Algorithmic Fairness in Sequential Decision-Making

Nicholas Teh
University of Oxford
Department of Computer Science

A thesis submitted for the degree of
Doctor of Philosophy in Computer Science

Trinity 2025

Abstract

Algorithms are increasingly entrusted with making critical decisions that in society, from assigning bandwidth on wireless networks to regularly curating recommended content for our social media feed. Ensuring fairness in sequential decision-making settings is challenging: decisions unfold over time, agents' preferences may evolve, and strategic behavior can distort intended guarantees. This thesis develops a theoretical foundation for the study of algorithmic fairness in sequential decision-making, with a particular focus on two canonical models—online resource allocation (Part I) and temporal voting (Part II).

In Part I, we study a model of indivisible items that arrive sequentially and must be allocated immediately and irrevocably to agents. In the absence of information about future arrivals, we prove strong impossibility results. However, we demonstrate that progressively richer predictive information can enable significantly fairer outcomes. These results underscore both the limitations imposed by uncertainty and the power of structured foresight in online fair division. We then consider a setting in which the entire sequence of future items is known in advance, but where fairness must be satisfied not only at the end but at every *prefix* of the allocation. We formalize the concept of cumulative fairness over time, which introduces novel computational challenges. Finally, we present the first axiomatic characterization of the widely studied maximum Nash welfare (MNW) rule in the offline binary setting. This result deepens our theoretical understanding of a rule that elegantly balances fairness and efficiency, and provides insights relevant to both offline and online allocation contexts.

In Part II, we turn to the related model of temporal voting, where a single project is selected in each round and voters accrue utility over time. We begin by analyzing welfare maximization objectives—both utilitarian and egalitarian—as well as standard proportionality guarantees, and examine their compatibility with manipulation-resistance. We then focus on stronger notions of proportional representation, adapted to the temporal setting, and study the computational complexity of verifying whether a given outcome satisfies these fairness guarantees. Finally, we investigate a con-

strained variant of temporal voting, which models scenarios where outcomes must be a permutation, such as in fair scheduling problems.

Across both settings, this thesis contributes: (i) novel fairness axioms for the sequential setting, (ii) algorithms with provable performance guarantees, (iii) complexity-theoretic results relating to the tractability of fairness objectives, and (iv) impossibility theorems that delineate the fundamental limits of achieving fairness in each setting. Collectively, these contributions advance our understanding of how to design systems that maintain fairness not just in isolated decisions, but across time.

Acknowledgments

I would like to thank my advisor, Edith Elkind. I have learned a great deal from her, not only on how to conduct rigorous and meaningful research, but also how to identify compelling questions and articulate clear motivations. She has given me the freedom to pursue the research directions that interest me most (along with many great ideas of her own), and introduced me to many wonderful people in the community. I am deeply grateful for her guidance, support, and generosity throughout my PhD.

I am also very thankful to Paul Goldberg for his consistent advice and support during my PhD. I appreciate his role as my examiner for various milestones along the way, and for stepping in at Oxford to meet with me regularly during the final year of my PhD. His willingness to engage with my work and offer thoughtful feedback, even on short notice, has been immensely helpful.

I have also learned a great deal from Warut Suksompong, and I've greatly enjoyed our collaborations. I'm especially grateful to him for inviting me back to Singapore so regularly. His astute intuition and keen eye for detail never fail to impress me. Working with him has helped me sharpen my own attention to detail and develop a more precise and thoughtful research approach. Our continued collaboration over the years has been a very rewarding part of my research journey, and I look forward to many more projects together.

I would also like to thank the examiners of this thesis, Elias Koutsoupias and Aris Filos-Ratsikas, for kindly agreeing to review my work.

I have also had the fantastic opportunity to work with many great collaborators during my PhD, including: Shiri Alouf-Heffetz, Niclas Boehmer, Florian Brandl, Laurent Bulteau, Davin Choo, Edith Elkind, Winston Fu, Jiarui Gan, Abheek Ghosh, Paul Goldberg, Matthias Greger, Ayumi Igarashi, Jian Zhang, Derek Khu, Sonja Kraiczy, Alexander Lam, Mohamad Latifian, Patrick Lederer, Eugene Lim, Yan Hao Ling, Tzeh Yuan Neoh, Svetlana Obraztsova, Jannik Peters, Bradley Phillips, Grzegorz Pierczyński, Tze-Yang Poon, Isaac Robinson, Jonathan Scarlett, Ulrike Schmidt-Kraepelin, Warut Suksompong, Stanisław Szufa, Nimrod Talmon, Giannis Tyrovolas, Tomasz Wąs, Valentin Zech, and Yair Zick.

During my time in Oxford, I've been fortunate to get to know many wonderful friends who have made my experience here truly unforgettable. Their support, laughter, and companionship have been invaluable—both within academia and beyond.

Special thanks to Abheek, Jiarui, Martin, Mohsen, Sonja, and Tomasz for making the office such a fun, lively, and welcoming place to work; to Charlotte, Denis, Ivin, Lucas, Nahshon, Shawn, and Zi Hao for all the great times in Oxford—going to formal dinner, playing mahjong, traveling, and simply enjoying each other's company; and especially to Ivin and Nahshon for the movie nights, shared meals, and endless laughter at home. Your friendship brought comfort, joy, and a deep sense of belonging—something I especially cherished during my final year.

Lastly, I'd like to express my deepest gratitude to my family—my mum; my late dad, who unexpectedly passed away last year and is dearly missed; and my sister—for their unwavering love, strength, and support. Their quiet sacrifices, endless patience, and steadfast belief in me have carried me through every stage of this journey.

Contents

Omitted Work

Some work conducted and published during my DPhil studies have been excluded from this thesis, to ensure coherence and succinctness. These are listed below.

- Fair Division
  - Maximum Weighted Nash Welfare for Binary Valuations [Suksompong and Teh, 2022]
  - Weighted Fair Division with Matroid-Rank Valuations [Suksompong and Teh, 2023]
  - Weighted Envy-Freeness for Submodular Valuations [Montanari et al., 2024]
  - Envy-Free House Allocation with Subsidy [Choo et al., 2024]
  - Fair Division of Chores with Budget Constraints [Elkind et al., 2024b]
  - Counting EFX Allocations [Neoh and Teh, 2025]
- Temporal Voting
  - Temporal Fairness in Multiwinner Voting [Elkind et al., 2024d]
  - Multiwinner Temporal Voting with Aversion to Change [Zech et al., 2024]
  - Temporal Voting Over Public Chores [Elkind et al., 2025b]
  - Issue-by-Issue Voting with Uncertainty [Alouf-Heffetz et al., 2022]
- Portioning
  - Portioning with Cardinal Preferences [Elkind et al., 2023]
- Algorithmic Mechanism Design
  - Fraud-Proof Revenue Division on Streaming Platforms [Ghosh et al., 2025]

Chapter 1

Introduction

Algorithms are increasingly entrusted with making consequential decisions that shape outcomes in every corner of society. From allocating ICU beds in overwhelmed hospitals and distributing municipal budgets across infrastructure projects, to curating the content we see on social media and regulating bandwidth on wireless networks, algorithmic systems now determine who gets what, when, and how. As these systems scale in scope and complexity, ensuring that their decisions are not only efficient but also fair has emerged as a central challenge at the intersection of computer science, economics, mathematics, and operations research.

But fairness is often not a singular, well-defined objective. Unlike classic goals such as maximizing throughput or minimizing cost, fairness is inherently multifaceted, context-sensitive, and often contested. What does it mean for an algorithm to be fair? How should fairness be formalized, measured, or achieved?

Fairness can come in many forms: *procedural fairness*, where the emphasis is on how decisions are made—whether the process is transparent, consistent, or accountable; *outcome fairness*, which focuses on what is received—e.g., whether resources are divided equitably or whether no one is unduly advantaged or harmed; *group fairness*, ensuring that protected groups are treated comparably, and *strategic fairness*, ensuring that individuals cannot manipulate the system for personal gain; along with many others. These perspectives reflect deep ethical, social, and operational values. In real-world systems, they underpin public trust, institutional legitimacy, and sustained engagement. Yet, fairness must also be translated into precise, testable, and enforceable algorithmic guarantees—a task that becomes even more complex in dynamic environments.

This thesis focuses on fairness in *sequential decision-making* settings, where decisions must be made repeatedly over time and outcomes for individuals can accumulate across rounds. Examples include: assigning tasks each day in a volunteer workforce, allocating cloud computing resources on demand, updating recommendation feeds, or selecting public projects for implementation every year.

In such settings, fairness cannot be assessed solely at a single point in time. A decision that appears fair in isolation may be deeply unfair when viewed across the long run. Thus, a central theme of this thesis is *temporal* fairness: how to ensure that fairness is maintained *over time*. This requires designing models and mechanisms that account for past decisions, evolving agent preferences, and the compounding nature of unfair treatment.

To investigate these questions, we focus on two foundational domains in algorithmic decision-making and computational social choice:

- **Online resource allocation**, where resources (modeled as indivisible items) arrive over time and must be allocated immediately to agents with heterogeneous preferences.

- **Temporal voting**, where collective decisions are made across rounds via collective approval ballots, and each round's outcome affects the utilities of (groups of) voters.

While static versions of both models have been extensively studied, this thesis centers on their sequential extensions. Across both domains, we introduce new models, fairness axioms, algorithmic techniques, and complexity-theoretic results that provide a deeper understanding of what fairness can mean—and how it can be achieved—when decisions must be made sequentially.

Online Resource Allocation

In the first part of this thesis, we examine the problem of *online fair division* with indivisible items. Fair division is the study of how to allocate resources among agents in a manner that is considered fair, particularly when agents have heterogeneous preferences. Since the foundational work of Steinhaus [1948], which introduced a mathematical framework for fair division, the field has grown to encompass a rich landscape of fairness notions and algorithmic procedures applicable to a wide range of allocation settings [Brams and Taylor, 1996, Robertson and Webb, 1998, Moulin, 2003, 2019]. Applications of fair division are wide-ranging, from allocating medical supplies to communities or schoolteachers to primary schools, to dividing assets in a divorce settlement and usage rights of a jointly invested facility.

While much of the existing literature focuses on the offline setting, where all resources are known in advance and a single fair allocation must be computed, many real-world scenarios require making decisions in an online manner. In such settings, items arrive sequentially over time, and must be allocated immediately and irrevocably upon arrival.

Applications of such an online model are wide ranging, and include tasks such as the allocation of resources in cloud computing environments [Bei et al., 2022], assigning donations to food banks [Aleksandrov et al., 2015], allocation of impressions in recommendation systems [Murray et al., 2020], or assigning content moderation tasks on social media platforms [Allouah et al., 2023]. Variants of this model also coincide with commonly-studied problems in *online scheduling* (such as the *semi-online machine covering* problem [Ebenlendr et al., 2006, Wu et al., 2007] and the *semi-online load balancing* problem [Angelelli et al., 2007, Cheng et al., 2005, Lee and Lim, 2013]). The online nature of item arrivals introduces significant challenges. Since future items are unknown at the time of decision-making, and agent preferences may evolve over time, classic fairness guarantees that are feasible in offline settings often become infeasible online. This raises natural questions about what fairness can mean in an online context, and whether any useful guarantees can still be achieved.

In **Chapter 3**, which is based on a paper currently under submission, we study an online fair division model with a fixed set of agents and a sequence of indivisible goods arriving over time. Each item must be immediately and irrevocably assigned to one of the agents. Our goal is that the final allocation—once all items have been assigned—is (approximately) *fair*. We investigate to what extent classical fairness notions can still be achieved in this setting. When these guarantees are provably impossible in the worst case, we explore whether *additional information*—such as partial knowledge about the sequence of future items—can help in designing more equitable algorithms.

In **Chapter 4**, which is based on a paper published at AAMAS 2025 [Elkind et al., 2025a], we consider the informed online fair division setting, where the algorithm is given the full sequence of item values and their arrival order in advance. Although such complete information would seem to reduce the problem to the offline setting—where approximate fairness guarantees are well understood—we instead ask for stronger fairness guarantees that must hold at every prefix of the item sequence. This ensures that agents are not unfairly disadvantaged at any intermediate stage of the process. We introduce new temporal fairness concepts tailored to this setting and analyze their algorithmic and computational properties.

In addition to the two chapters above that form the backbone of the first part of this thesis, we also include a standalone result in the offline fair division setting—one that is also closely connected to the growing literature on online fair division. In **Chapter 5**, which is based on a paper currently under submission, we study the popular maximum Nash welfare (MNW) rule for the offline setting. While MNW is well-known for achieving an appealing balance between fairness and efficiency, our work provides the first axiomatic characterization of MNW in the offline binary setting, within the space of all allocation rules. This result deepens the theoretical foundations of a fairness rule that has been influential in both offline and online contexts. In doing so, it offers insights that may inform the design and justification of fair and efficient algorithms in the online setting as well.

Temporal Voting

In the second part of this thesis, we study the closely related model of *temporal voting*, a sequential variant of the classic multiwinner voting framework with approval ballots. Instead of selecting a fixed committee or set of projects all at once, temporal voting operates over multiple rounds, selecting one project in each round based on approval ballots from voters. Each selected project provides utility to the voters, and the goal is to ensure *fair* or *representative* outcomes across the entire sequence.

In *multiwinner voting*, is typically to select a representative committee of candidates that reflects the preferences of the electorate. Applications span from electing diverse panels and boards to ensuring the presence of minority voices in political institutions [Lackner and Skowron, 2023, Phragmén, 1895]. More recently, multiwinner voting has found relevance in machine learning (ML) systems that incorporate human feedback, optimize collective utility, or aim for fairness across diverse user groups. For example, it has been applied to enhancing diversity in recommendation systems and social media feeds [Gawron and Faliszewski, 2024, Revel et al., 2025, Streviniotis and Chalkiadakis, 2022], supporting fair decision-making in blockchain governance [Boehmer et al., 2024, Burdges et al., 2020, Cevallos and Stewart, 2021], and providing representation guarantees in LLM-assisted democratic processes [Boehmer et al., 2025, Fish et al., 2024].

However, many real-world ML applications go beyond static selection tasks and involve decision-making over time, with preferences that may evolve across rounds. For instance, curriculum learning constructs sequences of training tasks for reinforcement learning agents; temporal fairness guarantees can prevent the curriculum from overemphasizing early-stage objectives at the expense of later goals [Wu et al., 2024, Yang et al., 2021]. Similarly, streaming service catalogs and content recommendation systems periodically refresh their offerings: ensuring proportionality across time ensures that under-served genres or user communities are not systematically sidelined across updates [Chen et al., 2024]. In the domain of generative AI, blending outputs from multiple models—each with its own inductive biases—calls for proportional merging strategies that preserve diversity of generated content over time [Peters, 2024].

These scenarios demand fairness guarantees not just within a single decision round, but across the entire time horizon. This motivates the adaptation of temporal analogues of classical multiwinner voting axioms to the temporal setting.

We begin by exploring foundational computational questions in the temporal voting setting—mirroring those traditionally studied in social choice. In **Chapter 6**, which is based on work published at AAAI 2024 [Neoh and Teh, 2024] and ECAI 2024 [Elkind et al., 2024c], we study how to maximize utilitarian and egalitarian welfare objectives, along with a standard proportionality notion inspired by fair division and public goods allocation. We also investigate the compatibility of these welfare and fairness objectives with strategic manipulation: when agents can misreport their approvals to benefit themselves, what guarantees can be preserved?

In **Chapter 7**, which is based on work published at AAAI 2025 [Elkind et al., 2025c], we turn our attention to stronger forms of proportional representation—closer in spirit to traditional multiwinner voting notions (i.e., *justified representation* and its variants). We study the verification problem for extended proportionality axioms in the temporal setting, establishing numerous complexity results. While verification is computationally harder in the temporal context than in static multiwinner voting, we identify natural structural assumptions that yield efficient algorithms.

Finally, in **Chapter 8**, based on work published at SAGT 2023 [Elkind et al., 2022], we consider a restricted setting where the outcome must be a permutation, capturing scheduling constraints. This variant models settings where projects or tasks must be ordered in time, and fairness must be maintained across the scheduled sequence. We study how fairness can be maintained under these additional structural constraints.

Chapter 2

Preliminaries

This chapter introduces some definitions and notation that we will use across multiple chapters. Preliminaries specific to a single chapter are presented in the chapter itself.

We assume that the reader is familiar with basic notions of classic complexity theory [Papadimitriou, 2007] and parameterized complexity [Flum and Grohe, 2006, Niedermeier, 2006].

For each positive integer k, let $[k] := \{1, \ldots, k\}$.

2.1 Online Resource Allocation

2.1.1 Model

In this section, we introduce the basic definitions and notation for the resource allocation setting, which we will study in Chapters 3–4.

We assume to be given a set $N = [n]$ of (fixed) *agents* and a set $G = \{g_1, \ldots, g_m\}$ of m *goods* arriving online. We label the goods $g_1, \ldots, g_m$ in the order that they arrive. Each agent $i \in N$ has a non-negative *valuation function* $v_i \colon 2^G \to \mathbb{R}_{\geq 0}$. The list $\mathbf{v} = (v_1, \ldots, v_n)$ is called a *valuation profile.* We write v instead of v_i when all agents have identical valuation functions. As with most works in the fair division literature, we assume that valuation functions are *additive*, i.e., for any subset of goods $S \subseteq G$, $v_i(S) = \sum_{g \in S} v_i(\{g\})$. For convenience, we write $v_i(g)$ instead of $v_i(\{g\})$ for a single good g. Every subset of goods in G can also be referred to as a *bundle.* An *allocation* $\mathcal{A} = (A_1, \ldots, A_n)$ is a *partition* of the goods into bundles, with agent $i \in N$ receiving bundle A_i; let $\Pi_n(G)$ denote the set of all possible allocations.

2.1.2 Fairness

One of the most common fairness notions considered in fair division is *envy-freeness* (EF): every agent's value for their own bundle should be at least as much as their value for any other agent's bundle, that is, $v_i(A_i) \geq v_i(A_j)$ for all $i, j \in N$.. However, existence of EF allocations cannot be guaranteed, even in the simplest case of two agents and a single good. Thus, we consider the two most common relaxations of EF. The first is a relatively strong and widely studied EF relaxation: *envy-freeness up to any good.*

Definition 2.1.1 (EFX)**.** An allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is *envy-free up to any good (EFX)* if for every pair of agents $i, j \in N$ and every $g \in A_j$, we have that $v_i(A_i) \geq v_i(A_j \setminus \{g\})$.

In the offline setting, the existence of EFX allocations for more than three agents is still an open problem, and remains an important open question [Procaccia, 2020]. As such, many works focus on a, relatively weaker but still natural variant of envy-freeness: *envy-freeness up to one good.*

Definition 2.1.2 (EF1)**.** An allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is *envy-free up to one good (EF1)* if for every pair of agents $i, j \in N$ with $A_j \neq \emptyset$ there exists a $g \in A_j$ such that $v_i(A_i) \geq v_i(A_j \setminus \{g\})$.

It is easy to see that EFX implies EF1. Furthermore, there are several algorithms that produce an EF1 allocation, such as the envy-cycle elimination algorithm [Lipton et al., 2004] or the classic round-robin procedure.

We also consider a weaker (compared to EF) fairness property, called *proportionality* (PROP), which requires that each agent receives her "proportional share"—that is, at least a $1/n$-th fraction of the total value of all goods according to her own valuation [Steinhaus, 1948]. Numerous prior works have looked into this fairness concept and its relaxed variants in the offline setting [Amanatidis et al., 2023, Aziz et al., 2023a]. Similar to the case of EF, a proportional allocation is not always guaranteed to exist, even with just two agents and a single good/round. We focus on the analogous "up to one good" relaxation commonly considered in the literature.[1]

Definition 2.1.3 (PROP1)**.** An allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is *proportional up to one good (PROP1)* if for every agent $i \in N$, either $A_i = G$ or there exists a $g \in G \setminus A_i$ such that $v_i(A_i \cup \{g\}) \geq \frac{v_i(G)}{n}$.

[1]A stronger variant of PROP1 would be proportionality *up to any good* (PROPX). However, it is not always satisfiable even in the single-shot setting [Aziz et al., 2020].

It is easy to see that EF1 implies PROP1. Moreover, while EF and PROP are equivalent in the case of $n = 2$, the same relationship cannot be established for EFX, EF1, or PROP1.

The last fairness notion we consider in this work is *maximin share fairness* (MMS), which was also the focus of Zhou et al. [2023] in the online setting. Intuitively, MMS guarantees that each agent receives a bundle she values at least as much as she would have gotten if she were allowed to partition all goods into n bundles and then receive the least valuable bundle (according to her own valuation).

Definition 2.1.4 (MMS)**.** Let $\Pi(G)$ be the set of all n-partition of G. The *maximin share* of each agent $i \in N$ is defined as $\mathit{MMS}_i := \max_{\mathcal{X}\in\Pi(G)} \min_{j\in N}\{v_i(X_j)\}$. Then, an allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is maximin share fair (MMS) if $v_i(A_i) \geq \mathit{MMS}_i$ for all $i \in N$.

While PROP implies MMS, MMS implies PROP1 [Caragiannis et al., 2025]. Finally, we also study approximate versions of the properties defined above.[2] For $\alpha \in [0, 1]$, we say that an allocation $\mathcal{A}$ is:

- α-EFX if for every pair $i, j \in N$ and every $g \in A_j$, we have that $v_i(A_i) \geq \alpha \cdot v_i(A_j \setminus \{g\})$;
- α-EF1 if for every pair $i, j \in N$, there exists a $g \in A_j$ such that $v_i(A_i) \geq \alpha \cdot v_i(A_j \setminus \{g\})$;
- α-PROP1 if for every $i \in N$, either $A_i = G$ or there exists a $g \in G \setminus A_i$ such that $v_i(A_i \cup \{g\}) \geq \alpha \cdot \frac{v_i(G)}{n}$;
- α-MMS if for every $i \in N$, $v_i(A_i) \geq \alpha \cdot \mathsf{MMS}_i$.

Note that for any $\alpha \in [0, 1]$, α-EFX implies α-EF1, and for any n, EF1 implies $\frac{1}{n}$-MMS [Amanatidis et al., 2018, Segal-Halevi and Suksompong, 2019]. We equivalently say that a rule satisfies the property if any allocation returned by the rule on a problem instance satisfies the the property. Note that when $\alpha = 1$, the property is satisfied exactly and the lower $\alpha \in [0, 1]$ is, the worse the approximation gets.

[2] We consider *multiplicative* approximations of these fairness properties, which is a well-established and common approach in fair division (as compared to *additive* approximations).

