–321.
- Li, F.; Plaxton, C. G.; and Sinha, V. B. 2021. Object allocation over a network of objects: Mobile agents with strict preferences. In *Proceedings of the 20th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 1578–1580.
- Lipton, R. J.; Markakis, E.; Mossel, E.; and Saberi, A. 2004. On approximately fair allocations of indivisible goods. In *Proceedings of the 5th ACM Conference on Electronic Commerce (EC)*, 125–131.
- Manurangsi, P.; and Suksompong, W. 2023. Differentially private fair division. In *Proceedings of the 37th AAAI Conference on Artificial Intelligence (AAAI)*, 5814–5822.
- Müller, L.; and Bentert, M. 2021. On reachable assignments in cycles. In *Proceedings of the 7th International Conference on Algorithmic Decision Theory (ADT)*, 273–288.
- Nishimura, N. 2018. Introduction to reconfiguration. *Algorithms*, 11(4): 52:1–52:25.
- Obraztsova, S.; Elkind, E.; and Faliszewski, P. 2020. On swap convexity of voting rules. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, 1910–1917.
- Obraztsova, S.; Elkind, E.; Faliszewski, P.; and Slinko, A. 2013. On swap-distance geometry of voting rules. In *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 383–390.
- Plaut, B.; and Roughgarden, T. 2020. Communication complexity of discrete fair division. *SIAM Journal on Computing*, 49(1): 206–243.
- Savitch, W. J. 1970. Relationships between nondeterministic and deterministic tape complexities. *Journal of Computer and System Sciences*, 4(2): 177–192.