2.1.3 Efficiency

It is easy to achieve most forms of fairness by simply refraining from allocating any items to any agent; however, such an outcome is clearly undesirable from a practical standpoint. To avoid such trivial solutions, most of the literature incorporates some notion of *efficiency* to ensure that the allocation meaningfully utilizes the available resources. The most common (often implicit) assumption is *completeness*, which requires that all goods be fully allocated among the agents. This assumption plays a particularly important role in the online setting, where goods must be allocated immediately and irrevocably as they arrive.

Beyond completeness, we also consider a stronger notion of efficiency popular in the economics literature: *Pareto-optimality.* This concept captures the idea that no agent's outcome can be improved without making another agent worse off. Formally, it is defined as follows.

Definition 2.1.5 (Pareto-optimality)**.** We say that an allocation $\mathcal{A}$ is *Pareto-optimal (PO)* if there does not exist another allocation $\mathcal{A}'$ such that for all $i \in N$, $v_i(A_i') \geq v_i(A_i)$, and for some $j \in N$, $v_j(A_j') > v_j(A_j)$. If such an allocation $\mathcal{A}'$ exists, we say that $\mathcal{A}'$ *Pareto-dominates* $\mathcal{A}$.

2.2 Temporal (Approval) Voting

In this section, we introduce the basic definitions and notation for the temporal voting setting, which we will study in Chapters 6–8.

Let $N = [n]$ be a set of n *agents*, let $P = \{p_1, \ldots, p_m\}$ be a set of m *projects* (or *candidates*), and let ℓ denote the number of *timesteps* (or *rounds*). For each $t \in [\ell]$, the set of projects approved by agent $i \in N$ at timestep t is captured by her *approval set* $s_{i,t} \subseteq P$. The approval sets of agent i are collected in her *approval vector* $\mathbf{s}_i = (s_{i,1}, \ldots, s_{i,\ell})$. Thus, a *temporal election* (or an *instance* of our problem) is a tuple $(N, P, \ell, (\mathbf{s}_i)_{i \in N})$. For each $p \in P, t \in [\ell]$ we denote by n_{pt} the number of agents in N that approve project p at timestep t.

An *outcome* of a temporal election $(P, N, \ell, (\mathbf{s}_i)_{i \in N})$ is a sequence $\mathbf{o} = (o_1, \ldots, o_\ell)$ of ℓ candidates such that for every $t \in [\ell]$ candidate $o_t \in P$ is chosen in round t. Unless otherwise stated, a candidate may be selected multiple times, i.e., it may be the case that $o_t = o_{t'}$ for $t \neq t'$.

The *utility* (or equivalently, *satisfaction*) of an agent $i \in N$ for an outcome $\mathbf{o}$ is given by $u_i(\mathbf{o}) = sat_i(\mathbf{o}) = |\{t \in [\ell] : o_t \in s_{i,t}\}|$.[3]

We also let $\Pi(\mathcal{I})$ denote the space of all possible outcomes for an instance $\mathcal{I}$.

[3]We choose to use both terms, even though they refer to the same concept, in order to remain consistent with the standard terminology in fair division and multiwinner voting, respectively.

Part I

Online Resource Allocation

Chapter 3

Online Fair Division with Additional Information

3.1 Overview

We begin by studying the problem of *online fair division* with indivisible *goods*.[1] In this setting, we are given a fixed set of agents together with a sequence of indivisible goods arriving sequentially. After each good arrives, it needs to be irrevocably allocated to an agent. Our goal is that once all goods have been assigned, the corresponding assignment should be *fair* to the agents.

We investigate whether traditional fairness notions that are well-studied in the classic single-shot fair division setting can still be satisfied in our online setting; and if not, whether the presence of *additional information* can aid in the design of fair algorithms. An ideal online algorithm would not require any information about upcoming goods at any round of the sequential process. However, this can make it impossible to satisfy fundamental fairness criteria, even in scenarios as simple as those involving only two agents [He et al., 2019, Zhou et al., 2023]. Prior work in this area circumvent respective impossibility results in their model by assuming that the preferences of agents are *normalized*, or equivalently, with additive preferences that the total value for the whole set of goods by each agent is known in advance [Barman et al., 2022, Gkatzelis et al., 2021, Zhou et al., 2023]. This, in spirit, is related to the field of *online algorithms with advice* (refer to a survey by Boyar et al. [2017]). For instance, our problem has a close relation to the classic *semi-online scheduling* problem whereby heterogeneous jobs arrive in an online manner and need

[1] We focus on items with nonnegative valuations, which is a major foundational focus in the fair division literature. While many of our results would extend to chores (i.e., items with nonpositive valuations), we restrict our attention to goods to maintain narrative clarity and focus.

to be scheduled on homogeneous machines with the goal to minimize the maximum load on any machine (refer to a survey by Dwibedy and Mohanty [2022]). In this setting, it is typically assumed that some additional *offline* (future) information is known about the instance. Examples of such information include the total processing time of all jobs [Albers and Hellwig, 2012, Kellerer et al., 2015], the ratio between the smallest and largest processing time of any job [He and Zhang, 1999], or that the jobs arrive in decreasing order of processing time [Seiden et al., 2000].

In a similar vein, Zhou et al. [2023] initiated the study of semi-online fair division under the *normalized valuations* assumption (i.e., algorithms are given information—about the sum of the total value of the goods for each agent—before goods arrive); they focus on the *maximin share* (MMS) fairness property and achieving approximations of it given this normalization assumption. In the absence of any information on future goods, they showed that even in the case of two agents, no multiplicative approximation to MMS is possible. On the other hand, if normalized valuations are assumed, a 0.5-approximation to MMS can be found in instances with two agents (although the case of three or more agents still remains impossible).

We take this as a starting point and initiate the study of other commonly study fairness axioms—on top of MMS—such as *envy-freeness* (EF) and *proportionality* (PROP) (and their popular relaxations—*envy-freeness up to any good* (EFX), *envy-freeness up to one good* (EF1), and *proportionality up to one good* (PROP1)). We not only consider normalized valuations, but stronger predictive information (such as *frequency predictions*, inspired by various lines of work in the online algorithms literature), and ask for which *semi-online* settings one can achieve these (approximate) fairness properties.

In Section 3.3, we begin by considering the online fair division model *without* additional information about future goods. We show that even with two agents, no algorithm can achieve any positive EF1 approximation. We use a result by Benadè et al. [2018] to show that no online algorithm can achieve PROP1 even in the case of two agents.[2]

Given these negative results, in Section 3.4, we turn to a model where the online algorithm has access to the sum of agents' valuations over all goods that will arrive (which is equivalent to assuming that agents' valuations are normalized), a fairly common assumption in prior works on online fair division. In this setting, we generalize (and modify) an algorithm by Zhou et al. [2023] that guarantees 0.5-MMS

[2] En-route, we point out a mistake in the proof Kahana and Hazon [2023] asserting the same claim.

for two agents (but not EF1, as we will show), by proposing an algorithm that always returns an allocation satisfying (i) PROP1 for any number of agents, and (ii) EF1 and 0.5-MMS simultaneously for two agents. We complement this by showing that for three or more agents, no online algorithm can achieve any positive approximation to EF1. We show a similar impossibility result with respect to EFX for two agents.

To motivate our investigation into whether stronger fairness guarantees can be achieved by providing the algorithm with richer information, in Section 3.5, we consider a model where the online algorithm has access to *frequency predictions* (i.e., for each agent and value, a predictor tells us the frequency with which this value will appear among the agents' valuations for items that will arrive). This is motivated by similar notions in the online knapsack [Im et al., 2021], online bin packing [Angelopoulos et al., 2023], and online matching [Mehta et al., 2007] literature. Here, we prove a general result: any *share-based* fairness notion that can be achieved by an offline algorithm in the single-shot setting[3] can also be achieved by an online algorithm with frequency predictions—we constructively design a meta-algorithm that gives this guarantee. In particular, this implies that the currently best known $\left(\frac{3}{4} + \frac{3}{3836}\right)$-MMS approximation guarantee can also be obtained in our setting using frequency predictions for any number of agents, thereby improving on the (tight) 0.5-MMS guarantee under normalized valuations established by Zhou et al. [2023]. We extend this framework to show that it is possible to achieve EFX for two agents, and complement this by showing that for three or more agents, any positive approximation is still impossible.

Finally, motivated by the relation of our problem to online scheduling (where many works consider the setting with identical machines [Dwibedy and Mohanty, 2022]), we study the setting where agents have identical valuation functions (i.e., each good is valued the same by every agent). In this setting, an existing algorithm implies that EF1 can be achieved without any information about the future. We complement this by showing that any positive EFX approximation remain impossible for (i) two agents without any information, and (ii) three agents or more given normalization information. For two agents given the normalization information, we provide an algorithm that returns a $\frac{\sqrt{5}-1}{2}$-EFX allocation, and show that this bound is tight (i.e., no better approximation to EFX is possible).

[3]Some examples studied in the literature include *round robin share* (RRS) [Conitzer et al., 2017, Gourvès et al., 2021], *minimum EFX share* (MXS) [Caragiannis et al., 2025], and $\left(\frac{3}{4} + \frac{3}{3836}\right)$-MMS [Akrami and Garg, 2024].

3.2 Preliminaries

As is standard in online fair division literature, we assume an adversarial model where an adversary both constructs the instance and controls the sequence of arriving goods. As we consider deterministic online algorithms (again, another standard consideration), the adversary may be adaptive, choosing each item's valuation profile based on the algorithm's prior allocation decisions. The algorithm is given the number of agents n, along with some information (if any), which depends on the setting we are considering. When a good g arrives, it observes all $\{v_i(g)\}_{i \in N}$ and must immediately and irrevocably assign g to an agent.[4] We then measure performance of the algorithm by the *competitive ratio*, i.e., the worst-case approximation factor of the algorithm (with respect to the fairness objective) of the final allocation $\mathcal{A}$ over all online instances.

3.3 Impossibilities Without Information

We begin by considering the basic setting of online fair division *without* any information, that is, goods arrive one at a time and must be irrevocably allocated to agents, with no knowledge of future items. Zhou et al. [2023, Theorem A.1] proved that in the absence of any information, there does not exist any online algorithm with a competitive ratio strictly larger than 0 with respect to MMS, even for $n = 2$. Here, we consider (the approximate variants of) two other commonly studied fairness concepts in the fair division literature—envy-freeness and proportionality—in the same setting.

We begin with envy-freeness. Benadè et al. [2018] showed that no online algorithm, in the absence of information, can achieve EF1 (and hence EFX). We strengthen this by proving a stronger impossibility result: no online algorithm in this setting can guarantee *any* positive approximation to EF1.

Proposition 3.3.1. *For $n \geq 2$, without future information, there does not exist any online algorithm with a competitive ratio strictly larger than* 0 *with respect to* EF1*.*

Proof. Suppose for a contradiction that there exists a γ-competitive algorithm for approximating EF1 without information, for some $\gamma \in (0, 1]$. Let the first two goods

[4]This corresponds to the notion of *completeness*, which is also a standard assumption in almost all prior work on online fair division, and most prior work in the classic fair division model. An exception in the latter is the setting involving the allocation of goods *with charity*, where an objective is to minimize the number of unallocated goods (or those given to charity), on top of ensuring an approximately fair allocation.

be g_1, g_2 with values $v_i(g_1) = 1$ for all $i \in N$, $v_1(g_2) = K$, and $v_j(g_2) = \frac{1}{K}$ for all $j \in N \setminus \{1\}$ and for some constant $K > 1$ Suppose w.l.o.g. that g_1 is allocated to agent 1. If g_2 is also allocated to agent 1, let the next (and final) good be g_3 with $v_i(g_3) = 0$ for all $i \in N$. Consequently, we get that $v_2(A_2) = 0$ and $v_2(A_1 \setminus \{g_1\}) = \frac{1}{K}$ and we get $\gamma \leq 0$ for any $K > 1$. Thus, g_2 cannot be allocated to agent 1. W.l.o.g., let it be allocated to agent 2. Then, let the remaining $n-1$ goods be $g_3, \ldots, g_{n+1}$ with valuations as follows:

$\mathbf{v}$	g_1	g_2	g_3	$\cdots$	g_{n+1}
1	1	K	K	$\ldots$	K
2	1	$\frac{1}{K}$	K	$\ldots$	K
3	1	$\frac{1}{K}$	0	$\ldots$	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ldots$	$\vdots$
n	1	$\frac{1}{K}$	0	$\ldots$	0

We split our analysis into two cases.

Case 1: Agent 2 is allocated at least one good in $\{g_3, \ldots, g_{n+1}\}$. If agent 1 is allocated nothing from $\{g_3, \ldots, g_{n+1}\}$, then $v_1(A_1) = v_1(g_1) = 1$ and $v_1(A_2 \setminus \{g\}) \geq K$ for any $g \in A_2$. This gives us $\gamma \leq \frac{1}{K} \to 0$ as $K \to \infty$, a contradiction. Thus, if agent 2 is allocated at least one good in $\{g_3, \ldots, g_{n+1}\}$, then agent 1 must also allocated at least one good in $\{g_3, \ldots, g_{n+1}\}$. However, this means there are $n-3$ remaining goods for $n-2$ agents in $N \setminus \{1, 2\}$. As such, some agent $i \in N \setminus \{1, 2\}$ must receive no goods, leaving us with a 0-EF1 allocation, a contradiction.

Case 2: Agent 2 is not allocated any good in $\{g_3, \ldots, g_{n+1}\}$. There must be some agent $i \in N \setminus \{2\}$ receiving at least two goods eventually (i.e., $|A_i| \geq 2$). Then, $v_2(A_2) = v_2(g_2) = \frac{1}{K}$ and $v_2(A_i \setminus \{g\}) \geq 1$ for $g \in \arg\max_{g' \in A_i} v_2(g')$. This gives us $\gamma \leq \frac{1}{K} \to 0$ as $K \to \infty$, a contradiction.

In both cases, we arrive at a contradiction, thereby proving our claim. □

Given this, a natural follow-up question is whether it is possible to approximate a weaker fairness notion in this setting. Prior work by Kahana and Hazon [2023, Thm. 13] claims that no algorithm can satisfy PROP1 in this setting. We show that their example used to disprove the existence of an algorithm that returns a PROP1 allocation is incorrect.

Example 3.3.2 (Kahana and Hazon [2023]). Consider an instance with $n = 2$ agents and let goods arrive in the order $g_1, \ldots, g_m$. Also consider some round t where g_t arrives. If t is odd, then let $v_1(g_t) = v_2(g_t) = 1$. If t is even and the good g_{t-1} was allocated to agent 1, then let $v_1(g_t) = 1$ and $v_2(g_t) = 0.875$; symmetrically, if g_{t-1} was allocated to agent 2, then let $v_1(g_t) = 0.875$ and $v_2(g_t) = 1$.

Kahana and Hazon [2023] claimed that no online algorithm can achieve PROP1 in the above instance. However, observe that a simple online algorithm that always assigns two goods to the first agent, then two goods to the second agent, and so on, in an alternating fashion would satisfy PROP1. Their proof fails in the induction, in which the base case (for $t = 0$) is not true. Nevertheless, we are able to reason about the non-existence of a PROP1 allocation from the following result by Benadè et al. [2018].

Theorem 3.3.3 (Benadè et al. [2018]). *Given an instance with two agents, T goods, and $v_i(g) \in [0, 1]$ for any agent i and good g, there is no online algorithm with* $\max_{i \in \{1,2\}}\{v_i(A_j) - v_i(A_i)\} \in \Omega(\sqrt{T})$.

Using the definition of PROP1, together with the facts that $v_i(G) = v_i(A_i) + v_i(A_j)$ for any $i \in N$ and the maximum value of any single good is 1, it is easy to see that for any sufficiently large T, a PROP1 allocation may not exist. However, the problem of approximating PROP1—and establishing a lower bound on the competitive ratio of any online algorithm—remains elusive. We therefore leave this as a compelling but challenging open problem.

3.4 Normalization Information

In the previous section, we observed that even a weak (but commonly studied) fairness notion —PROP1— cannot be guaranteed without any information about future goods. This naturally raises the question: if we are given *some* information about future goods, can we obtain better approximation guarantees, or perhaps even satisfy certain fairness properties in the online setting?

Here, we begin to address this question by considering a very common kind of information studied in the literature (both in the online fair division setting, and in other contexts): *normalization* information. For this, we assume that the agents' valuations over all goods sum to 1 (in the online setting with additive valuations, this is, without loss of generality, equivalent to knowing the sum of valuations for each agent over all goods). This assumption is fairly common in the online fair division

Algorithm 1: EF1 for $n = 2$ and PROP1 for $n \geq 2$ given normalization information

Initialize the empty allocation $\mathcal{A}$ where $A_i = \varnothing$ for all $i \in N$, the set of active agents $N' \leftarrow N$, and $\mathbf{x} = (x_1, \ldots, x_n)$
while there exists a good g that arrives **do**
  if $N' = \varnothing$ **then**
    Allocate all remaining incoming goods to agent 1
  end if
  for each agent $i \in N$ **do**
    $x_i \leftarrow v_i(A_i) + \frac{n-1}{n} \cdot \max_{g' \in \bigcup_{j \in N \setminus \{i\}} A_j} \{v_i(g'), v_i(g)\}$
    if $x_i \geq 1/n$ **then**
      $N' \leftarrow N' \setminus \{i\}$ // Set i as inactive
    end if
  end for
  Let $i \in \arg\max_{j \in N'} v_j(g)$ (breaking ties in favor of the agent $j \in N'$ with the smallest x_j)
  $A_i \leftarrow A_i \cup \{g\}$
end while
return $\mathcal{A} = (A_1, \ldots, A_n)$

literature, and was studied by Zhou et al. [2023] for MMS, Banerjee et al. [2022] and Gkatzelis et al. [2021] in the divisible goods setting, and Banerjee et al. [2023] in the public goods setting.

In our setting, Zhou et al. [2023] provided an algorithm for computing a 0.5-MMS allocation for $n = 2$, but showed that no online algorithm can have a positive competitive ratio for MMS for $n \geq 3$. Here, we generalize and modify the algorithm of Zhou et al. [2023, Algorithm 1] and show that the allocation returned by our algorithm satisfies EF1 for two agents (and hence 0.5-MMS for two agents), and PROP1 for any number of agents.[5] We further note that the algorithm of Zhou et al. [2023] does not satisfy PROP1 (and therefore also not EF1) even for the case of two agents.[6]

We first show that Algorithm 1 returns an EF1 allocation in the case of two agents.

Theorem 3.4.1. *For $n = 2$, given normalization information, Algorithm 1 returns an* EF1 *allocation.*

[5] Note that PROP1 is strictly weaker than (and hence not equivalent to) EF1, even when $n = 2$ [Aziz et al., 2023a].

[6] To see this, consider an instance with six goods and two agents with identical valuations. Let the agents value the goods at $1/4 - \varepsilon, \varepsilon, 3/16, 3/16, 3/16, 3/16$ (in order of arrival). The algorithm of Zhou et al. [2023] could assign the first and second good to the first agent and the last four goods to the second agent leading to a PROP1 violation of the first agent, as $1/4 + 3/16 < 1/2$.

Proof. Let $N = \{1, 2\}$. We first prove that at the end of the algorithm, no inactive agent envies another agent. Note that after the algorithm has terminated (i.e., all goods have appeared and been allocated), for any inactive agent $i \notin N'$ (let the other agent be $j \in \{1, 2\} \setminus \{i\}$), if $G = A_i$, then $v_i(A_i) = 1 \geq v_i(A_j)$ and we trivially get our claim. Thus, assume $A_i \subset G$. Then, Lines 7 and 8 of the algorithm prescribes that there exists some good $g \in G \setminus A_i$ such that $v_i(A_i) + \frac{n-1}{n} \cdot v_i(g) \geq \frac{1}{n}$ (or equivalently, since $n = 2$, $v_i(A_i) + \frac{1}{2} \cdot v_i(g) \geq \frac{1}{2}$). Then, multiplying both sides of the inequality by 2 and performing algebraic manipulation,

$$v_i(A_i) \geq 1 - v_i(A_i) - v_i(g) = v_i(A_j \setminus \{g\}),$$

where $g \in G \setminus A_i = A_j$ and $1 - v_i(A_i) = v_i(A_j)$, by completeness and the normalization information assumption, respectively. Thus, no inactive agent will envy the other agent.

It remains to prove that after the algorithm has terminated, there are no more active agents, i.e., $N' = \varnothing$. Suppose for a contradiction that when the algorithm has terminated and all goods have been allocated, there are still active agents, i.e., $|N'| \geq 1$. Consider such an agent $k \in N'$ (and let the other agent be $k' \in \{1, 2\} \setminus \{k\}$.

Note that $v_k(A_k) < \frac{1}{2}$ (otherwise k would have been set as inactive). We split our analysis into two cases, depending on whether k' is active or inactive after the algorithm has terminated.

Case 1: $k' \in N'$. Since agent k' is also active, we have that $v_{k'}(A_{k'}) < \frac{1}{2}$, by Lines 7 and 8 of the algorithm. This means that

$$v_1(A_1) + v_2(A_2) < 1. \tag{3.1}$$

Moreover, since both agents are active, by Line 12 of the algorithm, each good is allocated to the agent that values it at least as much as the other agent, i.e., for each good $g \in A_1$, $v_1(g) \geq v_2(g)$ and for each good $g' \in A_2$, $v_2(g') \geq v_2(g')$. Summing over both sides of each inequality with goods in A_1 and A_2 respectively, we get

$$\sum_{g \in A_1} v_1(g) \geq \sum_{g \in A_1} v_2(g) \quad \text{and} \quad \sum_{g' \in A_2} v_2(g') \geq \sum_{g' \in A_2} v_1(g').$$

This is equivalent to

$$v_1(A_1) \geq v_2(A_1) \quad \text{and} \quad v_2(A_2) \geq v_1(A_2).$$

By the normalization information and completeness assumptions, we know that $v_2(A_1) = 1 - v_2(A_2)$ and $v_1(A_2) = 1 - v_1(A_1)$. Substituting this into the inequalities above, we get that

$$v_1(A_1) \geq 1 - v_2(A_2) \quad \text{and} \quad v_2(A_2) \geq 1 - v_1(A_1),$$

Combining these two inequalities, we get

$$v_1(A_1) + v_2(A_2) \geq 1 - v_2(A_2) + 1 - v_1(A_1),$$

which simplifies to

$$v_1(A_1) + v_2(A_2) \geq 1,$$

contradicting (3.1).

Case 2: $k' \notin N'$**.** Since agent k' is inactive, we have $v_{k'}(A_{k'}) < \frac{1}{2}$ (because if $v_{k'}(A_{k'}) \geq \frac{1}{2}$, then agent k' would have been added to the set of inactive agents *before* the last good was added to their bundle). This gives us

$$v_1(A_1) + v_2(A_2) < 1. \tag{3.2}$$

Note that for every good $g \in A_{k'}$, $v_{k'}(g) \geq v_k(g)$ (otherwise, since agent k remained in N' throughout, g should have been given to agent k instead of agent k', by Algorithm 12 of the algorithm). Taking the sum over all $g \in A_{k'}$ on both sides of the inequality, we get that

$$v_{k'}(A_{k'}) = \sum_{g \in A_{k'}} v_{k'}(g) \geq \sum_{g \in A_{k'}} v_k(g) = v_k(A_{k'})$$

Then, since $v_k(A_{k'}) = 1 - v_k(A_k)$ by the completeness and normalization information assumptions, we get that

$$v_{k'}(A_{k'}) \geq 1 - v_k(A_k) \implies v_{k'}(A_{k'}) + v_k(A_k) \geq 1,$$

contradicting (3.2).

In both cases, we arrived at a contradiction, and our result follows. □

Since EF1 implies 0.5-MMS when there are two agents [Amanatidis et al., 2018, Segal-Halevi and Suksompong, 2019], we also obtain the same 0.5-MMS guarantee as Zhou et al. [2023, Algorithm 1]. Next, we show that the algorithm returns a PROP1 allocation for any number of agents, thereby also providing the first satisfiable fairness property in this setting for three (or more) agents. The proof works using a similar high-level argument structure to that for EF1 when $n = 2$ (see the proof idea above).

Theorem 3.4.2. *For $n \geq 2$, given normalization information, Algorithm 1 returns a* PROP1 *allocation.*

Proof. We first prove that at the end of the algorithm, every inactive agent satisfies the PROP1 condition. Note that after the algorithm has terminated (i.e., all goods have appeared and been allocated), for any inactive agent $i \in N \setminus N'$, if $G = A_i$, then $v_i(A_i) = 1 \geq \frac{1}{n}$ and we trivially get our claim. Thus, assume $A_i \subset G$. Then, Lines 7 and 8 of the algorithm prescribes that there exists some good $g \in G \setminus A_i$ such that $v_i(A_i \cup \{g\}) \geq v_i(A_i) + \frac{n-1}{n} \cdot v_i(g) \geq \frac{1}{n}$, and thus every inactive agent satisfies PROP1.

It remains to prove that after the algorithm has terminated, there are no more active agents, i.e., $N' = \varnothing$. Suppose for a contradiction that when the algorithm has terminated and all goods have been allocated, there are still active agents, i.e., $|N'| \geq 1$. Consider such an agent $k \in N'$.

Suppose there are m goods in total, and label the goods $g_1, \ldots, g_m$ in the order they arrived. For each $j \in [m]$, let $a_j \in N$ denote the agent that received good g_j.

Note that for every good $g_j \in G \setminus A_k$, $v_k(g_j) \leq v_{a_j}(g_j)$ (otherwise, since agent k remained in N' throughout, g_j should have been given to agent k instead of agent a_j, by Line 12 of the algorithm). Taking the sum over all goods in $G \setminus A_k$, we get

$$\sum_{g_j \in G \setminus A_k} v_k(g_j) \leq \sum_{g_j \in G \setminus A_k} v_{a_j}(g_j). \tag{3.3}$$

Consequently, we get that

$$\begin{aligned} v_k(G) &= v_k(A_k) + \sum_{g_j \in G \setminus A_k} v_k(g_j) \\ &\leq v_k(A_k) + \sum_{g_j \in G \setminus A_k} v_{a_j}(g_j) \quad \text{(by (3.3))} \\ &= \sum_{i \in N} v_i(A_i). \end{aligned}$$

For every inactive agent $i \in N \setminus N'$, we have that $v_i(A_i) < \frac{1}{n}$ (because if $v_i(A_i) \geq \frac{1}{n}$, then the agent i would have been added to the set of inactive agents *before* the last good was added to their bundle). For every active agent $i \in N'$, we also have that $v_i(A_i) < \frac{1}{n}$ (because otherwise they would have been set as inactive in the final iteration). Taking these two facts together, we get that

$$\sum_{i \in N} v_i(A_i) < \frac{1}{n} \cdot n = 1.$$

Since we showed earlier that $v_k(G) \leq \sum_{i \in N} v_i(A_i)$, this implies that $v_k(G) \leq \sum_{i \in N} v_i(A_i)$ < 1, a contradiction to the fact that each agent's value over all goods sum to 1 (by the normalization information assumption). $\square$

We complement these positive results with two impossibility results, demonstrating that no algorithm can achieve better guarantees for the considered fairness notions in this setting with normalized information. We first consider the stronger fairness notion of EFX and show that no online algorithm can guarantee a positive competitive ratio even for two agents.

Theorem 3.4.3. *For $n \geq 2$, given normalization information, there does not exist any online algorithm with a competitive ratio strictly larger than 0 with respect to* EFX.

Proof. Suppose for a contradiction that there exists a γ-competitive algorithm for approximating EFX given normalization information, for some $\gamma \in (0, 1]$. Let $k \in \mathbb{N}$ be large. Consider the case of $n = 2$. Let the first two goods be g_1 and g_2, with $v_1(g_1) = v_2(g_1) = 2^{-k^3}, v_1(g_2) = 2^{-k^2}$, and $v_2(g_2) = 2^{-k^3}$. Assume w.l.o.g. that g_1 is allocated to agent 2. If g_2 is allocated to agent 1, let the next (and final) good be g_3 with valuations as follows:

v	g_1	g_2	g_3
1	2^{-k^3}	2^{-k^2}	$1 - 2^{-k^3} - 2^{-k^2}$
2	2^{-k^3}	2^{-k^3}	$1 - 2^{-k^3+1}$

Then, if g_3 is allocated to agent 1, we have that $v_2(A_2) = 2^{-k^3}$ and $v_2(A_1 \setminus \{g_2\}) = 1 - 2^{-k^3+1}$ (observe that $v_2(g_2) < v_2(g_3)$ for sufficiently large k). This gives us

$$\gamma \leq \frac{2^{-k^3}}{1 - 2^{-k^3+1}} = \frac{1}{2^{k^3} - 2} \to 0$$

as $k \to \infty$, giving us a contradiction. On the other hand, if g_3 is allocated to agent 2, then $v_1(A_1) = 2^{-k^2}$ and $v_1(A_2 \setminus \{g_1\}) = v_1(g_3) = 1 - 2^{-k^3} - 2^{-k^2}$ (observe that $v_1(g_1) < v_1(g_3)$ for some sufficiently large k). This gives us

$$\gamma \leq \frac{2^{-k^2}}{1 - 2^{-k^3} - 2^{-k^2}} = \frac{1}{2^{k^2} - 2^{-k^3+k^2} - 1} \to 0$$

as $k \to \infty$, giving us a contradiction. Thus, g_2 cannot be allocated to agent 1. Let it be allocated to agent 2. Then, let each successive good g that arrives have value $v_1(g) = 2^{-k^2}$ and $v_2(g) = 2^{-k^3}$ (up to a maximum of $\ell - 2 = 2^{k^2} - 2$ such goods).

Consider the earliest point in time where the algorithm allocates a good that arrives to agent 1. If it exists, let such a good be g_j (i.e., the j-th good that arrives overall). We split our analysis into two cases.

Case 1: $\{g_3, \ldots, g_\ell\} \subset A_2$. This means that none of the goods that arrived as described above was allocated to agent 1. Let the $(\ell+1)$-th and final good be $g_{\ell+1}$, with valuations as follows:

v	g_1	g_2	$\cdots$	g_ℓ	$g_{\ell+1}$
1	2^{-k^3}	2^{-k^2}	$\cdots$	2^{-k^2}	$2^{-k^2} - 2^{-k^3}$
2	2^{-k^3}	2^{-k^3}	$\cdots$	2^{-k^3}	$1 - 2^{-k^3+k^2}$

If $g_{\ell+1}$ is also allocated to agent 2, then $v_1(A_1) = 0$ and we get a 0-EFX allocation, a contradiction. Thus, $g_{\ell+1}$ must be allocated to agent 1. We have that $v_1(A_1) = 2^{-k^2} - 2^{-k^3}$ and $v_1(A_2) \setminus \{g_1\}) = (2^{k^2} - 1) \cdot 2^{-k^2} = 1 - 2^{-k^2}$ (observe that $v_1(g_1) = \min_{g \in A_2} v_1(g)$ for some sufficiently large k). This gives us

$$\gamma \leq \frac{2^{-k^2} - 2^{-k^3}}{1 - 2^{-k^2}} = \frac{1 - 2^{-k^3+k^2}}{2^{k^2} - 1} \to 0$$

as $k \to \infty$, giving us a contradiction.

Case 2: $\exists j : j \leq \ell$ such that $g_j \in A_1$. This means that $\{g_3, \ldots, g_{j-1}\} \subset A_2$. Let the $(j+1)$-th (and final) good be g_{j+1}, with valuations as follows:

v	g_1	g_2	$\cdots$	g_{j-1}	g_j	g_{j+1}
1	2^{-k^3}	2^{-k^2}	$\cdots$	2^{-k^2}	2^{-k^2}	$1 - 2^{-k^3} - (j-1) \cdot 2^{-k^2}$
2	2^{-k^3}	2^{-k^3}	$\cdots$	2^{-k^3}	2^{-k^3}	$1 - j2^{-k^3}$

Note that $k^3 - 1 > k^2$ for some sufficiently large $k > 1$. Equivalently, we get (via algebraic manipulation) that

$$2^{-k^3+1} < 2^{-k^2}. \tag{3.4}$$

Then, since $j \leq \ell = 2^{k^2}$, algebraic manipulation gives us $1-(j-1)\cdot 2^{-k^2} \geq 2^{-k^2}$. Combining this with (3.4) gives us

$$1 - 2^{-k^3} - (j-1) \cdot 2^{-k^2} > 2^{-k^3}. \tag{3.5}$$

Now, if g_{j+1} is allocated to agent 2, then $v_1(A_1) = 2^{-k^2}$ and $v_1(A_2 \setminus \{g_1\}) = 1 - 2^{-k^3} - 2^{-k^2}$ (note that by (3.5), we have that $v_1(g_1) < v_1(g_{j+1})$). This gives us

$$\gamma \le \frac{2^{-k^2}}{1 - 2^{-k^3} - 2^{-k^2}} = \frac{1}{2^{k^2} - 2^{-k^3+k^2} - 1} \to 0$$

as $k \to \infty$, giving us a contradiction.

In all cases, we arrive at a contradiction, thereby proving our result.

For the case of $n \ge 3$, we refer the reader to the proof of Theorem 3.6.4. □

Next, we turn to EF1 and show that for three or more agents, no online algorithm can guarantee a positive competitive ratio, thereby complementing our positive result for $n = 2$.

Theorem 3.4.4. *For $n \ge 3$, given normalization information, there does not exist any online algorithm with a competitive ratio strictly larger than 0 with respect to EF1.*

Proof. Suppose for a contradiction that there exists a γ-competitive algorithm for approximating EF1 given normalization information, for some $\gamma \in (0, 1]$. Let $\varepsilon > 0$ be a sufficiently small constant and $\delta > 1$ be a large constant. Let the first two goods be g_1, g_2 with valuations as follows:

v	g_1	g_2
1	ε	$\frac{1}{n}$
2	ε	$\frac{\varepsilon}{\delta}$
3	ε	$\frac{\varepsilon}{\delta}$
$\vdots$	$\vdots$	$\vdots$
n	ε	$\frac{\varepsilon}{\delta}$

Without loss of generality, we assume that g_1 is allocated to agent 1. If g_2 is allocated to agent 1, let the next $n-1$ (and final) goods be $g_3, \ldots, g_{n+1}$ with the following valuations:

v	g_1	g_2	g_3	$\cdots$	g_n	g_{n+1}
1	ε	$\frac{1}{n}$	$\frac{1}{n} + \varepsilon$	$\cdots$	$\frac{1}{n} + \varepsilon$	$\frac{1}{n} - (n-1)\varepsilon$
2	ε	$\frac{\varepsilon}{\delta}$	$\frac{\varepsilon}{\delta^2}$	$\cdots$	$\frac{\varepsilon}{\delta^2}$	$1 - \varepsilon - \frac{\varepsilon}{\delta} - (n-2)\frac{\varepsilon}{\delta^2}$
3	ε	$\frac{\varepsilon}{\delta}$	$\frac{\varepsilon}{\delta^2}$	$\cdots$	$\frac{\varepsilon}{\delta^2}$	$1 - \varepsilon - \frac{\varepsilon}{\delta} - (n-2)\frac{\varepsilon}{\delta^2}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
n	ε	$\frac{\varepsilon}{\delta}$	$\frac{\varepsilon}{\delta^2}$	$\cdots$	$\frac{\varepsilon}{\delta^2}$	$1 - \varepsilon - \frac{\varepsilon}{\delta} - (n-2)\frac{\varepsilon}{\delta^2}$

Suppose each good in $\{g_3, \ldots, g_{n+1}\}$ is allocated to a unique agent in $N \setminus \{1\}$ (i.e., each agent in $N \setminus \{1\}$ receives exactly one good). Then, consider the agent that received g_3, let it be agent i. However, $v_i(A_i) = v_i(g_3) = \frac{\varepsilon}{\delta^2}$ and $v_i(A_1 \setminus \{g_1\}) = \frac{\varepsilon}{\delta}$, giving us

$$\gamma \leq \frac{\frac{\varepsilon}{\delta^2}}{\frac{\varepsilon}{\delta}} = \frac{1}{\delta} \to 0$$

as $\delta \to \infty$, a contradiction. Thus, we must have that either some good in $\{g_3, \ldots, g_{n+1}\}$ is allocated to agent 1 and/or some agent in $N \setminus \{1\}$ receives strictly more than one good in the set. However, in either of these cases, by the pigeonhole principle, there must exist an agent in $N \setminus \{1\}$ that receives no good, leading to a 0-EF1 allocation. Hence, g_2 cannot be allocated to agent 1.

W.l.o.g., let g_2 be allocated to agent 2, and the next $n-1$ (and final) goods be $g_3, \ldots, g_{n+1}$ with the following valuations:

$\mathbf{v}$	g_1	g_2	g_3	$\cdots$	g_n	g_{n+1}
1	ε	$\frac{1}{n}$	$\frac{1}{n}+\varepsilon$	$\ldots$	$\frac{1}{n}+\varepsilon$	$\frac{1}{n}-(n-1)\varepsilon$
2	ε	$\frac{\varepsilon}{\delta}$	ε	$\ldots$	ε	$1-(n-1)\varepsilon-\frac{\varepsilon}{\delta}$
3	ε	$\frac{\varepsilon}{\delta}$	ε	$\ldots$	ε	$1-(n-1)\varepsilon-\frac{\varepsilon}{\delta}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
n	ε	$\frac{\varepsilon}{\delta}$	ε	$\ldots$	ε	$1-(n-1)\varepsilon-\frac{\varepsilon}{\delta}$

If agent 1 and 2 collectively is allocated at least two goods from $\{g_3, \ldots, g_{n+1}\}$, then by the pigeonhole principle, there exists some agent in $N \setminus \{1, 2\}$ that will receive no good, and we get a 0-EF1 allocation, a contradiction. Thus, we assume that agent 1 and 2 collectively is allocated at most one good from $\{g_3, \ldots, g_{n+1}\}$. We split our analysis into two cases.

Case 1: $|(A_1 \cup A_2) \cap \{g_3, \ldots, g_{n+1}\}| = 1$. Let such a good in the intersection be g^*. If g^* was allocated to agent 1, then $v_2(A_2) = \frac{\varepsilon}{\delta}$ and $v_2(A_1 \setminus \{g\}) = v_2(g_1) = \varepsilon$ for $g \in A_1 \setminus \{g_1\}$ and sufficiently small ε. Consequently, we get that

$$\gamma \leq \frac{\frac{\varepsilon}{\delta}}{\varepsilon} = \frac{1}{\delta} \to 0$$

as $\delta \to \infty$, a contradiction. Thus, we must have that g^* is allocated to agent 2. Then, $v_1(A_1) = \varepsilon$ and $v_1(A_2 \setminus \{g\}) \geq \frac{1}{n} - (n-1)\varepsilon$ for $g \in \arg\min_{g' \in A_2} v_1(g')$, giving us

$$\gamma \leq \frac{\varepsilon}{\frac{1}{n} - (n-1)\varepsilon} \to 0$$

as $\varepsilon \to 0$, a contradiction.

Case 2: $|(A_1 \cup A_2) \cap \{g_3, \ldots, g_{n+1}\}| = 0$**.** Then, by the pigeonhole principle, there exists some agent $i \in N \backslash \{1, 2\}$ that receives at least two goods from $\{g_3, \ldots, g_{n+1}\}$. Then, $v_2(A_2) = v_2(g_2) = \frac{\varepsilon}{\delta}$ and $v_2(A_i \setminus \{g\}) \geq \varepsilon$ for $g \in \arg\min_{g' \in A_i} v_2(g')$ and some sufficiently small ε. Consequently, we get that

$$\gamma \leq \frac{\frac{\varepsilon}{\delta}}{\varepsilon} = \frac{1}{\delta} \to 0$$

as $\delta \to 0$, a contradiction.

In both cases, we arrive at a contradiction, and our result follows. □

The algorithm presented in this section offers some hope for achieving fairness, assuming access to some information about the future—specifically, normalization information, a well-studied and commonly used assumption in online fair division. Despite this, fundamental impossibilities still remain. This motivates the question: can stronger fairness guarantees be achieved by providing the algorithm with additional information? We explore this possibility in the next section.

3.5 Frequency Predictions

We consider a stronger form of information—*frequency predictions*—inspired by similar notions studied in the online knapsack [Im et al., 2021], online bin packing [Angelopoulos et al., 2023], and online matching [Mehta et al., 2007] literature. These types of predictive information have also been explored in the context of revenue management [Balseiro et al., 2023]. Specifically, we assume access to a predictor that, for each agent and value, tells us the frequency with which this value will appear among the agent's valuations for goods that will arrive. Formally, for each agent $i \in N$, V_i denotes the multiset containing values that agent i will have for goods that will arrive (we call this the *frequency multiset*). Note that this also gives us knowledge of the precise number of items that will arrive, but not their arrival order. We are interested in understanding whether access to frequency predictions can improve the fairness guarantees in the online setting.

In this setting, we establish a considerably general result: any *share-based* fairness notion [Babaioff and Feige, 2024] that is achievable in the single-shot setting can also be obtained via an online algorithm with frequency predictions. Formally, a share s is a function mapping each frequency multiset V_i and number of agents n to a share $s(V_i, n)$. A share is *feasible*, if for every instance, there exists an allocation giving each agent a utility of at least $s(V_i, n)$. Examples of feasible shares (i.e., shown to be achievable in the single-shot setting) include *round robin share* (RRS) [Conitzer et al., 2017, Gourvès et al., 2021], *minimum EFX share* (MXS) [Caragiannis et al., 2025], or $(\frac{3}{4} + \frac{3}{3836})$-MMS [Akrami and Garg, 2024]. Then, our result is as follows.

Theorem 3.5.1. *Let s be a feasible share in the single-shot fair division setting. Then, in the online fair division setting, there exists an online algorithm using frequency predictions that guarantees each agent a bundle value of at least $s(V_i, n)$.*

Proof. Let $g_1, \ldots, g_m$ be the order of goods that arrive in the online setting. Fix some $t \in [m]$. Given an agent $i \in N$, let V_i^t denote the frequency multiset for goods in $\{g_t, \ldots, g_m\}$. We note that $V_i^1 = V_i$ is agents i's frequency multiset for all goods that will arrive. We say that a valuation function v_i of agent i is *consistent* with V_i^t if $V_i^t = \{v_i(g_t), \ldots, v_i(g_m)\}$. Let $v_i^{t,\mathsf{IDO}}$ be a valuation function of agent i that is consistent with V_i^t and where for each $t \leq j \leq j'$, $v_i^{t,\mathsf{IDO}}(g_j) \leq v_i^{t,\mathsf{IDO}}(g_{j'})$, with the valuation profile $\mathbf{v}^{t,\mathsf{IDO}} = (v_1^{t,\mathsf{IDO}}, \ldots, v_n^{t,\mathsf{IDO}})$.[7]

Now, consider a single-shot fair division instance with goods $\{g_t, \ldots, g_m\}$. Let $\pi^t = (a_t, \ldots, a_m)$ be a *picking sequence* over goods $\{g_t, \ldots, g_m\}$, whereby $a_i \in N$ for each $i \in \{t, \ldots, m\}$, and agents take turns (according to π^t) picking their highest-valued good (according to their valuation function) from among the remaining goods available, breaking ties in a consistent manner (say, in favor of lower-indexed goods). Then, given a valuation profile $\mathbf{v} = (v_1, \ldots, v_n)$ and picking sequence π^t, we denote $\mathcal{A}^{\pi^t,\mathbf{v}} = \left(A_1^{\pi^t,\mathbf{v}}, \ldots, A_n^{\pi^t,\mathbf{v}}\right)$ as the allocation returned in the single-shot instance over the set of goods $\{g_t, \ldots, g_m\}$ induced by the picking sequence π^t and valuation profile $\mathbf{v}$.

We then obtain the following result, which will be useful later on.

Lemma 3.5.2. *Fix $t \in [m]$. For any picking sequence π^t and any valuation profiles $\mathbf{v} = (v_1, \ldots, v_n)$ and $\mathbf{v}^{t,\mathsf{IDO}} = (v_1^{t,\mathsf{IDO}}, \ldots, v_n^{t,\mathsf{IDO}})$ such that for each $i \in N$, v_i and $v_i^{t,\mathsf{IDO}}$ are both consistent with V_i^t, we have that*

$$v_i\left(A_i^{\pi^t,\mathbf{v}}\right) \geq v_i^{t,\mathsf{IDO}}\left(A_i^{\pi^t,\mathbf{v}^{t,\mathsf{IDO}}}\right).$$

Proof. Consider any $t \in [m]$. Suppose we have a picking sequence π^t and valuation profiles $\mathbf{v} = (v_1, \ldots, v_n)$ and $\mathbf{v}^{t,\mathsf{IDO}} = (v_1^{t,\mathsf{IDO}}, \ldots, v_n^{t,\mathsf{IDO}})$ such that for each $i \in N$, v_i and $v_i^{t,\mathsf{IDO}}$ are both consistent with V_i^t.

An equivalent way to express a picking sequence π^t is by looking at the *turns* each agent gets to pick a good. Note that for goods $\{g_t, \ldots, g_m\}$, there will be a total

[7] IDO is shorthand for *identical ordering*, as is standard in the literature.

of exactly $m - t + 1$ turns (where at each turn, an agent gets to pick his favorite remaining good).

We introduce several notation:

- Let $\mathsf{turns}_i \subseteq [m - t + 1]$ be the set of turns whereby agents i gets to pick the good under π^t. Note that $\bigcup_{i \in N} \mathsf{turns}_i = [m - t + 1]$ and for any $i \neq j$, $\mathsf{turns}_i \cap \mathsf{turns}_j = \varnothing$.
- Let $\mathsf{top}_i(k)$ be the k-th most valuable good of agent $i \in N$ in according to the frequency multiset V_i^t, for $k \in [m - t + 1]$.
- Let $h_{k,\mathbf{v}}$ be the k-th good that was picked according to π^t under $\mathbf{v}$.

Consider any agent $i \in N$. Note that for each $k \in \mathsf{turns}_i$ (i.e., in the k-th entry of π^t, it is agent i's turn to pick), under $\mathbf{v}^{t,\mathsf{IDO}}$, since goods in $\{g_t, \dots, g_m\}$ are identically ordered according to all agents' valuations, agent i will get to (and must) pick her k-th favorite good, i.e., $v_i^{t,\mathsf{IDO}}(h_{k,\mathbf{v}^{t,\mathsf{IDO}}}) = \mathsf{top}_i(k)$. Summing over all such $k \in \mathsf{turns}_i$, we get

$$v_i^{t,\mathsf{IDO}}\left(A_i^{\pi^t,\mathbf{v}^{t,\mathsf{IDO}}}\right) = \sum_{k \in \mathsf{turns}_i} v_i^{t,\mathsf{IDO}}(h_{k,\mathbf{v}^{t,\mathsf{IDO}}}) = \sum_{k \in \mathsf{turns}_i} \mathsf{top}_i(k). \tag{3.6}$$

Now, for each $k \in \mathsf{turns}_i$, under $\mathbf{v}$, we note that agent i's value for the good picked will be at least her value for her k-th favorite good, i.e., $v_i(h_{k,\mathbf{v}}) \geq \mathsf{top}_i(k)$ as only $k - 1$ goods have been selected before this turn. Summing over all $k \in \mathsf{turns}_i$, we get

$$v_i\left(A_i^{\pi^t,\mathbf{v}}\right) \geq \sum_{k \in \mathsf{turns}_i} v_i(h_{k,\mathbf{v}}) = \sum_{k \in \mathsf{turns}_i} \mathsf{top}_i(k). \tag{3.7}$$

Combining (3.6) and (3.7), we get $v_i\left(A_i^{\pi^t,\mathbf{v}}\right) \geq v_i^{t,\mathsf{IDO}}\left(A_i^{\pi^t,\mathbf{v}^{t,\mathsf{IDO}}}\right)$, as desired. $\square$

Next, we introduce the following algorithm (Algorithm 2). Note that given frequency predictions, we know the exact *number* of goods that will arrive (let it be m). This allows us to use a **for** loop rather than a **while** loop.

Next, we prove the following property about Algorithm 2.

Lemma 3.5.3. *Given the picking sequence π^1, Algorithm 2 returns an allocation $\mathcal{A} = (A_1, \dots, A_n)$ such that for all agents $i \in N$ and any $v_i^{1,\mathsf{IDO}}$ consistent with V_i,*

$$v_i\left(A_i\right) \geq v_i^{1,\mathsf{IDO}}\left(A_i^{\pi^1,\mathbf{v}^{1,\mathsf{IDO}}}\right) \quad \text{and} \quad |A_i| = \left|A_i^{\pi^1,\mathbf{v}^{1,\mathsf{IDO}}}\right|$$

Algorithm 2: Algorithm given frequency predictions and a picking sequence π^1

Initialize the empty allocation $\mathcal{A}$ where $A_i = \varnothing$ for all $i \in N$
for $t \in [m]$ **do**
  for $i \in N$ **do**
    $V_i^{t+1} \leftarrow V_i^t \setminus \{v_i(g_t)\}$
    Define v_i^t as follows

$$v_i^t(g_j) := \begin{cases} v_i(g_t) & \text{if } j = t \\ v_i^{t+1,\mathsf{IDO}} & \text{otherwise} \end{cases}$$

  end for
  Let $\mathbf{v}^t := (v_1^t, \dots, v_n^t)$
  Let $i^* \in N$ be the agent such that $g_t \in A_{i^*}^{\pi^t, \mathbf{v}^t}$
  Let π^{t+1} be equivalent to π^t, except the entry of agent i^* that picked g_t is removed
  $A_{i^*} \leftarrow A_{i*} \cup \{g_t\}$
end for
return $\mathcal{A} = (A_1, \dots, A_n)$

Proof. For each $i \in N$ and $t \in [m]$, let A_i^t denote the partial allocation of agent i *after* g_t is allocated to an agent and let $A_i^0 = \varnothing$ be the initial empty allocation. We will prove that for every $i \in N$ and $k \in \{0, \dots, m\}$,

$$v_i(A_i^k) + v_i^{k+1,\mathsf{IDO}}\left(A_i^{\pi^{k+1}, \mathbf{v}^{k+1,\mathsf{IDO}}}\right) \geq v_i^{1,\mathsf{IDO}}\left(A^{\pi^1, \mathbf{v}_i^{1,\mathsf{IDO}}}\right), \text{ and} \tag{3.8}$$

$$|A_i^k| + \left|A_i^{\pi^{k+1}, \mathbf{v}^{k+1,\mathsf{IDO}}}\right| = \left|A_i^{\pi^1, \mathbf{v}^{1,\mathsf{IDO}}}\right| \tag{3.9}$$

We proceed by induction. First, consider the base case: we are given that $A_i^0 = \varnothing$ for all $i \in N$. Thus, for all $i \in N$, we have that

(i) $v_i(A_i^0) + v_i^{1,\mathsf{IDO}}\left(A_i^{\pi^1, \mathbf{v}^{1,\mathsf{IDO}}}\right) = v_i(\varnothing) + v_i^{1,\mathsf{IDO}}\left(A_i^{\pi^1, \mathbf{v}^{1,\mathsf{IDO}}}\right) \geq v_i^{1,\mathsf{IDO}}\left(A_i^{\pi^1, \mathbf{v}^{1,\mathsf{IDO}}}\right)$, and

(ii) $|A_i^0| + \left|A_i^{\pi^{t+1}, \mathbf{v}^{1,\mathsf{IDO}}}\right| = 0 + \left|A_i^{\pi^{t+1}, \mathbf{v}^{1,\mathsf{IDO}}}\right| = \left|A_i^{\pi^1, \mathbf{v}^{1,\mathsf{IDO}}}\right|$,

thus proving the base case where $k = 0$.

Next, suppose that for each $i \in N$ and $\ell \in \{0, \dots, m-1\}$, we have

$$v_i(A_i^\ell) + v_i^{\ell+1,\mathsf{IDO}}\left(A_i^{\pi^{\ell+1}, \mathbf{v}^{\ell+1,\mathsf{IDO}}}\right) \geq v_i^{1,\mathsf{IDO}}\left(A^{\pi^1, \mathbf{v}_i^{1,\mathsf{IDO}}}\right), \text{ and} \tag{3.10}$$

$$|A_i^\ell| + \left|A_i^{\pi^{\ell+1}, \mathbf{v}^{\ell+1,\mathsf{IDO}}}\right| = \left|A_i^{\pi^1, \mathbf{v}^{1,\mathsf{IDO}}}\right|. \tag{3.11}$$

We will show that

$$v_i(A_i^{\ell+1}) + v_i^{\ell+2,\mathsf{IDO}}\left(A_i^{\pi^{\ell+2},\mathbf{v}^{\ell+2,\mathsf{IDO}}}\right) \geq v_i^{1,\mathsf{IDO}}\left(A^{\pi^1,\mathbf{v}_i^{1,\mathsf{IDO}}}\right), \text{ and} \tag{3.12}$$

$$|A_i^{\ell+1}| + \left|A_i^{\pi^{\ell+2},\mathbf{v}^{\ell+2,\mathsf{IDO}}}\right| = \left|A_i^{\pi^1,\mathbf{v}^{1,\mathsf{IDO}}}\right|. \tag{3.13}$$

Now, we have that for all $i \in N$ and any $v_i^{\ell+1,\mathsf{IDO}}$, it is easy to observe that

$$\left|A_i^{\pi^{\ell+1},\mathbf{v}^{\ell+1}}\right| = \left|A_i^{\pi^{\ell+1},\mathbf{v}^{\ell+1,\mathsf{IDO}}}\right|, \tag{3.14}$$

since under the same picking sequence $\pi^{\ell+1}$, any agent i should have the same number of goods, regardless of her valuation function.

Let agent $i^* \in N$ be the agent that is allocated $g_{\ell+1}$ by the algorithm. This gives us

$$A_i^{\ell+1} = A_i^{\ell} \text{ for all } i \in N \setminus \{i^*\}, \text{ and } A_{i^*}^{\ell+1} = A_{i^*}^{\ell} \cup \{g_{\ell+1}\}. \tag{3.15}$$

Now, since $\pi^{\ell+2}$ is equivalent to $\pi^{\ell+1}$ but with the entry of agent i^* that picked $g_{\ell+1}$ removed (see Line 9 of the algorithm), we have that

$$A_i^{\pi^{\ell+2},\mathbf{v}^{\ell+2,\mathsf{IDO}}} = A_i^{\pi^{\ell+1},\mathbf{v}^{\ell+1}} \text{ for all } i \in N\setminus\{i^*\}, \text{ and } A_{i^*}^{\pi^{\ell+2},\mathbf{v}^{\ell+2,\mathsf{IDO}}} = A_{i^*}^{\pi^{\ell+1},\mathbf{v}^{\ell+1,\mathsf{IDO}}}\setminus\{g_{\ell+1}\}, \tag{3.16}$$

where the left equality holds by Line 5 of the algorithm.

Lastly, we note that by Line 5 of the algorithm, for each $i \in N$ and $j \in \{\ell + 1, \ldots, m\}$,

$$v_i^{\ell+2,\mathsf{IDO}}(g_j) = v_i^{\ell+1}(g_j). \tag{3.17}$$

Then, we have that for all agents $i \in N \setminus \{i^*\}$,

$$\begin{aligned}
v_i(A_i^{\ell+1}) + v_i^{\ell+2,\mathsf{IDO}}\left(A_i^{\pi^{\ell+2},\mathbf{v}^{\ell+2,\mathsf{IDO}}}\right) &= v_i(A_i^{\ell}) + v_i^{\ell+2,\mathsf{IDO}}\left(A_i^{\pi^{\ell+2},\mathbf{v}^{\ell+2,\mathsf{IDO}}}\right) \quad \text{(by (3.15))} \\
&= v_i(A_i^{\ell}) + v_i^{\ell+1}\left(A_i^{\pi^{\ell+1},\mathbf{v}^{\ell+1}}\right) \quad \text{(by (3.16) and (3.17))} \\
&\geq v_i(A_i^{\ell}) + v_i^{\ell+1,\mathsf{IDO}}\left(A_i^{\pi^{\ell+1},\mathbf{v}^{\ell+1,\mathsf{IDO}}}\right) \quad \text{(by Lemma 3.5.2)} \\
&\geq v_i^{1,\mathsf{IDO}}\left(A_i^{\pi^1,\mathbf{v}^{1,\mathsf{IDO}}}\right) \quad \text{(by (3.10))}
\end{aligned}$$

and

$$\begin{aligned}
|A_i^{\ell+1}| + \left|A_i^{\pi^{\ell+2},\mathbf{v}^{\ell+2,\mathsf{IDO}}}\right| &= |A_i^{\ell}| + \left|A_i^{\pi^{\ell+1},\mathbf{v}^{\ell+1}}\right| \quad \text{(by (3.15) and (3.16))} \\
&= |A_i^{\ell}| + \left|A_i^{\pi^{\ell+1},\mathbf{v}^{\ell+1,\mathsf{IDO}}}\right| \quad \text{(by (3.14))} \\
&= \left|A_i^{\pi^1,\mathbf{v}^{1,\mathsf{IDO}}}\right| \quad \text{(by (3.11))}.
\end{aligned}$$

Moreover, by Line 5 of the algorithm, we have that $v_{i^*}(g_{\ell+1}) = v_{i^*}^{\ell+1}(g_{\ell+1})$. Then, we have that

$$\begin{aligned}
& v_{i^*}(A_{i^*}^{\ell+1}) + v_{i^*}^{\ell+2,\mathsf{IDO}}\left(A_{i^*}^{\pi^{\ell+2},\mathbf{v}^{\ell+2,\mathsf{IDO}}}\right) \\
&= v_{i^*}\left(A_{i^*}^{\ell} \cup \{g_{\ell+1}\}\right) + v_{i^*}^{\ell+2,\mathsf{IDO}}\left(A_{i^*}^{\pi^{\ell+1},\mathbf{v}^{\ell+1,\mathsf{IDO}}} \setminus \{g_{\ell+1}\}\right) \quad \text{(by (3.15) and (3.16))} \\
&= v_{i^*}\left(A_{i^*}^{\ell} \cup \{g_{\ell+1}\}\right) + v_{i^*}^{\ell+1}\left(A_{i^*}^{\pi^{\ell+1},\mathbf{v}^{\ell+1,\mathsf{IDO}}} \setminus \{g_{\ell+1}\}\right) \quad \text{(by (3.17))} \\
&= v_{i^*}(A_{i^*}^{\ell}) + v_{i^*}^{\ell+1}\left(A_{i^*}^{\pi^{\ell+1},\mathbf{v}^{\ell+1,\mathsf{IDO}}}\right) \quad (\text{since } v_{i^*}(g_{\ell+1}) = v_{i^*}^{\ell+1}(g_{\ell+1})) \\
&\geq v_{i^*}(A_{i^*}^{\ell}) + v_{i^*}^{\ell+1,\mathsf{IDO}}\left(A_{i^*}^{\pi^{\ell+1},\mathbf{v}^{\ell+1,\mathsf{IDO}}}\right) \quad \text{(by Lemma 3.5.2)} \\
&\geq v_{i^*}^{1,\mathsf{IDO}}\left(A_{i^*}^{\pi^{1},\mathbf{v}^{1,\mathsf{IDO}}}\right) \quad \text{(by (3.10))}
\end{aligned}$$

and

$$\begin{aligned}
|A_{i^*}^{\ell+1}| + \left|A_{i^*}^{\pi^{\ell+2},\mathbf{v}^{\ell+2,\mathsf{IDO}}}\right| &= \left|A_{i^*}^{\ell} \cup \{g_{\ell+1}\}\right| + \left|A_{i^*}^{\pi^{\ell+1},\mathbf{v}^{\ell+1,\mathsf{IDO}}} \setminus \{g_{\ell+1}\}\right| \quad \text{(by (3.15) and (3.16))} \\
&= \left|A_{i^*}^{\ell}| + |A_{i^*}^{\pi^{\ell+1},\mathbf{v}^{\ell+1,\mathsf{IDO}}}\right| \\
&= \left|A_{i^*}^{\pi^{1},\mathbf{v}^{1,\mathsf{IDO}}}\right| \quad \text{(by (3.11))}.
\end{aligned}$$

Thus, by induction, (3.8) and (3.9) holds.

Now, when Algorithm 2 ends (i.e., when $t = m$), we have that for all $i \in N$,

(i) $v_i\left(A_i^m\right) + v_i^{m+1,\mathsf{IDO}}\left(A_i^{\pi^{m+1},\mathbf{v}^{m+1,\mathsf{IDO}}}\right) \geq v_i^{1,\mathsf{IDO}}\left(A^{\pi^1,\mathbf{v}_i^{1,\mathsf{IDO}}}\right)$, and

(ii) $|A_i^m| + \left|A_i^{\pi^{m+1},\mathbf{v}^{m+1,\mathsf{IDO}}}\right| = \left|A_i^{\pi^1,\mathbf{v}^{1,\mathsf{IDO}}}\right|$,

where we let $V_i^{m+1} = \varnothing$. Consequently, we get that $A_i^{\pi^{m+1},\mathbf{v}^{m+1,\mathsf{IDO}}} = \varnothing$ for all $i \in N$. This gives us, for all $i \in N$,

$$v_i(A_i) = v_i\left(A_i^m\right) \geq v_i^{1,\mathsf{IDO}}\left(A^{\pi^1,\mathbf{v}_i^{1,\mathsf{IDO}}}\right) \quad \text{and} \quad |A_i| = |A_i^m| = |A_i^{\pi^1,\mathbf{v}^{1,\mathsf{IDO}}}|,$$

as desired, and concludes our proof for this lemma. □

Finally, let s be any share that is feasible in the single-shot fair division setting. This means that, given any $\mathbf{v}^{1,\mathsf{IDO}}$ that is consistent with $V_i = V_i^1$, there exists an allocation $\mathcal{X} = (X_1, \ldots, X_n)$ such that for all $i \in N$

$$v_i^{1,\mathsf{IDO}}(X_i) \geq s(V_i, n).$$

Let π be the picking sequence such that $\mathcal{X} = \mathcal{A}^{\pi,\mathbf{v}^{1,\mathsf{IDO}}}$. We note that there must exist such a picking sequence. To see this, observe that we can sort the goods in decreasing order of agents' preferences (since valuations are IDO, all agents will have the same such order). Then, consider the agent that received each of the goods (in this decreasing order) under $\mathcal{X}$, and construct the picking sequence in that order. Since at each round, the agent selects his favorite unallocated good, we will obtain exactly the allocation $\mathcal{X}$.

Then, using π and $(V_i)_{i\in N}$ as inputs to Algorithm 2, by Lemma 3.5.3, the algorithm will return an allocation $\mathcal{A} = (A_1, \ldots, A_n)$ such that for all $i \in N$,

$$v_i(A_i) \geq v_i^{1,\mathsf{IDO}}(A_i^{\pi,\mathbf{v}^{1,\mathsf{IDO}}}) = v_i^{1,\mathsf{IDO}}(X_i) \geq s(V_i, n),$$

giving us our result as desired. □

As discussed before, with normalization information, no online algorithm can achieve better than 0.5-MMS for $n = 2$ or even any positive approximation to MMS for $n > 2$. In contrast, frequency predictions allow us to leverage the best-known MMS approximation in the single-shot setting for any number of agents, giving us the following result.

Corollary 3.5.4. *For $n \geq 2$, given frequency predictions, there exists an online algorithm that returns a $(\frac{3}{4} + \frac{3}{3836})$-*MMS *allocation.*

Interestingly, we can further extend this framework to show that EFX is achievable for two agents with frequency predictions, by leveraging the leximin++ cut-and-choose procedure of Plaut and Roughgarden [2020, Algorithms 1 and 2]. Our result is as follows.

Theorem 3.5.5. *For $n = 2$, given frequency predictions, there exists an online algorithm that always returns an* EFX *allocation.*

Proof. Without loss of generality, we assume $v_i(G) = \sum_{k\in V_i} k = 1$ for each $i \in \{1, 2\}$. We run the leximin++ cut-and-choose algorithm by Plaut and Roughgarden [2020, Algorithms 1 and 2], where agent 1 divides the set of goods into two (according to valuation function $v_1^{1,\mathsf{IDO}}$) and agent 2 chooses her favorite bundle (according to valuation function $v_2^{1,\mathsf{IDO}}$). Let $\mathcal{A}' = (A_1', A_2')$ be the resulting allocation, whereby $(A_1', A_2') \in \arg\min_{(X_1,X_2)\in\Pi} |v_1^{1,\mathsf{IDO}}(X_1) - v_1^{1,\mathsf{IDO}}(X_2)|$, where Π is the space of all allocations in this case, and ties are broken as follows: first let $i \in \arg\min_{j\in\{1,2\}} v_1^{1,\mathsf{IDO}}(A_j')$, if there are further ties, let $i \in \arg\max_{j\in\{1,2\}} v_1^{1,\mathsf{IDO}}|A_j'|$.

Let π be the picking sequence such that $\mathcal{A}' = \mathcal{A}^{\pi,\mathbf{v}^{1,\mathsf{IDO}}}$. We note that there must exist such a picking sequence. To see this, observe that we can sort the goods in decreasing order of agents' preferences (since valuations are IDO, all agents will have the same such order). Then, consider the agent that received each of the goods (in this decreasing order) under $\mathcal{A}'$, and construct the picking sequence in that order. Since at each round, the agent selects his favorite unallocated good, we will obtain exactly the allocation $\mathcal{A}'$.

Without loss of generality, suppose $v_2^{1,\mathsf{IDO}}(A_1') \leq v_2^{1,\mathsf{IDO}}(A_2')$. This necessarily means that

$$v_2^{1,\mathsf{IDO}}(A_2') \geq \frac{v_2(G)}{2} = \frac{1}{2}. \tag{3.18}$$

Then, using π and (V_1, V_2) as inputs to Algorithm 2, by Lemma 3.5.3, the algorithm will return an allocation $\mathcal{A} = (A_1, A_2)$ such that for each $i \in N$ and any $v_i^{1,\mathsf{IDO}}$ consistent with V_i,

$$v_i(A_i) \geq v_i^{1,\mathsf{IDO}}(A_i'). \tag{3.19}$$

Combining this with (3.18), we get that $v_2(A_2) \geq v_2^{1,\mathsf{IDO}}(A_2') \geq \frac{1}{2}$, and agent 2 will not envy agent 1. It remains to prove that agent 1 does not envy agent 2. Note that if $v_1^{1,\mathsf{IDO}}(A_1') \geq v_1^{1,\mathsf{IDO}}(A_2')$, then $v_1^{1,\mathsf{IDO}}(A_1') \geq \frac{1}{2}$ and together with (3.19), we get that $v_1(A_1) \geq v_1^{\mathsf{IDO}}(A_1') \geq \frac{1}{2}$ and agent 1 will not envy agent 2. Thus, we assume that

$$v_1^{1,\mathsf{IDO}}(A_1') < v_1^{1,\mathsf{IDO}}(A_2') \tag{3.20}$$

Suppose for a contradiction that $\mathcal{A}$ is not EFX, i.e., there exists a $g^* \in A_2$ such that

$$v_1(A_1) < v_1(A_2 \setminus \{g^*\}). \tag{3.21}$$

Let $f : [m] \to [m]$ be a bijection such that $v_1(g) = v_1^{1,\mathsf{IDO}}(f(g))$ for all $g \in G$. Then, consider the allocation $\mathcal{B} = (B_1, B_2)$ where

$$B_1 = \{f(g) \mid g \in A_1\} \cup \{f(g^*)\} \quad \text{and} \quad B_2 = \{f(g) \mid g \in A_2 \setminus \{f(g^*)\}\}.$$

Note that this is possible because of (3.20). We split our analysis into two cases.

Case 1: $v_1^{1,\mathsf{IDO}}(B_1) > v_1^{1,\mathsf{IDO}}(B_2)$. Then we have that

$$v_1^{1,\mathsf{IDO}}(B_1) > v_1^{1,\mathsf{IDO}}(B_2) = v_1(A_2 \setminus \{g^*\}) > v_1(A_1) \geq v_1^{1,\mathsf{IDO}}(A_1'), \tag{3.22}$$

where the equality follows from the definition of B_2 and f, and the strict and weak inequalities on the right follow from (3.21) and (3.19), respectively. Then, since $v_1^{1,\mathsf{IDO}}(B_1) = 1 - v_1^{1,\mathsf{IDO}}(B_2)$ and $v_1^{1,\mathsf{IDO}}(A_1') = 1 - v_1^{1,\mathsf{IDO}}(A_2')$, substituting this into (3.22), we get

$$v_1^{1,\mathsf{IDO}}(A_2') > v_1^{1,\mathsf{IDO}}(B_2).$$

Combining this with (3.20) and (3.22), we get

$$v_1^{1,\mathsf{IDO}}(B_1) > v_1^{1,\mathsf{IDO}}(A_1') > v_1^{1,\mathsf{IDO}}(A_2') > v_1^{1,\mathsf{IDO}}(B_2).$$

Since all the above terms are at least 0, we get that Thus, $|v_1^{1,\mathsf{IDO}}(B_1) - v_1^{1,\mathsf{IDO}}(B_2)| < |v_1^{1,\mathsf{IDO}}(A_1') - v_1^{1,\mathsf{IDO}}(A_2')|$, which contradicts the way (A_1', A_2') is chosen.

Case 2: $v_1^{1,\mathsf{IDO}}(B_1) \leq v_1^{1,\mathsf{IDO}}(B_2)$. If $v_1^{1,\mathsf{IDO}}(B_1) > v_1^{1,\mathsf{IDO}}(A_1')$, then since $v_1^{1,\mathsf{IDO}}(B_1) = 1 - v_1^{1,\mathsf{IDO}}(B_2)$ and $v_1^{1,\mathsf{IDO}}(A_1') = 1 - v_1^{1,\mathsf{IDO}}(A_2')$, we get

$$v_1^{1,\mathsf{IDO}}(A_2') \geq v_1^{1,\mathsf{IDO}}(B_2).$$

Combining this with the facts we know, we get

$$v_1^{1,\mathsf{IDO}}(A_2') \geq v_1^{1,\mathsf{IDO}}(B_2) \geq v_1^{1,\mathsf{IDO}}(B_1) \geq v_1^{1,\mathsf{IDO}}(A_1').$$

Since all the above terms are at least 0, we get that Thus, $|v_1^{1,\mathsf{IDO}}(B_1) - v_1^{1,\mathsf{IDO}}(B_2)| < |v_1^{1,\mathsf{IDO}}(A_1') - v_1^{1,\mathsf{IDO}}(A_2')|$, which contradicts the way (A_1', A_2') is chosen. Thus, we assume $v_1^{1,\mathsf{IDO}}(B_1) \leq v_1^{1,\mathsf{IDO}}(A_1')$. However, note that by definition of B_1, we get that $v_1^{1,\mathsf{IDO}}(B_1) \geq v_1(A_1)$. Together with (3.19), we get

$$v_1^{1,\mathsf{IDO}}(B_1) \geq v_1(A_1) \geq v_1^{1,\mathsf{IDO}}(A_1').$$

This means that $v_1^{1,\mathsf{IDO}}(B_1) = v_1^{1,\mathsf{IDO}}(A_1')$. However, note that by definition of B_1 and the way Algorithm 2 works, $|B_1| > |A_1| = |A_1'|$, giving us another contradiction.

In both cases, we arrive at a contradiction, and our result follows. □

Together with the results of Section 3.4, this raises a natural follow-up question: does an EF1 allocation exist for $n \geq 3$ given frequency predictions? Despite considerable effort, this question remains elusive and unresolved. We conjecture that no online algorithm can guarantee EF1 in this setting, even with frequency predictions—though achieving a positive competitive ratio may still be possible. We leave this as an open problem.

To end this section, we complement the positive result of Theorem 3.5.5 by showing a strong impossibility result with respect to EFX for any higher values of n, even with frequency predictions.

Theorem 3.5.6. *For $n \geq 3$, given frequency predictions, there does not exist any online algorithm with a competitive ratio strictly larger than 0 with respect to* EFX.

Proof. Suppose for a contradiction that there exists a γ-competitive algorithm for approximating EFX when $n \geq 3$ and with frequency predictions, for some $\gamma \in (0,1]$. Let $\varepsilon > 0$ be a small constant and $K > 0$ be sufficiently large. Let $V_i = \{K^2, \ldots, K^2, K, \varepsilon, \varepsilon\}$ be the multiset with $n+1$ elements for each $i \in N$. Let the $n+1$ goods be $g_1, \ldots, g_{n+1}$ with valuations as follows:

$\mathbf{v}$	g_1	g_2	g_3	g_4	$\cdots$	g_n	g_{n+1}
1	K	K^2	ε	K^2	$\ldots$	K^2	ε
2	K	ε	ε	K^2	$\ldots$	K^2	K^2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ldots$	$\vdots$	$\vdots$
n	K	ε	ε	K^2	$\ldots$	K^2	K^2

Suppose w.l.o.g. that g_1 is allocated to agent 1. We split our analysis into two cases.

Case 1: g_2 is allocated to agent 1. Then no goods in $\{g_3, \ldots, g_{n+1}\}$ can be allocated to agent 1 (otherwise there must exist some agent in $N \setminus \{1\}$ that receives no goods, leading to a 0-EFX allocation). Moreover, if some agent in $N \setminus \{1\}$ receives two goods from $\{g_3, \ldots, g_{n+1}\}$, then there must exist some agent other agent from $N \setminus \{1\}$ that receives no goods, leading to a 0-EFX allocation. Thus, we must have that each good in $\{g_3, \ldots, g_{n+1}\}$ is allocated to a different agent in $N \setminus \{1\}$. Let the agent that is allocated g_3 be i. Then, $v_i(A_i) = \varepsilon$ and $v_i(A_1 \setminus \{g_2\}) = K$. This gives us $\gamma \leq \frac{\varepsilon}{K} \to 0$ as $K \to \infty$, a contradiction.

Case 2: g_2 is not allocated to agent 1. Suppose w.l.o.g. that g_2 is allocated to agent 2 instead. If at least one of $\{g_3, \ldots, g_{n+1}\}$ is allocated to agent 2 and agent 1 is not allocated any good from $\{g_4, \ldots, g_n\}$, then $v_1(A_1) \leq K + 2\varepsilon$ and $v_1(A_2 \setminus \{g\}) \geq K^2$ for $g \in \arg\min_{g' \in A_2} v_1(g')$. This gives us

$$\gamma \leq \frac{K + 2\varepsilon}{K^2} = \frac{1}{K} + \frac{2\varepsilon}{K^2} \to 0$$

as $K \to \infty$, a contradiction. Therefore, if at least one of $\{g_3, \ldots, g_{n+1}\}$ is allocated to agent 2, agent 1 must be allocated some good from $\{g_4, \ldots, g_n\}$. However, this means that there are at most $n-3$ remaining goods and $n-2$ agents, leaving some agent in $N \setminus \{1, 2\}$ with an empty bundle, leading to a 0-EFX allocation. Thus, none of the goods in $\{g_3, \ldots, g_{n+1}\}$ can be allocated to agent 2. This implies that there must be some agent $i \in N \setminus \{2\}$ that receives two goods eventually. However, $v_2(A_2) = \varepsilon$ and $v_2(A_i \setminus \{g\}) \geq K$ for $g \in \arg\min_{g' \in A_i} v_2(g')$. This gives us $\gamma \leq \frac{\varepsilon}{K} \to 0$ as $K \to \infty$, a contradiction.

We arrived at a contradiction in both cases, thereby proving our claim. $\square$

These negative results highlight a significant gap that remains between the offline (single-shot) and online settings: in the former, while an exact EFX allocation is guaranteed to exist for $n = 3$ [Chaudhury et al., 2024] and an 0.618-EFX allocation exists in general for all n [Amanatidis et al., 2020], no comparable guarantees hold in the online setting—even with frequency predictions.

3.6 Identical Valuations

Finally, we also study the special case where all agents have *identical valuation functions*. This setting is motivated by both theoretical and practical considerations. From a theoretical perspective, this case serves as a natural benchmark, and aligns with a rich line of work in fair division that has focused on this restricted but structurally appealing setting [Barman and Sundaram, 2020, Elkind et al., 2025a, Mutzari et al., 2023, Plaut and Roughgarden, 2020]. From an applied perspective, this setting captures scenarios where agents value goods based on uniform or shared criteria (e.g., identical machines processing uniform jobs), and is especially relevant in the context of online multiprocessor scheduling. In particular, a large body of work on semi-online scheduling [Cheng et al., 2005, Kellerer et al., 1997] considers the goal of minimizing the makespan—the maximum load assigned to any processor—when tasks arrive sequentially (analogous to MMS in our setting). The majority of these models (and other variants in the online scheduling literature) assume identical valuations (i.e., identical machines) (refer to the survey by Dwibedy and Mohanty [2022]). In particular, it is known that in this setting, one can achieve $\frac{2}{3}$-MMS for $n = 2$ [Kellerer et al., 1997] and $\frac{1}{n-1}$-MMS for $n \geq 3$ [Tan and Wu, 2007].

We further study the fairness guarantees achievable in this setting beyond MMS. Observe that, under identical valuations with frequency predictions, the problem collapses to the classic single-shot offline fair-division setting—so any offline guarantee (for example, the existence of EFX allocations for any number of agents) immediately applies here. Consequently, we focus on the two remaining regimes—no information and normalization information. Importantly, even though each agent's valuation for any given good is the same, the precise value of each future arrival remains unknown.

In the single-shot setting, it is known that EFX allocations always exist (and can be computed in polynomial time) under identical valuations for any number of agents [Plaut and Roughgarden, 2020]. On the positive side, the algorithm of Elkind et al. [2025a, Theorem 3.7] (which allocates the next arriving good to an agent with the least bundle value so far) shows that EF1 allocations can be found for identical valuations in the online setting with no future information, which we reproduce here for completeness.

Proposition 3.6.1 (Elkind et al. [2025a], Theorem 3.7)**.** *For $n \geq 2$, under identical valuations and without future information, there exists an online algorithm that always returns an* EF1 *allocation.*

We complement these known positive results with an impossibility result for EFX. Thus, even in this setting with identical valuations, the lack of future information can still limit fairness guarantees.

Proposition 3.6.2. *For $n \geq 2$, under identical valuations and without future information, there does not exist any online algorithm with a competitive ratio strictly larger than* 0 *with respect to* EFX.

Proof. Suppose for a contradiction that there exists a γ-competitive algorithm for approximating EFX under identical valuations, for some $\gamma \in (0, 1]$. Let $K > 0$ be a large constant. Consider the case of $n = 2$, and let the first two goods be g_1 and g_2 with $v(g_1) = v(g_2) = 1$. W.l.o.g., suppose g_1 is allocated to agent 1. If g_2 is also allocated to agent 1, let the third (and final) good be g_3 with $v(g_3) = 0$. We have that $v(A_2) = 0$, giving us a 0-EFX allocation. Thus, g_2 has to be allocated to agent 2. Then, let the third (and final) good be g_3 with $v(g_3) = K$. W.l.o.g., suppose g_3 is allocated to agent 1. Then, $v(A_2) = 1$ and $v(A_1 \setminus \{g_1\}) = K$, giving us

$$\gamma \leq \frac{1}{K} \to 0$$

as $K \to \infty$, a contradiction. For the case of $n \geq 3$, refer to the proof of Theorem 3.6.4 (where we give a counterexample that assumes identical valuations and normalization information—the latter is defined in the next section). □

Theorem 3.4.3 and Proposition 3.6.2 essentially give strong impossibility results for achieving approximate EFX even in the case of two agents, even if we assume either normalization information or identical valuations, respectively. In what follows, we consider the setting where we have *both* normalization information and identical valuations. For two agents, we show that while an EFX allocation still may fail to exist, we can obtain a $\frac{\sqrt{5}-1}{2}$-EFX allocation, and also prove that this bound is tight (i.e., no online algorithm can do better than this given the assumptions).

Theorem 3.6.3. *For $n = 2$, under identical valuations and given normalization information, there exists an online algorithm that returns a* $\frac{\sqrt{5}-1}{2}$-EFX *allocation. Moreover, in this setting, there does not exist any online algorithm with a competitive ratio strictly larger than* $\frac{\sqrt{5}-1}{2}$ *with respect to* EFX.

Proof. We first show the upper bound. Suppose for a contradiction that there exists a γ-competitive algorithm for approximating EFX when $n = 2$, for some $\gamma > \frac{\sqrt{5}-1}{2}$. Let $k = \lceil \frac{\sqrt{5}-2}{\varepsilon} \rceil$, and let the first k goods be $g_1, \ldots, g_k$ with values $v(g_1) = \cdots = v(g_k) = \varepsilon$. If each agent is allocated at least one of these k goods, let the $(k+1)$-th (and final) good be g_{k+1} with $v(g_{k+1}) = 1 - (\sqrt{5} - 2) = 3 - \sqrt{5}$. Suppose g_{k+1} is allocated to agent $i \in \{1, 2\}$, and let the other agent be j. We have that $v(A_j) < \sqrt{5} - 2$ and $v(A_i \setminus \{g\}) \geq 3 - \sqrt{5}$ for $g \in \arg\min_{g' \in A_i} v(g')$. Consequently, we obtain

$$\gamma \leq \frac{\sqrt{5}-2}{3-\sqrt{5}} = \frac{\sqrt{5}-1}{4},$$

a contradiction. Thus, all k goods must be allocated to a single agent. W.l.o.g., let it be agent 1. Then, let the next two (and final) goods be g_{k+1} and g_{k+2} with $v(g_{k+1}) = v(g_{k+2}) = \frac{1-k\varepsilon}{2}$. If g_{k+1} is allocated to agent 1, then $v(A_2) \leq \frac{1-k\varepsilon}{2}$ and $v(A_1 \setminus \{g\}) = (k-1) \cdot \varepsilon + \frac{1-k\varepsilon}{2}$ where $g \in \arg\min_{g' \in A_1} v(g')$. We get that

$$\gamma \leq \frac{\frac{1-k\varepsilon}{2}}{(k-1) \cdot \varepsilon + \frac{1-k\varepsilon}{2}} = \frac{1-k\varepsilon}{k\varepsilon - 2\varepsilon + 1}. \tag{3.23}$$

Since $\frac{\sqrt{5}-2}{\varepsilon} \leq k < \frac{\sqrt{5}-2}{\varepsilon} + 1$, multiplying by ε throughout gives us

$$\sqrt{5} - 2 \leq k\varepsilon \leq \sqrt{5} - 2 + \varepsilon.$$

By the squeeze theorem, we get $\lim_{\varepsilon \to 0} k\varepsilon = \sqrt{5} - 2$. Thus applying this to (3.23) gives us

$$\gamma \leq \frac{1-k\varepsilon}{k\varepsilon - 2\varepsilon + 1} \to \frac{1-\sqrt{5}+2}{\sqrt{5}-2+1} = \frac{3-\sqrt{5}}{\sqrt{5}-1} = \frac{\sqrt{5}-1}{2},$$

as $\varepsilon \to 0$, a contradiction.

Next, we prove the lower bound. Consider the allocation $\mathcal{A}$ returned by the following algorithm (Algorithm 3).

We have that

$$v(A_2) \geq 1 - \frac{\sqrt{5}-1}{2} = \frac{3-\sqrt{5}}{2} \quad \text{and} \quad v(A_1) \leq \frac{\sqrt{5}-1}{2}.$$

Then, we get that

$$v(A_2) \geq \frac{3-\sqrt{5}}{2} = \left(\frac{\sqrt{5}-1}{2}\right)^2 \geq \frac{\sqrt{5}-1}{2} \cdot v(A_1). \tag{3.24}$$

Algorithm 3: Returns a $\frac{(\sqrt{5}-1)}{2}$-EFX allocation for $n = 2$, given normalization information, and agents have identical valuations

```
1: Initialize 𝒜 = (A_1, A_2) = (∅, ∅)
2: while there exists a good g that arrives do
3:     if v(A_1 ∪ {g}) ≤ (√5−1)/2 then
4:         A_1 ← A_1 ∪ {g}
5:     else
6:         A_2 ← A_2 ∪ {g}
7:     end if
8: end while
9: return 𝒜 = (A_1, A_2)
```

By Lines 3 and 4 of the algorithm, we get that for all $g \in A_2$, $v(A_1 \cup \{g\}) > \frac{\sqrt{5}-1}{2}$, giving us

$$v(g) > \frac{\sqrt{5}-1}{2} - v(A_1). \tag{3.25}$$

We split our analysis into two cases.

Case 1: $v(A_1) < \sqrt{5} - 2$. Then, algebraic manipulation gives us

$$v(A_2) = 1 - v(A_1) < 2 \times \left(\frac{\sqrt{5}-1}{2} - v(A_1)\right) < 2 \times \min_{g \in A_2} v(g),$$

where the rightmost inequality follows from the fact that (3.25) holds for all $g \in A_2$ (in particular, the minimum-valued one). This means that $|A_2| = 1$ and $v(A_1) \geq 0 = v(A_2 \setminus \{g\})$ for any $g \in A_2$. Together with (3.24), the result follows.

Case 2: $v(A_1) \geq \sqrt{5} - 2$. Then, we have that for any $g \in A_2$,

$$\begin{aligned}
\frac{v(A_1)}{v(A_2 \setminus \{g\})} &= \frac{v(A_1)}{v(A_2) - v(g)} \\
&> \frac{v(A_1)}{(1 - v(A_1)) - \left(\frac{\sqrt{5}-1}{2} - v(A_1)\right)} \qquad \text{(by (3.24))} \\
&\geq \frac{\sqrt{5}-2}{1 - \frac{\sqrt{5}-1}{2}} \qquad \text{(by the assumption of this case)} \\
&= \frac{\sqrt{5}-1}{2}.
\end{aligned}$$

This gives us $v(A_1) > \frac{\sqrt{5}-1}{2} \cdot v(A_2 \setminus \{g\})$ for any $g \in A_2$. Together with (3.24), the result follows.

Thus, the algorithm gives us a $\frac{\sqrt{5}-1}{2}$-EFX allocation, and a lower bound in this setting. $\square$

However, the strong impossibility result still persists for $n \geq 3$.

Theorem 3.6.4. *For $n \geq 3$, under identical valuations and given normalization information, there does not exist any online algorithm with a competitive ratio strictly larger than* 0 *with respect to* EFX.

Proof. Suppose for a contradiction that there exists a γ-competitive algorithm for approximating EFX when $n \geq 3$, with normalization information and identical valuations, for some $\gamma \in (0, 1]$. Let $\varepsilon > 0$ be a sufficiently small constant. Let the first two goods be g_1 and g_2, with $v(g_1) = v(g_2) = \varepsilon$. W.l.o.g. suppose g_1 is allocated to agent 1. Then, if g_2 is allocated to agent 1, let the third (and final) good be g_3 with $v(g_3) = 1 - 2\varepsilon$. There must exist an agent that does not receive any good—let this be agent $i \in N \setminus \{1\}$. We have that $v(A_i) = 0$ and $v(A_1 \setminus \{g_1\}) = \varepsilon$, giving us $\gamma \leq \frac{0}{\varepsilon} = 0$ for any positive $\varepsilon > 0$, a contradiction.

Thus, g_2 cannot be allocated to agent 1. Let the next $n - 1$ goods be $g_3, \ldots, g_{n+1}$ with $v(g_k) = \frac{1-2\varepsilon}{n-1}$ for all $k \in \{3, \ldots, n + 1\}$. Let $i \in N$ be the agent that does not receive any good from $\{g_3, \ldots, g_{n+1}\}$, and $j \in N \setminus \{i\}$ be the agent that receives some good from $\{g_3, \ldots, g_{n+1}\}$ and $|A_j| \geq 2$. Then, $v(A_i) \leq \varepsilon$ and $v(A_j \setminus \{g\}) \geq \frac{1-2\varepsilon}{n-1}$ for $g \in \arg\min_{g' \in A_j} v(g')$. Consequently, we get that

$$\gamma \leq \frac{\varepsilon}{\frac{1-2\varepsilon}{n-1}} = \frac{(n-1)\varepsilon}{1-2\varepsilon} \to 0$$

as $\varepsilon \to 0$, arriving at a contradiction. $\square$

3.7 Conclusion

We have explored the existence of standard notions of fairness in the online fair division setting with and without additional information. In the absence of future information, our results indicate that online algorithms cannot achieve fair outcomes. However, when the online algorithm is given normalization information or frequency predictions, (relatively) stronger fairness guarantees are possible. Under normalization information, we propose an algorithm is EF1 and 0.5-MMS for $n = 2$, and PROP1 for any n. Given frequency predictions, we introduce a meta-algorithm that leverages frequency predictions to match the best-known offline guarantees for a broad class of "share-based" fairness notions. Our complementary impossibility results across the multiple settings emphasizes the limitations of these additional information, despite its relative power.

Beyond the open questions raised in our work, several promising directions remain. One of these directions is to build on our framework in the spirit of *online algorithms with predictions*, e.g., by analyzing the robustness of fairness guarantees when the provided information is noisy or imperfect. Studying how algorithmic performance degrades with prediction errors could yield insights into designing more robust algorithms in this setting.

Chapter 4

Temporal Fair Division

4.1 Overview

The standard model of online fair division assumes that the valuation of each item is revealed only upon its arrival. As discussed in the previous chapter, it is impossible to guarantee a complete EF1 (envy-free up to one item) allocation of goods in the absence of any information about future items [Benadè et al., 2024]. However, we have also seen that access to partial information about the future can significantly improve fairness guarantees.

In this chapter, we take this idea to its logical extreme and study the *informed online fair division* model, also referred to as *temporal fair division*, where the algorithm is assumed to have *complete information* about all items' valuations and their arrival order in advance. Of course, with full information, one can trivially obtain an EF1 allocation at the end of the process by treating the instance as an offline problem and applying any known EF1 algorithm (e.g., Lipton et al. [2004], Aziz et al. [2019], Caragiannis et al. [2019]). However, this static approach disregards the dynamic nature of online allocation. In particular, it fails to address the temporal fairness of intermediate allocations—agents may experience long stretches during which their cumulative bundles appear significantly less favorable than others, leading to sustained feelings of unfairness.

To address this, we introduce the notion of *temporal EF1* (TEF1), a dynamic extension of EF1 adapted to the informed online setting. Notably, we also extend our analysis to that for *chores* (i.e., items that are valued nonpositively by agents). TEF1 requires that the allocation at every round is EF1 with respect to the cumulative bundles of items allocated up to that point.

This concept has also been studied by He et al. [2019] for the allocation of goods. Similar to our paper, their objective is to ensure that EF1 is satisfied at each round, but they allow agents to *swap* their bundles, thereby relaxing the "irrevocable" assumption of the standard online fair division model.

When the valuations are known upfront for the allocation of chores, the model is similar to the field of work on *job scheduling*. There have been numerous papers studying fair scheduling, but the fairness is typically represented by an objective function which the algorithm aims to minimize or approximate [Schwiegelshohn and Yahyapour, 2000, Im and Moseley, 2020, Baruah, 1995]. On the other hand, there is little work on satisfying envy-based notions in scheduling problems, but Li et al. [2021] study the compatibility of EF1 and Pareto optimality in various settings. While we consider separately the cases of goods allocation and chores allocation, to the best of our knowledge, there is no prior work which studies an online fair division model with both goods and chores in the same instance under any information assumption.

The central goal of this chapter is to explore the existence and computational complexity of achieving TEF1 allocations in both goods and chores settings. Specifically, we seek to answer the following questions:

> *Which restricted settings guarantee the existence of a TEF1 allocation, and can we compute such an allocation in polynomial time in these settings? Is it computationally tractable to determine the existence*[1] *of a TEF1 allocation? In terms of existence and tractability, is TEF1 compatible with natural notions of efficiency?*

In Section 4.3, we establish the existence of TEF1 allocations in several restricted settings—including the case of two agents, settings with only two types of items, instances with generalized binary valuations, and scenarios where agents exhibit unimodal preferences. For each setting, we present a polynomial-time algorithm that computes a TEF1 allocation. Despite these positive results, we also show that there exist significant computational barriers. For goods, we show that determining whether a TEF1 allocation exists is NP-hard. For chores, we prove that even if one is given a partial TEF1 allocation, it is NP-hard to decide whether it can be extended to a complete TEF1 allocation over all remaining chores.

In Section 4.4, we turn our attention to efficiency. Although TEF1 allocations are guaranteed to exist and can be computed in polynomial time for two agents (in both goods and chores settings), we show that this guarantee cannot be upheld when we additionally require Pareto-optimality (PO). Moreover, we prove that even in this

[1] Prior work by He et al. [2019] has shown that a TEF1 allocation is not guaranteed to exist for goods in the general setting with three or more agents.

simple two-agent case, it is NP-hard to determine whether a TEF1 and PO allocation exists. As a corollary, we obtain the computational intractability of checking whether there exists a TEF1 allocation that maximizes any p-mean welfare objective, a broad class that includes most commonly studied social welfare criteria.

Finally, in Section 4.5, we consider the special case where the same set of items arrive at each round. We show that even in this restricted case, it is NP-hard to determine whether repeating a particular allocation across two consecutive rounds yields a TEF1 allocation. Nonetheless, we complement this hardness result with a positive finding: for the special case of two rounds, we provide a polynomial-time algorithm for computing a TEF1 allocation.

4.2 Preliminaries

In this chapter, we also extend our analysis to the case of chores—items that agents have nonpositive utilities for. This requires a generalization of several standard notation to accommodate negative utility values and capture fairness appropriately in this context.

An *instance* of our problem can be represented as a tuple $\mathcal{I} = \langle N, T, \{O_t\}_{t\in[T]}, \mathbf{v} = (v_1, \ldots, v_n)\rangle$, where $N = [n]$ is a set of *agents*, T is the number of *rounds*, for each $t \in [T]$ the set O_t consists of items that arrive at round t, with $O = \cup_{t\in[T]} O_t$, and for each $i \in N$ the *valuation function* $v_i : O \to \mathbb{R}$ specifies the values that agent i assigns to items in O.

We assume that agents have additive valuations, i.e., we extend the functions v_i to subsets of O by setting $v_i(S) = \sum_{o\in S} v_i(O)$ for each $S \subseteq O$. We define the *cumulative set of items* that arrive in rounds $1, \ldots, t$ by $O^t := \bigcup_{\ell\in[t]} O_\ell$. Note that $O = O^T$.

We consider both *goods allocation*, where $v_i(o) \geq 0$ for each $i \in N$ and $o \in O$, and *chores allocation*, where $v_i(o) \leq 0$ for each $i \in N$ and $o \in O$. For clarity, in the goods setting we use g instead of o and refer to the items as *goods*, while in the chores setting we use c instead of o and refer to the items as *chores*.

An *allocation* $\mathcal{A} = (A_1, \ldots, A_n)$ of items in O to the agents is an ordered partition of O, i.e., $\bigcup_{i\in N} A_i = O$ and $A_i \cap A_j = \varnothing$ for all $i, j \in N$ with $i \neq j$. For $t \in [T]$, $i \in N$ we write $A_i^t = A_i \cap O^t$; then $\mathcal{A}^t = (A_1^t, \ldots, A_n^t)$ is the allocation after round t, with $\mathcal{A} = \mathcal{A}^T$. For $t < T$, we may refer to $\mathcal{A}^t$ as a *partial allocation*.

Our goal is to find an allocation that is fair after each round. The main fairness notion that we consider is EF1 (as defined in Chapter 2 for goods). However, we restate its definition here, since we are defining it for chores as well.

Definition 4.2.1. In a goods (resp., chores) allocation instance, an allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is said to be EF1 if for each pair of agents $i, j \in N$, either $A_i = \varnothing$ or there exists a good $g \in A_j$ (resp., chore $c \in A_i$) such that $v_i(A_i) \geq v_i(A_j \setminus \{g\})$ (resp. $v_i(A_i \setminus \{c\}) \geq v_i(A_j)$).

To capture fairness in a *cumulative* sense, we introduce the notion of *temporal envy-freeness up to one item (TEF1)*, which requires that at every prefix of rounds the cumulative allocation of items that have arrived so far satisfies EF1.

Definition 4.2.2 (Temporal EF1)**.** For every $t \in [T]$, an allocation $\mathcal{A}^t = (A^t_1, \ldots, A^t_n)$ is said to be *temporally envy-free up to one item (TEF1)* if for each $\ell \in [t]$ the allocation $\mathcal{A}^\ell$ is EF1.

A key distinction between TEF1 and EF1 is that, while the EF1 property only places constraints on the final allocation, TEF1 requires envy-freeness up to one item at every round.

However, He et al. [2019, Thm. 4.2] show that for goods TEF1 allocations may fail to exist; they present an example with 3 agents and 23 items, which can be generalized to $n > 3$ agents. We remark that the construction of He et al. [2019] cannot be translated to the chores setting. Indeed, while we conjecture that a non-existence result of this form also holds for chores, this remains an open question.

4.3 On the Existence of TEF1 Allocations

As some instances do not admit TEF1 allocations, our first goal is to explore if there are restricted classes of instances for which TEF1 allocations are guaranteed to exist. In this section we identify several such settings.

To simplify the presentation, we will first demonstrate that it usually suffices to consider instances where only one item appears at each round (i.e., $T = m$ and $|O_t| = 1$ for all $t \in [T]$). Indeed, any impossibility result for this special setting also holds for the general case, and we will now argue that the converse is true as well.

Lemma 4.3.1. *Given an instance $\mathcal{I}$ with $|O| = m$ items, we can construct an instance $\mathcal{I}^{=1}$ with the same set of items and exactly m rounds so that $|O_t| = 1$ for each $t \in [m]$ and if $\mathcal{I}^{=1}$ admits a* TEF1 *allocation, then so does $\mathcal{I}$.*

Proof. Consider an arbitrary instance $\mathcal{I} = \langle N, T, \{O_t\}_{t\in[T]}, \mathbf{v} = (v_1, \ldots, v_n)\rangle$. Renumber the items in a non-decreasing fashion with respect to the rounds, so that for any two rounds $t, r \in [T]$ with $t < r$ and items $o_j \in O_t, o_{j'} \in O_r$ it holds that $j < j'$. We construct $\mathcal{I}^{=1} = \langle N, m, \{\widetilde{O}_t\}_{t\in[m]}, \mathbf{v}\rangle$ by setting $\widetilde{O}_t = \{o_t\}$ for each $t \in [m]$. Let $\mathcal{A}$ be a TEF1 allocation for $\mathcal{I}^{=1}$. We construct an allocation $\mathcal{B}$ for instance $\mathcal{I}$ by allocating all items in the same way as in $\mathcal{A}$: if $\mathcal{A}$ allocates at item j to agent i in round r, we identify a $t \in [T]$ such that $\sum_{\ell=1}^{t-1} |O_\ell| < r \leq \sum_{\ell=1}^{t} |O_\ell|$ and place j into B_i in round t. To see that $\mathcal{B}$ satisfies TEF1, note that if $\mathcal{B}^t$ violates EF1 for some $t \in [T]$, then for $r = \sum_{\ell=1}^{t} |O_\ell|$ the allocation $\mathcal{A}^r$ satisfies $A_i^r = B_i^t$ for all $i \in N$ and hence violates EF1 as well. $\square$

In what follows, unless specified otherwise, we simplify the notation based on the transformation in the proof of Lemma 4.3.1: we assume that $|O_t| = 1$ for each $t \in T$ and denote the unique item that arrives in round t by o_t (or g_t, or c_t, if we focus on goods/chores).

4.3.1 Two Agents

He et al. [2019, Thm. 3.4] put forward a polynomial-time algorithm that always outputs a TEF1 allocation for goods when $n = 2$; in particular, this implies that a TEF1 allocation is guaranteed to exist for $n = 2$. We will now extend this result to the case of chores.

Intuitively, in each round the algorithm greedily allocates the unique chore that arrives in that round to an agent that does not envy the other agent in the current (partial) allocation. A counter s keeps track of the last round in which $\mathcal{A}^s$ was envy-free; if for some round $t \in [m]$ the allocation of a chore c_t results in both agents envying each other in $\mathcal{A}^t \setminus \mathcal{A}^s$, then the agents' bundles in $\mathcal{A}^t \setminus \mathcal{A}^s$ are swapped.

Theorem 4.3.2. *For $n = 2$, Algorithm 4 returns a* TEF1 *allocation for chores, and runs in polynomial time.*

Proof. The polynomial runtime of Algorithm 4 is easy to verify: there is only one **for** loop, with a counter that runs from 1 to m, and each operation within the loop runs in $\mathcal{O}(m)$ time. Thus, we focus on proving correctness.

For each $t \in [m]$, we define $r_t < t$ as the latest round before t such that $\mathcal{A}^{r_t}$ is EF. This implies that if $\mathcal{A}^\ell \setminus \mathcal{A}^{r_t}$ is EF1 for all $\ell = r_t, r_t + 1, \ldots, t$, then $\mathcal{A}^t$ is also EF1. Therefore, it suffices to show that $\mathcal{A}^t \setminus \mathcal{A}^{r_t}$ is EF1 for each $t \in [m]$. We will prove this by induction on t.

For $t = 1$, the claim is immediate, as any allocation of a single chore is EF1. Now, suppose that $t > 1$. If $t = r_t + 1$ the allocation $\mathcal{A}^t \setminus \mathcal{A}^{r_t}$ consists of a single chore, so, again, the claim is immediate. Otherwise, $r_{t-1} = r_t$ and by the induction hypothesis

Algorithm 4: Returns a TEF1 allocation for chores when $n = 2$

Input: Set of agents $N = \{1, \dots, n\}$, set of chores $O = \{c_1, \dots, c_m\}$, and valuation profile $\mathbf{v} = (v_1, v_2)$

Output: TEF1 allocation $\mathcal{A}$ of chores in O to agents in N

1: Initialize $s \leftarrow 0$ and $\mathcal{A}^0 \leftarrow (\varnothing, \varnothing)$
2: **for** $t = 1, 2, \dots, m$ **do**
3:     **if** $v_1(A_1^{t-1} \setminus A_1^s) \geq v_1(A_2^{t-1} \setminus A_2^s)$ **then**
4:         $\mathcal{A}^t \leftarrow (A_1^{t-1} \cup \{c_t\}, A_2^{t-1})$
5:     **else**
6:         $\mathcal{A}^t \leftarrow (A_1^{t-1}, A_2^{t-1} \cup \{c_t\})$
7:     **end if**
8:     **if** $v_1(A_1^t \setminus A_1^s) < v_1(A_2^t \setminus A_2^s)$ and $v_2(A_2^t \setminus A_2^s) < v_2(A_1^t \setminus A_1^s)$ **then**
9:         $\mathcal{A}^t \leftarrow (A_1^s \cup A_2^t \setminus A_2^s, A_2^s \cup A_1^t \setminus A_1^s)$
10:     **end if**
11:     **if** $v_1(A_1^t \setminus A_1^s) \geq v_1(A_2^t \setminus A_2^s)$ and $v_2(A_2^t \setminus A_2^s) \geq v_2(A_1^t \setminus A_1^s)$ **then**
12:         $s \leftarrow t$
13:     **end if**
14: **end for**
15: **return** $\mathcal{A} = (A_1^m, A_2^m)$

it holds that $\mathcal{A}^{t-1} \setminus \mathcal{A}^{r_t}$ is EF1. Let r'_t be the earliest round ahead of r_t such that $\mathcal{A}^{r'_t} \setminus \mathcal{A}^{r_t}$ is EF (if such a round exists). We divide the remainder of the proof into two cases depending on whether a partial bundle swap (as in line 9 of the algorithm) occurs at round r'_t.

Case 1: Round r'_t does not exist or no swap at round r'_t. Suppose without loss of generality that $v_1(A_1^{t-1} \setminus A_1^{r_t}) < v_1(A_2^{t-1} \setminus A_2^{r_t})$, i.e., agent 1 envies agent 2 in $\mathcal{A}^{t-1} \setminus \mathcal{A}^{r_t}$. Then agent 2 does not envy agent 1 (otherwise we would swap the bundles, contradicting the definition of r_t), and consequently receives c_t. If agent 2 envies agent 1 after receiving c_t, this envy can be removed by removing c_t. We also know that $\mathcal{A}^t \setminus \mathcal{A}^{r_t}$ is EF1 w.r.t. agent 1 (who did not receive a chore in round t) because by our inductive assumption, $\mathcal{A}^{t-1} \setminus \mathcal{A}^{r_t}$ is EF1, concluding the proof of this case.

Case 2: Swap occurs at round r'_t. We assume that $r'_t > t$, because if $r'_t = t$, then $\mathcal{A}^t \setminus \mathcal{A}^{r_t}$ is EF and therefore EF1. For each $i \in \{t-1, t\}$, let $\mathcal{B}^i \setminus \mathcal{A}^s$ refer to the algorithm's allocation of the chores $O^i \setminus O^s$ *before* the bundle swap, and suppose without loss of generality that $v_1(B_1^{t-1} \setminus B_1^{r_t}) < v_1(B_2^{t-1} \setminus B_2^{r_t})$. We therefore must have $v_2(B_2^{t-1} \setminus B_2^{r_t}) \geq v_2(B_1^{t-1} \setminus B_1^{r_t})$ to avoid contradicting the definition of r_t. Since agent 2 is not envied by agent 1 in round $t-1$, it receives chore c_t, so we have $\mathcal{B}^t \setminus \mathcal{B}^{r_t} = (B_1^{t-1} \setminus B_1^{r_t}, (B_2^{t-1} \setminus B_2^{r_t}) \cup \{c_t\})$. This means that after the bundle swap is executed, we have $\mathcal{A}^t \setminus \mathcal{A}^{r_t} = ((B_2^{t-1} \setminus B_2^{r_t}) \cup \{c_t\}, B_1^{t-1} \setminus B_1^{r_t})$. Recall that $v_1(B_2^{t-1} \setminus B_2^{r_t}) > v_1(B_1^{t-1} \setminus B_1^{r_t})$, so c_t can be removed from agent 1's bundle to eliminate their envy towards agent 2. Also, by the inductive assumption, there exists a chore $c \in A_2^{t-1} \setminus A_2^{r_t}$ such that $v_2((A_2^{t-1} \setminus A_2^{r_t}) \setminus \{c\}) \geq v_2(A_1^{t-1} \setminus A_1^{r_t})$. Observe that $A_2^t \setminus A_2^{r_t} = A_2^{t-1} \setminus A_2^{r_t}$ and $A_1^t \setminus A_1^{r_t} = (A_1^{t-1} \setminus A_1^{r_t}) \cup \{c_t\}$. Combining this with the inductive assumption, we have that there exists a chore $c \in A_2^t \setminus A_2^{r_t}$ such that

$$\begin{aligned} v_2((A_2^t \setminus A_2^{r_t}) \setminus \{c\}) &= v_2((A_2^{t-1} \setminus A_2^{r_t}) \setminus \{c\}) \\ &\geq v_2(A_1^{t-1} \setminus A_1^{r_t}) \\ &\geq v_2((A_1^{t-1} \setminus A_1^{r_t}) \cup \{c_t\}) \\ &= v_2(A_1^t \setminus A_1^{r_t}). \end{aligned}$$

Therefore, $\mathcal{A}^t \setminus \mathcal{A}^{r_t}$ is EF1 in this case.

We have shown that $\mathcal{A}^t \setminus \mathcal{A}^{r_t}$ is EF1 regardless of whether the allocation has undergone a bundle swap, so by induction, Algorithm 4 returns a TEF1 allocation for chores. □

Next, we consider *temporal envy-freeness up to any item (TEFX)*, the temporal variant of the stronger notion of *envy-freeness up to any item (EFX)*. We restate the definition of EFX here (compared to the one in Chapter 2), since we are defining it for chores as well.

Definition 4.3.3. In a goods (resp., chores) allocation instance, an allocation $\mathcal{A} = (A_1, \ldots, A_n)$ is said to be EFX if for all pairs of agents $i, j \in N$, either $A_i = \varnothing$ or and all goods $g \in A_j$ (resp., chores $c \in A_i$) we have $v_i(A_i) \geq v_i(A_j \setminus \{g\})$ (resp. $v_i(A_i \setminus \{c\}) \geq v_i(A_j)$).

Definition 4.3.4 (Temporal EFX)**.** For every $t \in [T]$, an allocation $\mathcal{A}^t = (A_1^t, \ldots, A_n^t)$ is said to be *temporal envy-free up to any item (TEFX)* if for each $\ell \leq t$ the allocation $\mathcal{A}^\ell$ is EFX.

Unfortunately, TEFX allocations (for goods or chores) may not exist, even for two agents with identical valuations, and even when there are only two types of items.

Proposition 4.3.5. *A* TEFX *allocation for goods or chores may not exist, even for $n = 2$ with identical valuations and two types of items.*

Proof. We first prove the result for the case of goods. Consider the instance with two agents $N = \{1, 2\}$ and three goods $O = \{g_1, g_2, g_3\}$, where agents have identical valuations: $v(g_1) = v(g_2) = 1$ and $v(g_3) = 2$. In order for the partial allocation at the end of the second round to be TEFX, each agent must be allocated exactly one of $\{g_1, g_2\}$—suppose that agent 1 is allocated g_1 and agent 2 is allocated g_2. In the third round, without loss of generality, suppose that g_3 is allocated to agent 1. Then, agent 2 will still envy agent 1 even after dropping g_1 from agent 1's bundle, as $v(A_2) = v(g_2) = 1 < v(A_1 \setminus \{g_1\}) = v(g_3) = 2$.

Next, we prove the result for chores. Consider the instance with two agents $N = \{1, 2\}$ and three chores $O = \{c_1, c_2, c_3\}$, where agents have identical valuations: $v(c_1) = v(c_2) = -1$ and $v(c_3) = -2$. In order for the partial allocation at the end of the secound round to be TEFX, each agent must be allocated exactly one of $\{c_1, c_2\}$—suppose that agent 1 is allocated c_1 and agent 2 is allocated c_2. In the third round, without loss of generality, suppose that c_3 is allocated to agent 1. Then, agent 1 will still envy agent 2 even after dropping c_1 from her own bundle, as $v(A_1 \setminus \{c_1\}) = v(c_3) = -2 < v(A_2) = v(c_2) = -1$. □

4.3.2 Other Restricted Settings

The next natural question we ask is whether there are other special cases where an TEF1 allocation is guaranteed to exist. We answer this question affirmatively by demonstrating the existence of TEF1 allocations in three special cases, each supported by a polynomial-time algorithm that returns such an allocation.

4.3.2.1 Two Types of Items

The first setting we consider is one where items can be divided into two *types*, and each agent values all items of a particular type equally. Formally, let $S_1, S_2 \subseteq O$ be a partition of the set of items, so that $S_1 \cap S_2 = \varnothing$, and $S_1 \cup S_2 = O$. Then, for any $r \in \{1, 2\}$, two items $o, o' \in S_r$, and agent $i \in N$, we have that $v_i(o) = v_i(o')$.

Settings with only two types of items/tasks arise naturally in various applications, such as distributing food and clothing donations from a charity, or allocating cleaning and cooking chores in a household.

This preference restriction has been studied for chores in offline settings [Aziz et al., 2023b, Garg et al., 2024], and we remark that agents may have distinct valuations for up to $2n$ different items, unlike the extensively studied *bi-valued* preferences [Ebadian et al., 2022, Garg et al., 2022] which involve only two distinct item values.

We show that for this setting, a TEF1 allocation for goods or chores always exists and can be computed in polynomial time. Intuitively, the algorithm treats the two item types independently: items of the first type are allocated in a round-robin manner from agent 1 to n, while items of the second type are allocated in reverse round-robin order from agent n to 1. Then, our result is as follows.

Theorem 4.3.6. *When there are two types of items, a* TEF1 *allocation for goods or chores exists and can be computed in polynomial time.*

Proof. Consider the following greedy algorithm (Algorithm 5).

Algorithm 5: Returns a TEF1 allocation for goods or chores when there are two types of items

---

Input: Set of agents $N = \{1, \ldots, n\}$, set of items $O = \{o_1, \ldots, o_m\}$, and valuation profile $\mathbf{v} = (v_1, \ldots, v_n)$
Output: TEF1 allocation $\mathcal{A}$ of items in O to agents in N

Initialize $\alpha \leftarrow 1$, $\beta \leftarrow n$, and $\mathcal{A}^0 \leftarrow (\varnothing, \ldots, \varnothing)$
for $t = 1, 2, \ldots, m$ **do**
  if $\alpha = n + 1$ **then**
    $\alpha \leftarrow 1$
  else if $\beta = 0$ **then**
    $\beta \leftarrow n$
  end if
  if $o_t \in T_1$ **then**
    $A_\alpha^t \leftarrow A_\alpha^{t-1} \cup \{o_t\}$, $A_j^t \leftarrow A_j^{t-1}$ for all $j \in N \setminus \{\alpha\}$, and $\alpha \leftarrow \alpha + 1$
  else
    $A_\beta^t \leftarrow A_\beta^{t-1} \cup \{o_t\}$, $A_j^t \leftarrow A_j^{t-1}$ for all $j \in N \setminus \{\beta\}$, and $\beta \leftarrow \beta - 1$
  end if
end for
return $\mathcal{A} = (A_1^m, \ldots, A_n^m)$

The polynomial runtime of the Algorithm 5 is easy to verify: there is only one **for** loop which runs in $\mathcal{O}(m)$ time, and the other operations within run in $\mathcal{O}(mn)$ time. Thus, we focus on proving correctness.

Intuitively, α and β each keep a counter of which agent should be next allocated an item of type T_1 and T_2, respectively. For this reason, for each $r \in \{1, 2\}$, we can observe that with respect to only items of type T_r, the algorithm allocates these items in a round-robin fashion. We can therefore make the following two observations:

(i) for any pair of agents $i, j \in N$, if $|A_i^t \cap T_1| > |A_j^t \cap T_1|$, then $i < j$; if $|A_i^t \cap T_2| > |A_j^t \cap T_2|$, then $i > j$; and

(ii) for any pair of agents $i, j \in N$, round $t \in [m]$, and $r \in \{1, 2\}$, we have that $\big||A_i^t \cap T_r| - |A_j^t \cap T_r|\big| \leq 1$.

The first observation follows from fact that the α counter is increasing in agent indices whereas the β counter is decreasing in agent indices. The second observation follows from the widely-known fact that, with respect to items of a specific type, a round-robin allocation always returns a balanced allocation, i.e., the bundle sizes of any two agents differ by no more than one.

Next, we have that for any two agents $i, j \in N$, round $t \in [m]$, and $r, r' \in \{1, 2\}$ where $r \neq r'$, if $|A_i^t \cap T_r| > |A_j^t \cap T_r|$, then $|A_i^t \cap T_{r'}| \leq |A_j^t \cap T_{r'}|$. To see this, suppose for a contradiction that there exists agents $i, j \in N$ and round $t \in [m]$ such that both $|A_i^t \cap T_1| > |A_j^t \cap T_1|$ and $|A_i^t \cap T_2| > |A_j^t \cap T_2|$. Then, observation (i) will give us $i < j$ and $i > j$ respectively, a contradiction.

For the case of goods, we have that for any pair of agents $i, j \in N$ and round $t \in [m]$, if $i < j$, then

$$v_i(A_i^t \cap T_1) \geq v_i(A_j^t \cap T_1) \tag{4.1}$$

because i precedes j in the round-robin allocation order, and by the well-established EF1 property of the round-robin algorithm for goods, there exists a good $g \in A_j^t \cap T_2$ such that

$$v_i(A_i^t \cap T_2) \geq v_i(A_j^t \cap T_2 \setminus \{g\}). \tag{4.2}$$

Combining (4.1) and (4.2), there exists a good $g \in A_j^t$ such that

$$v_i(A_i^t) = v_i(A_i^t \cap T_1) + v_i(A_i^t \cap T_2) \geq v_i(A_j^t \cap T_1) + v_i(A_j^t \cap T_2 \setminus \{g\}) = v_i(A_j^t \setminus \{g\}).$$

Moreover, if $i > j$, then

$$v_i(A_i^t \cap T_2) \geq v_i(A_i^t \cap T_2), \tag{4.3}$$

and there exists a good $g \in A_j^t \cap T_1$ such that

$$v_i(A_i^t \cap T_1) \geq v_i(A_j^t \cap T_1 \setminus \{g\}). \tag{4.4}$$

Combining (4.3) and (4.4), there exists a good $g \in A_j^t$ such that

$$v_i(A_i^t) = v_i(A_i^t \cap T_1) + v_i(A_i^t \cap T_2) \geq v_i(A_j^t \cap T_1 \setminus \{g\}) + v_i(A_j^t \cap T_2) = v_i(A_j^t \setminus \{g\}).$$

For the case of chores, we have that for any pair of agents $i, j \in N$ and round $t \in [m]$, if $i > j$, then

$$v_i(A_i^t \cap T_1) \geq v_i(A_j^t \cap T_1), \tag{4.5}$$

and again by the EF1 property of the round-robin algorithm for chores Aziz et al. [2019], there exists a chore $c \in A_i^t \cap T_2$ such that

$$v_i(A_i^t \cap T_2 \setminus \{c\}) \geq v_i(A_j^t \cap T_2). \tag{4.6}$$

Combining (4.5) and (4.6), there exists a chore $c \in A_i^t$ such that

$$v_i(A^t \setminus \{c\}) = v_i(A_i^t \cap T_1) + v_i(A_i^t \cap T_2 \setminus \{c\}) \geq v_i(A_j^t \cap T_1) + v_i(A_j^t \cap T_2) = v_i(A_j^t).$$

Moreover, if $i < j$, then

$$v_i(A_i^t \cap T_2) \geq v_i(A_j^t \cap T_2), \tag{4.7}$$

and there exists a chore $c \in A_i^t \cap T_1$ such that

$$v_i(A_i^t \cap T_1 \setminus \{c\}) \geq v_i(A_j^t \cap T_1). \tag{4.8}$$

Combining (4.7) and (4.8), there exists a chore $c \in A_i^t$ such that

$$v_i(A^t \setminus \{c\}) = v_i(A_i^t \cap T_1 \setminus \{c\}) + v_i(A_i^t \cap T_2) \geq v_i(A_j^t \cap T_1) + v_i(A_j^t \cap T_2) = v_i(A_j^t).$$

Thus, our result holds. □

4.3.2.2 Generalized Binary Valuations

The next setting we consider is one where agents have *generalized binary valuations* (also known as *restricted additive valuations* [Akrami et al., 2022, Camacho et al., 2023]). This class of valuation functions generalizes both identical and binary valuations, which are both widely studied in fair division [Halpern et al., 2020, Plaut and Roughgarden, 2020, Suksompong and Teh, 2022].

Formally, we say that agents have *generalized binary valuations* if for every agent $i \in N$ and item $o_j \in O$, $v_i(o_j) \in \{0, p_j\}$, where $p_j \in \mathbb{R} \setminus \{0\}$.

We show that for this setting, a TEF1 allocation can be computed efficiently, with the following result. We remark that the resulting allocation also satisfies *Pareto-optimality* (Definition 2.1.5).

Theorem 4.3.7. *When agents have generalized binary valuations, a* TEF1 *allocation for goods or chores exists and can be computed in polynomial time.*

Proof. We first prove the result for goods. Consider the following greedy algorithm (Algorithm 6) which iterates through the rounds and allocates each good to the agent who has the least value for their bundle.

Algorithm 6: Returns a TEF1 allocation of goods under generalized binary valuations

Input: Set of agents $N = \{1, \ldots, n\}$, set of goods $O = \{g_1, \ldots, g_m\}$, and valuation profile $\mathbf{v} = (v_1, \ldots, v_n)$
Output: TEF1 allocation of goods $\mathcal{A}$ in O to agents in N

```
Initialize the empty allocation 𝒜^0 where A_i^0 = ∅ for all i ∈ N.
for t = 1, 2, ..., m do
    Let S := {i' ∈ N | v_{i'}(g_t) > 0}
    if S = ∅ then
        Let i be any agent in N
    else
        Let i ∈ arg min_{i'∈S} v_{i'}(A_{i'}^{t-1}), with ties broken arbitrarily
    end if
    A_i^t ← A_i^{t-1} ∪ {g_t} and A_j^t ← A_j^{t-1} for all j ∈ N \ {i}
end for
return 𝒜 = (A_1^m, ..., A_n^m)
```

We first show that for any $i, j \in N$ and $t \in [m]$, it holds that $v_i(A_i^t) \geq v_j(A_i^t)$. Suppose for a contradiction that there exists some $i, j \in N$ and $t \in [m]$ such that $v_i(A_i^t) < v_j(A_i^t)$. This means there exists some good $g \in A_i^t$ whereby $v_i(g) = 0$ and $v_j(g) > 0$. However, then the algorithm would not have allocated g to i, a contradiction.

Next, we will prove by induction that for every $t \in [m]$, $\mathcal{A}^t$ is TEF1. The base case is trivially true: when $t = 1$, if every agent values g_1 at 0, then allocating it to any agent will satisfy TEF1, whereas if some agent values g_1, allocating it to any agent will also be TEF1: the envy by any other agent towards this agent will disappear with the removal of g_1 from the agent's bundle (every agent's bundle will then be the empty set).

Then, we prove the inductive step. Assume that for some $k \in [m-1]$, $\mathcal{A}^k$ is TEF1. We will show that $\mathcal{A}^{k+1}$ is also TEF1. Due to the assumption, it suffices to show that for all $i, j \in N$, there exists a good $g \in A_j^{k+1}$ such that $v_i(A_i^{k+1}) \geq v_i(A_j^{k+1} \setminus \{g\})$. Consider the agent $i \in N$ that is allocated g_{k+1}.

We first show agent i must be unenvied before being allocated g_{k+1}. Suppose towards a contradiction this is not the case, i.e., there exists some other agent $j \neq i$ whereby $v_j(A_j^k) < v_j(A_i^k)$. Together with the fact that $v_j(A_i^k) \leq v_i(A_i^k)$ from the result above, we get that

$$v_j(A_j^k) < v_j(A_i^k) \leq v_i(A_i^k),$$

contradicting the fact that i is an agent with the minimum bundle value and thus chosen by the algorithm to receive g_{k+1}. As such, i must be unenvied before being allocated g_{k+1}, i.e., for any other agent $j \in N \setminus \{i\}$, we have that $v_j(A_j^k) \geq v_j(A_i^k)$.

Consequently, we get that

$$v_j(A_j^{k+1}) = v_j(A_j^k) \geq v_j(A_i^k) = v_j(A_i^{k+1} \setminus \{g_{k+1}\}).$$

Thus, by induction, the result holds.

Next, we prove the result for chores. Consider the following greedy algorithm (Algorithm 7) which iterates through the rounds, allocating each chore to an agent with zero value for it if possible, and otherwise, allocates the chore to an agent who does not envy any other agent.

Algorithm 7: Returns an TEF1 allocation of chores under generalized binary valuations

Input: Set of agents $N = \{1, \ldots, n\}$, set of chores $O = \{c_1, \ldots, c_m\}$, and valuation profile $\mathbf{v} = (v_1, \ldots, v_n)$
Output: TEF1 allocation of chores $\mathcal{A}$ in O to agents in N

Initialize the empty allocation $\mathcal{A}^0$ where $A_i^0 = \varnothing$ for all $i \in N$.
for $t = 1, 2, \ldots, m$ **do**
  if there exists an agent $i \in N$ such that $v_i(c_t) = 0$ **then**
    Let $i \in \{i' \in N \mid v_{i'}(c_t) = 0\}$
  else
    Let $i \in \arg\max_{i' \in N} v_{i'}(A_{i'}^{t-1})$
  end if
  $A_i^t \leftarrow A_i^{t-1} \cup \{c_t\}$ and $A_j^t \leftarrow A_j^{t-1}$ for all $j \in N \setminus \{i\}$
end for
return $\mathcal{A} = (A_1^m, \ldots, A_n^m)$

We first show that for any $i, j \in N$ and $t \in [m]$, it holds that

$$v_i(A_i^t) \geq v_j(A_i^t). \tag{4.9}$$

Suppose for a contradiction that there exists some $i, j \in N$ and $t \in [m]$ such that $v_i(A_i^t) < v_j(A_i^t)$. This means there exists some chore $c \in A_i^t$ whereby $v_i(c) < 0$ and $v_j(c) = 0$. However, then the algorithm would not have allocated c to i, a contradiction.

Next, we will prove by induction that for every $t \in [m]$, $\mathcal{A}^t$ is TEF1. The base case is trivially true: when $t = 1$, if there exists an agent that values c_1 at 0, then allocating it to any such agent will satisfy TEF1, whereas if all agents values c_1

negatively, allocating it to any agent will also be TEF1: the envy by this agent towards any other agent will disappear with the removal of c_1 from the former agent's bundle (every agent's bundle will then be the empty set).

Then, we prove the inductive step. Assume that for some $k \in [m-1]$, $\mathcal{A}^k$ is TEF1. We will show that $\mathcal{A}^{k+1}$ is also TEF1, i.e., for all $i, j \in N$, there exists a chore $c \in A_i^{k+1}$ such that $v_i(A_i^{k+1} \setminus \{c\}) \geq v_i(A_j^{k+1})$.

Suppose agent i is allocated the chore c_{k+1}. If $v_i(c_{k+1}) = 0$, then each agents' valuation for every other agent's bundle (including his own) remains the same, and thus $\mathcal{A}^{k+1}$ remains TEF1. If $v_i(c_{k+1}) < 0$, then we know that $v_j(c_{k+1}) < 0$ for all $j \in N$. We then proceed to show that agent i must not envy any other agent before being allocated c_{k+1}. Suppose for contradiction this is not the case, i.e., that there exists some other agent $j \neq i$ whereby $v_i(A_i^k) < v_i(A_j^k)$. Since c_{k+1} is allocated to the agent with the highest bundle, we have that $v_i(A_i^k) \geq v_j(A_j^k)$, and therefore

$$v_i(A_j^k) > v_i(A_i^k) \geq v_j(A_j^k).$$

However, this contradicts (4.9).

Since agent i does not envy another agent before being allocated c_{k+1}, we get that for any $j \neq i$,

$$v_i(A_i^{k+1} \setminus \{c_{k+1}\}) = v_i(A_i^k) \geq v_i(A_j^k) = v_i(A_j^{k+1}) \quad \text{and} \quad v_j(A_j^{k+1}) = v_j(A_j^k).$$

Thus, by induction, we get that $\mathcal{A}^{t+1}$ is TEF1. $\square$

4.3.2.3 Unimodal Preferences

The last setting that we consider is the class of *unimodal preferences*, which consists of the widely studied *single-peaked* and *single-dipped* preference structures in social choice [Black, 1948, Arrow, 2012] and cake cutting [Thomson, 1994, Bhardwaj et al., 2020]. We adapt these concepts for the online fair division setting with a single item at each timestep.

Definition 4.3.8. A valuation profile $\mathbf{v}$ is *single-peaked* if for each agent $i \in N$, there is an item o_{i^*} where for each $j, k \in [m]$ such that $j < k < i^*$, $v_i(o_j) \leq v_i(o_k) \leq v_i(o_{i^*})$, and for each $j, k \in [m]$ such that $i^* < j < k$, $v_i(o_{i^*}) \geq v_i(o_j) \geq v_i(o_k)$.

Definition 4.3.9. A valuation profile $\mathbf{v}$ is *single-dipped* if for each agent $i \in N$, there is an item o_{i^*} where for each $j, k \in [m]$ such that $j < k < i^*$, $v_i(o_j) \geq v_i(o_k) \geq v_i(o_{i^*})$, and for each $j, k \in [m]$ such that $i^* < j < k$, $v_i(o_{i^*}) \leq v_i(o_j) \leq v_i(o_k)$.

In other words, under single-peaked (resp. single-dipped) valuations, agents have a specific item o_{i^*} that they prefer (resp. dislike) the most, and prefer (resp. dislike) items less as they arrive further away in time from o_{i^*}.

Note that this restricted preference structure is well-defined for the setting of a single item arriving per round, but may not be compatible with a generalization to multiple items per round as described in Lemma 4.3.1 (unless the items in each round are identically-valued by agents).[2]

Unimodal preferences may arise in settings where agents place higher value on resources at the time surrounding specific events. For example, in disaster relief, the demand for food and essential supplies peaks as a natural disaster approaches, then declines once the immediate crisis passes. Similarly, in project management, the workload for team members intensifies (in terms of required time and effort) as the project nears its deadline, but significantly decreases during the final stages, such as editing and proofreading.

Unimodal preferences also generalizes other standard preference restrictions studied in fair division and voting models, such as settings where agents have *monotonic valuations* [Elkind et al., 2025c] or *identical rankings* [Plaut and Roughgarden, 2020].

We propose efficient algorithms for computing a TEF1 allocation for goods when agents have single-peaked valuations, and for chores when agents have single-dipped valuations.

Theorem 4.3.10. *When agents have single-peaked valuations, a* TEF1 *allocation for goods exists and can be computed in polynomial time. When agents have single-dipped valuations, a* TEF1 *allocation for chores exists and can be computed in polynomial time.*

Proof. Consider the following greedy algorithm (Algorithm 8). Note that the same algorithm works for both settings for goods when valuations are single-peaked, and for chores when valuations are single-dipped.

The polynomial runtime of the Algorithm 8 is easy to verify: there is only one **for** loop which runs in $\mathcal{O}(m)$ time, and the other operations within run in $\mathcal{O}(n)$ time. Thus, we focus on proving correctness.

We first prove the case for goods, when valuations are single-peaked.

For each $i \in [m]$, let $g_i = o_i$, and thus $O = \{g_1, \dots, g_m\}$. We can assume that $m = \alpha n$ for some $\alpha \in \mathbb{Z}_{>0}$; otherwise we can simply add dummy goods to O until

[2]Specifically, in the multiple items per round case, if the bundles of items at each timestep are unimodally valued, the single-item per round transformation of the instance may not necessarily be unimodal.

Algorithm 8: Returns a TEF1 allocation for goods when valuations are single-peaked and chores when valuations are single-dipped

Input: Set of agents $N = \{1, \ldots, n\}$, set of items $O = \{o_1, \ldots, o_m\}$, and valuation profile $\mathbf{v} = (v_1, \ldots, v_n)$

Output: TEF1 allocation $\mathcal{A}$ of items in O to agents in N

```
Initialize A^0 ← (∅, ..., ∅)
for t = 1, 2, ..., m do
    Let i := arg min_{i∈N} |A_i^{t-1}|, with ties broken lexicographically
    A_i^t ← A_i^{t-1} ∪ {g_t} and A_i^t ← A_i^{t-1}
end for
return A = (A_1^m, ..., A_n^m)
```

that condition is fulfilled. Then, Algorithm 8 will return $\mathcal{A}$, where for each $i \in N$, $A_i = \{g_i, g_{i+n}, \ldots, g_{i+(\alpha-1)n}\}$.

For each $i \in N$ and $j \in [\alpha]$, let

- $T_j := \{g_{(j-1)n+1}, g_{(j-1)n+2}, \ldots, g_{jn}\}$,
- $g'_{i,j} \in A_i \cap T_j$ be the unique good in T_j that was allocated to agent i
- $g^* := \arg\max_{g \in O} v_i(g)$ (with ties broken arbitrarily), and $g^* \in T_{i^*}$ for some $i^* \in [\alpha]$.

Then, we will show that for all $r \in [\alpha]$, $v_i(A_i^r) \geq v_i(A_j^r \setminus \{g\})$ for some $g \in A_j^r$. We split our analysis into two cases.

Case 1: $i < j$. If $r < i^*$, then since agent i's valuation for each subsequent good up to round T_r is non-decreasing, we have that for all $k \in \{2, \ldots, r\}$,

$$v_i(g'_{i,k}) \geq v_i(g'_{j,k-1}).$$

Consequently, we get that

$$v_i(A_i^r) \geq \sum_{k=2}^{r} v_i(g'_{i,k}) \geq \sum_{k=2}^{r} v_i(g'_{j,k-1}) = v_i(A_j^r \setminus \{g'_{j,r}\}).$$

If $r \geq i^*$, then we split our analysis into two further cases.

Case 1(a): g'_{i,i^*} appears before g^*. Then, for all $k \in \{2, \ldots, i^*\}$,

$$v_i(g'_{i,k}) \geq v_i(g'_{j,k-1}),$$

and for all $k \in \{i^* + 1, \ldots, r\}$,

$$v_i(g'_{i,k}) \geq v_i(g'_{j,k}).$$

Consequently, we get that

$$\begin{aligned} v_i(A_i^r) &\geq \sum_{k=2}^{i^*} v_i(g'_{i,k}) + \sum_{k=i^*+1}^{r} v_i(g'_{i,k}) \\ &\geq \sum_{k=2}^{i^*} v_i(g'_{j,k-1}) + \sum_{k=i^*+1}^{r} v_i(g'_{j,k}) = v_i(A_j^r \setminus \{g'_{j,i^*}\}). \end{aligned}$$

Case 1(b): g'_{i,i^*} appears after (or is) g^*. Then, for all $k \in \{2, \ldots, i^* - 1\}$,

$$v_i(g'_{i,k}) \geq v_i(g'_{j,k-1}),$$

and for all $k \in \{i^*, \ldots, r\}$,

$$v_i(g'_{i,k}) \geq v_i(g'_{j,k}).$$

Consequently, we get that

$$\begin{aligned} v_i(A_i^r) &\geq \sum_{k=2}^{i^*-1} v_i(g'_{i,k}) + \sum_{k=i^*}^{r} v_i(g'_{i,k}) \\ &\geq \sum_{k=2}^{i^*-1} v_i(g'_{j,k-1}) + \sum_{k=i^*}^{r} v_i(g'_{j,k}) = v_i(A_j^r \setminus \{g'_{j,i^*-1}\}). \end{aligned}$$

Case 2: $i > j$. If $r \leq i^*$, then since agent i's valuation for each subsequent good up to round T_r is nondecreasing, we have that for all $k \in [r]$,

$$v_i(g'_{i,k}) \geq v_i(g'_{j,k}). \tag{4.10}$$

Consequently, we get that

$$v_i(A_i^r) \geq \sum_{k \in [r-1]} v_i(g'_{i,k}) \geq \sum_{k \in [r-1]} v_i(g'_{j,k}) \quad \text{(by (4.10))} = v_i(A_j^r \setminus \{g'_{j,r}\}).$$

If $r > i^*$, then we split our analysis into two further cases.

Case 2(a): g'_{i,i^*} appears before (or is) g^*. Then for all $k \in [i^*]$,

$$v_i(g'_{i,k}) \geq v_i(g'_{j,k}),$$

and for all $k \in \{i^* + 1, \ldots, r-1\}$,

$$v_i(g'_{i,k}) \geq v_i(g'_{j,k+1}).$$

Consequently, we get that

$$\begin{aligned} v_i(A_i^r) &\geq \sum_{k \in [i^*]} v_i(g'_{i,k}) + \sum_{k=i^*+1}^{r-1} v_i(g'_{i,k}) \\ &\geq \sum_{k \in [i^*]} v_i(g'_{j,k}) + \sum_{k=i^*+1}^{r-1} v_i(g'_{j,k+1}) = v_i(A_j^r \setminus \{g'_{j,i^*+1}\}). \end{aligned}$$

Case 2(b): g'_{i,i^*} appears after g^*. Then, for all $k \in [i^* - 1]$,

$$v_i(g'_{i,k}) \geq v_i(g'_{j,k}),$$

and for all $k \in \{i^*, \ldots, r-1\}$,

$$v_i(g'_{i,k}) \geq v_i(g'_{j,k+1}).$$

Consequently, we get that

$$\begin{aligned} v_i(A_i^r) &\geq \sum_{k \in [i^*-1]} v_i(g'_{i,k}) + \sum_{k=i^*}^{r-1} v_i(g'_{i,k}) \\ &\geq \sum_{k \in [i^*-1]} v_i(g'_{j,k}) + \sum_{k=i^*}^{r-1} v_i(g'_{j,k+1}) = v_i(A_j^r \setminus \{g'_{j,i^*}\}). \end{aligned}$$

Thus, our result follows.

Next, we prove the case for chores, when valuations are single-dipped.

For each $j \in [m]$, let $o_i = c_i$, and thus $O = \{c_1, \ldots, c_m\}$. We can assume that $m = \alpha n$ for some $\alpha \in \mathbb{Z}_{>0}$; otherwise we can simply add dummy chores to O until that condition is fulfilled. Then, Algorithm 8 will return $\mathcal{A}$, where for each $i \in N$, $A_i = \{c_i, c_{i+n}, \ldots, c_{i+(\alpha-1)n}\}$.

For each $i \in N$ and $j \in [\alpha]$, let

- $T_j := \{c_{(j-1)n+1}, c_{(j-1)n+2}, \ldots, c_{jn}\}$,
- $c'_{i,j} \in A_i \cap T_j$ be the unique chore in T_j that was allocated to agent i
- $c^* := \arg\min_{c \in O} v_i(c)$ (with ties broken arbitrarily), and $c^* \in T_{i^*}$ for some $i^* \in [\alpha]$.

Case 1: $i < j$. If $r \leq j^*$, then since agent i's valuation for each subsequent chore up to round T_{r-1} is nonincreasing, we have that for all $k \in [r-1]$,

$$v_i(c'_{i,k}) \geq v_i(c'_{j,k}).$$

Consequently, we get that

$$v_i(A_i^r \setminus \{c'_{i,r}\}) = \sum_{k \in [r-1]} v_i(c'_{i,k}) \geq \sum_{k \in [r-1]} v_i(c'_{j,k}) \geq v_i(A_j^r).$$

If $r > j^*$, then we split our analysis into two further cases.

Case 1(a): c'_{j,j^*} appears before (or is) c^*. Then for all $k \in [j^*]$,

$$v_i(c'_{i,k}) \geq v_i(c'_{j,k})$$

and for all $k \in \{j^* + 2, \ldots, r\}$,

$$v_i(c'_{i,k}) \geq v_i(c'_{j,k-1}).$$

Consequently, we get that

$$\begin{aligned} v_i(A_i^r \setminus \{c'_{i,j^*+1}\}) &= \sum_{k \in [j^*]} v_i(c'_{i,k}) + \sum_{k=j^*+2}^{r} v_i(c'_{i,k}) \\ &\geq \sum_{k \in [j^*]} v_i(c'_{j,k}) + \sum_{k=j^*+2}^{r} v_i(c'_{j,k-1}) \geq v_i(A_j^r). \end{aligned}$$

Case 1(b): c'_{j,j^*} appears after c^*. Then for all $k \in [j^* - 1]$,

$$v_i(c'_{i,k}) \geq v_i(c'_{j,k})$$

and for all $k \in \{j^* + 1, \ldots, r\}$,

$$v_i(c'_{i,k}) \geq v_i(c'_{j,k-1}).$$

Consequently, we get that

$$\begin{aligned} v_i(A_i^r \setminus \{c'_{i,j^*}\}) &= \sum_{k \in [1, j^*-1]} v_i(c'_{i,k}) + \sum_{k=j^*+1}^{r} v_i(c'_{i,k}) \\ &\geq \sum_{k \in [j^*-1]} v_i(c'_{j,k}) + \sum_{k=j^*+1}^{r} v_i(c'_{j,k-1}) \geq v_i(A_j^r). \end{aligned}$$

Case 2: $j < i$**.** If $r < j^*$, then since agent i's valuation for each subsequent chore up to round T_r is nondecreasing, we have that for all $k \in [r-1]$,

$$v_i(c'_{i,k}) \geq v_i(c'_{j,k+1}).$$

Consequently, we get that

$$v_i(A_i^r \setminus \{c'_{i,r}\}) = \sum_{k \in [r-1]} v_i(c'_{i,k}) \geq \sum_{k \in [r-1]} v_i(c'_{j,k+1}) \geq v_i(A_j^r).$$

If $r \geq j^*$, then we split our analysis into two further cases.

Case 2(a): c_{j,j^*} **appears before (or is)** c^***.** Then for all $k \in [j^* - 1]$,

$$v_i(c'_{i,k}) \geq v_i(c'_{j,k+1})$$

and for all $k \in \{j^* + 1, \ldots, r\}$,

$$v_i(c'_{i,k}) \geq v_i(c'_{j,k}).$$

Consequently, we get

$$\begin{aligned} v_i(A_i^r \setminus \{c'_{i,j^*}\}) &= \sum_{k \in [j^*-1]} v_i(c'_{i,k}) + \sum_{k=j^*+1}^{r} v_i(c'_{i,k}) \\ &\geq \sum_{k \in [j^*-1]} v_i(c'_{j,k+1}) + \sum_{k=j^*+1}^{r} v_i(c'_{k,j}) \geq v_i(A_j^r). \end{aligned}$$

Case 2(b): c_{j,j^*} **appears after** c^***.** Then for all $k \in [j^* - 2]$,

$$v_i(c'_{i,k}) \geq v_i(c'_{j,k+1}$$

and for all $k \in \{j^*, \ldots, r\}$,

$$v_i(c'_{i,k}) \geq v_i(c'_{j,k}).$$

Consequently, we get that

$$\begin{aligned} v_i(A_i^r \setminus \{c'_{i,j^*-1}\}) &= \sum_{k \in [j^*-2]} v_i(c'_{i,k}) + \sum_{k=j^*}^{r} v_i(c'_{i,k}) \\ &\geq \sum_{k \in [j^*-2]} v_i(c'_{j,k+1}) + \sum_{k=j^*}^{r} v_i(c'_{j,k}) \geq v_i(A_j^r). \end{aligned}$$

Thus, our result follows. $\square$

We note that while a simple greedy algorithm performs well in the case of single-peaked valuations for goods and single-dipped valuations for chores, it fails in the reverse scenario—single-dipped valuations for goods and single-peaked valuations for chores. This is due to the fact that, in the latter case, the position of the dip or peak becomes critical and significantly complicates the way we allocate the item. We leave the existence of polynomial-time algorithm(s) for the reverse scenario as an open question.

4.3.3 Hardness Results for TEF1 Allocations

The non-existence of TEF1 goods allocations for $n \geq 3$ prompts us to explore whether we can determine if a given instance admits a TEF1 allocation for goods. Unfortunately, we show that this problem is NP-hard, with the following result.

Theorem 4.3.11. *Given an instance of the temporal fair division problem with goods and $n \geq 3$, determining whether there exists a* TEF1 *allocation is* NP-*hard.*

Proof. We reduce from the 1-IN-3-SAT problem which is NP-hard. An instance of this problem consists of a conjunctive normal form formula F with three literals per clause; it is a yes instance if there exists a truth assignment to the variables such that each clause has exactly one `True` literal, and a no instance otherwise.

Consider an instance of 1-IN-3-SAT given by the CNF F which contains n variables $\{x_1, \dots, x_n\}$ and m clauses $\{C_1, \dots, C_m\}$. We construct an instance $\mathcal{I}$ with three agents and $2n + 2$ goods. For each $i \in [n]$, we introduce two goods t_i, f_i. We also introduce two additional goods s and r. Let the agents' (identical) valuations be defined as follows:

$$v(g) = \begin{cases} 5^{m+n-i} + \sum_{j\,:\,x_i \in C_j} 5^{m-j}, & \text{if } g = t_i, \\ 5^{m+n-i} + \sum_{j\,:\,\neg x_i \in C_j} 5^{m-j}, & \text{if } g = f_i, \\ \sum_{j \in [m]} 5^{j-1}, & \text{if } g = r, \\ \sum_{i \in [n]} 5^{m+i-1} + 2 \times \sum_{j \in [m]} 5^{j-1}, & \text{if } g = s. \end{cases}$$

Intuitively, for each variable index $i \in [n]$, we associate with it a unique value 5^{m+n-i}. For each clause index $j \in [m]$, we also associate with it a unique value 5^{m-j}. Note that no two indices (regardless of whether its a variable or clause index) share the same value. Then, the value of each good t_i comprises of the unique value associated with i, and the sum over all unique values of clauses C_j which x_i appears as a *positive literal* in; whereas the value of each good f_i comprises of the unique value associated with i, and the sum over all unique values of clauses C_j which x_i appears as a *negative literal* in. We will utilize this in our analysis later.

Then, we have the set of goods $O = \{s, t_1, f_1, t_2, f_2, \ldots, t_n, f_n, r\}$. Note that $v(O) = v(s) + v(r) + \sum_{i\in[n]} v(t_i) + \sum_{i\in[n]} v(f_i)$. Also observe that $\sum_{i\in[n]} 5^{m+n-i} = \sum_{i\in[n]} 5^{m+i-1}$. Now, as each clause contains exactly three literals, we have

$$\sum_{i\in[n]} \sum_{j:x_i\in C_j} 5^{m-j} + \sum_{i\in[n]} \sum_{j:\neg x_i\in C_j} 5^{m-j} = 3 \times \sum_{j\in[m]} 5^{j-1}.$$

Then, combining the equations above, we get that

$$v(O) = 3 \times \sum_{i\in[n]} 5^{m+i-1} + 6 \times \sum_{j\in[m]} 5^{j-1}. \tag{4.11}$$

Let the goods be in the following order:

$$s, t_1, f_1, t_2, f_2, \ldots, t_n, f_n, r.$$

We first prove the following result.

Lemma 4.3.12. *There exists a truth assignment α such that each clause in F has exactly one* `True` *literal if and only if there exists an allocation $\mathcal{A}$ such that $v(A_1) = v(A_2) = v(A_3)$ for instance $\mathcal{I}$.*

Proof. For the 'if' direction, consider an allocation $\mathcal{A}$ such that $v(A_1) = v(A_2) = v(A_3)$. Then, we have that $O = A_1 \cup A_2 \cup A_3$ and $v(A_1) = v(A_2) = v(A_3) = \frac{1}{3}v(O)$. Since agents have identical valuations, without loss of generality, let $s \in A_1$. Then, since $v(A_1^1) = v(s) = \frac{1}{3}v(O)$, agent 1 should not receive any more goods after s, and each remaining good should go to agent 2 or 3.

Again, without loss of generality, we let $r \in A_2$. Then since $v(A_2) = \frac{1}{3}v(O)$, we have that

$$\begin{aligned} v(A_2 \setminus \{r\}) &= \left(\sum_{i\in[n]} 5^{m+i-1} + 2 \times \sum_{j\in[m]} 5^{j-1} \right) - \sum_{j\in[m]} 5^{j-1} \\ &= \sum_{i\in[n]} 5^{m+i-1} + \sum_{j\in[m]} 5^{j-1}. \end{aligned}$$

Note that this is only possible if for each $i \in [m]$, t_i and f_i are allocated to different agents. The reason is because the only way agent 1 can obtain the first term of the above bundle value (less good r) is if he is allocated exactly one good from each of $\{t_i, f_i\}$ for all $i \in [n]$.

Then, from the goods that exist in bundle A_2, we can construct an assignment α: for each $i \in [n]$, let $x_i = \texttt{True}$ if $t_i \in A_2$ and $x_i = \texttt{False}$ if $f_i \in A_2$. Then, from the second term in the expression of $v(A_1 \setminus \{r\})$ above, we can observe that each clause has exactly one `True` literal (because the sum is only obtainable if exactly one literal appears in each clause, and our assignment will set each of these literals to `True`).

For the 'only if' direction, consider a truth assignment α such that each clause in F has exactly one `True` literal. Then, for each $i \in [n]$, let

$$\ell_i = \begin{cases} t_i & \text{if } x_i = \texttt{True} \text{ under } \alpha, \\ f_i & \text{if } x_i = \texttt{False} \text{ under } \alpha. \end{cases}$$

We construct the allocation $\mathcal{A} = (A_1, A_2, A_3)$ where

$$A_1 = \{s\}, \quad A_2 = \{\ell_1, \ldots, \ell_n, r\}, \quad \text{and} \quad A_3 = O \setminus (A_1 \cup A_2).$$

Again, observe that $\sum_{i \in [n]} 5^{m+n-i} = \sum_{i \in [n]} 5^{m+i-1}$. Also note that $v(A_1) = \frac{1}{3}v(O)$. Then, as each clause has exactly one `True` literal, $v(A_2) = \sum_{i \in [n]} 5^{m+i-1} + 2 \times \sum_{j \in [m]} 5^{j-1}$, and together with (4.11), we get that $v(A_3) = \frac{2}{3}v(O) - v(A_1) = v(A_1)$ and hence $v(A_1) = v(A_2) = v(A_3)$, as desired. $\square$

Now consider another instance $\mathcal{I}'$ that is similar to $\mathcal{I}$, but with an additional 21 goods $\{g_1, \ldots, g_{21}\}$. Let agents' valuations over these new goods be defined as follows:

v	g_1	g_2	g_3	g_4	g_5	g_6	g_7
1	90	80	70	100	100	100	15
2	90	70	80	100	100	100	95
3	80	90	70	100	100	100	25

	g_8	g_9	g_{10}	g_{11}	g_{12}	g_{13}	g_{14}
1	10000	11000	12000	20000	20000	20000	20000
2	10000	11000	12000	20000	20000	20000	20000
3	10000	11000	12000	20000	20000	18500	20000

	g_{15}	g_{16}	g_{17}	g_{18}	g_{19}	g_{20}	g_{21}
1	20000	20000	20000	20000	20000	19010	18005
2	20000	20000	20000	12000	12000	19085	14106
3	20000	20000	20000	20000	20000	19010	19496

Then, we have the set of goods $O' = O \cup \{g_1, \ldots, g_{21}\}$.

Let the goods be in the following order:

$$s, t_1, f_1, t_2, f_2, \ldots, t_n, f_n, r, g_1, \ldots, g_{21}.$$

We now present the final lemma that will give us our result.

Lemma 4.3.13. *If there exists a partial allocation $\mathcal{A}^{2n+2}$ over the first $2n+2$ goods such that $v(A_1^{2n+2}) = v(A_2^{2n+2})$, then there exists a* TEF1 *allocation $\mathcal{A}$. Conversely, if there does not exist a partial allocation $\mathcal{A}^{2n+2}$ over the first $2n+2$ goods such that $v(A_1^{2n+2}) = v(A_2^{2n+2})$, then there does not exists a* TEF1 *allocation $\mathcal{A}$.*

We use a program as a gadget to verify the lemma (see Appendix A.1), leveraging its output to support its correctness. Specifically, if there exists a partial allocation $\mathcal{A}^{2n+2}$ over the first $2n+2$ goods such that $v(A_1^{2n+2}) = v(A_2^{2n+2})$, then our program will show the existence of a TEF1 allocation by returning all such TEF1 allocations. If there does not exist such a partial allocation, our program essentially does an exhaustive search to show that a TEF1 allocation does not exist. This lemma shows that there exists a TEF1 allocation over O' if and only if $v(A_1^{2n+2}) \neq v(A_2^{2n+2})$, and by Claim 4.3.12, this implies that a TEF1 allocation over O' exists if and only if there is a truth assignment α such that each clause in F has exactly one `True` literal. $\square$

However, we note that the above approach cannot be extended to show hardness for the setting with chores. Nevertheless, we are able to show a similar, though weaker, intractability result for the case of chores in general. The key difference is that we assume that we can start from any partial TEF1 allocation.

Theorem 4.3.14. *For every $t \in [T]$, given any partial* TEF1 *allocation $\mathcal{A}^t$ for chores, deciding if there exists an allocation $\mathcal{A}$ that is* TEF1 *is* NP*-hard.*

Proof. We reduce from the NP-hard problem Partition. An instance of this problem consists of a multiset S of positive integers; it is a yes-instance if S can be partitioned into two subsets S_1 and S_2 such that the sum of the numbers in S_1 equals the sum of the numbers in S_2, and a no-instance otherwise.

Consider an instance of Partition given by a multiset set $S = \{s_1, \dots, s_m\}$ of m positive integers. Then, we construct a set $S' = \{s'_1, \dots s'_m\}$ such that for each $j \in [m]$, $s'_j = s_m - K$ where $K := \max\{s_1, \dots, s_m\} + \varepsilon$ for some small $\varepsilon > 0$. We then scale members of S' such that they sum to -2, i.e., $\sum_{s' \in S'} s' = -2$.

Next, we construct an instance with four agents and $m+4$ chores $O = \{b_1, b_2, b_3, b_4, c_1, \dots, c_m\}$, where agents have the following valuation profile $\mathbf{v}$ for $j \in \{1, \dots, m\}$:

$\mathbf{v}$	b_1	b_2	b_3	b_4	c_1	$\dots$	c_j	$\dots$	c_m
1	-1	0	0	0	-1	$\dots$	-1	$\dots$	-1
2	-1	-1	-1	-1	s'_1	$\dots$	s'_j	$\dots$	s'_m
3	-1	-1	-1	-1	s'_1	$\dots$	s'_j	$\dots$	s'_m
4	0	0	0	-1	-1	$\dots$	-1	$\dots$	-1

Also, suppose we are given the partial allocation $\mathcal{A}^4$ where for each $i \in \{1, 2, 3, 4\}$, chore b_i is allocated to agent i, as illustrated in the table above. Note that the partial allocation $\mathcal{A}^4$ is TEF1.

We first establish the following two lemmas. The first lemma states that after chores b_1, b_2, b_3, b_4 are allocated, in order to maintain TEF1, each remaining chore in $\{c_1, \dots, c_m\}$ cannot be allocated to either agent 1 or agent 4. The result is as follows.

Lemma 4.3.15. *In any* TEF1 *allocation, agents* 1 *and* 4 *cannot be allocated any chore in* $\{c_1, \dots, c_m\}$.

Proof. Consider any TEF1 allocation $\mathcal{A}$. Suppose for a contradiction that at least one of agent 1 and 4 is allocated a chore in $\{c_1, \dots, c_m\}$. Assume without loss of generality that agent 1 was the first (if not only) agent that received such a chore.

Consider the first round $j + 4$ (for some $j \in [m]$) whereby agent 1 is allocated some chore $c_j \in \{c_1, \dots, c_m\}$. Then,

$$v_1(A_1^{j+4} \setminus \{b_1\}) = -1 < 0 = v_1(A_4^{j+4}),$$

a contradiction to $\mathcal{A}$ being TEF1. □

The second lemma states that in any TEF1 allocation, the sum of values that agents 2 and 3 obtain from the chores in $\{c_1, \dots, c_m\}$ that are allocated to them must be equal. We formalize it as follows.

Lemma 4.3.16. *In any* TEF1 *allocation, let* C_2, C_3 *be the subsets of* $\{c_1, \dots, c_m\}$ *that were allocated to agents* 2 *and* 3 *respectively. Then,* $v_2(C_2) = v_3(C_3)$.

Proof. Consider any TEF1 allocation $\mathcal{A}$. Suppose for a contradiction that $v_2(C_2) \neq v_3(C_3)$. Since $v_2(C_2) + v_3(C_3) = \sum_{s' \in S'} s' = -2$, it means one of $\{v_2(C_2), v_3(C_3)\}$ is strictly less than -1, and the other is strictly more than -1. Without loss of generality, assume $v_2(C_2) > v_3(C_3)$, i.e., $v_3(C_3) < -1$. We get that

$$v_3(A_3 \setminus \{b_3\}) = v_3(C_3) < -1 = v_3(A_1),$$

contradicting the fact that $\mathcal{A}$ is a TEF1 allocation. □

We will now prove that there exists an allocation $\mathcal{A}$ satisfying TEF1 if and only if the set S can be partitioned into two subsets of equal sum.

For the 'if' direction, suppose $S = \{s_1, \ldots, s_m\}$ can be partitioned into two subsets S_1, S_2 of equal sum. This means that $S' = \{s'_1, \ldots, s'_m\}$ can be correspondingly partitioned into two subsets S'_1, S'_2 of equal sum (of -1 each). Let C_1, C_2 be the partition of chores in $\{c_1, \ldots, c_m\}$ with values corresponding to the partitions S'_1, S'_2 respectively. Then we allocate all chores in C_1 to agent 2 and all chores in C_2 to agent 3. By Lemma 4.3.15, we have that agents 1 and 4 cannot envy any other agent at any round. Also, for any round $t \in [T]$ and $i, j \in \{2, 3\}$ where $i \neq j$, $v_i(A_i^t \setminus \{b_i\}) \geq -1 \geq v_i(A_j^t)$, and for all $i \in \{2, 3\}$ and $k \in \{1, 4\}$, $v_i(A_i^t \setminus \{b_i\}) \geq -1 = v_i(A_k^t)$. Thus, the allocation $\mathcal{A}$ that, for each $i \in \{1, 2, 3, 4\}$, allocates b_i to agent i and for each $j \in \{2, 3\}$, allocates C_j to agent j, is TEF1.

For the 'only if' direction, suppose we have an allocation $\mathcal{A}$ satisfying TEF1. By Lemma 4.3.15, it must be that any chore in $\{c_1, \ldots, c_m\}$ is allocated to either agent 2 or 3. Let C_2, C_3 be the subsets of chores in $\{c_1, \ldots, c_m\}$ that are allocated to agents 2 and 3 respectively, under $\mathcal{A}$. Then, by Lemma 4.3.16, we have that $v_2(C_2) = v_3(C_3)$. By replacing the chores with their corresponding values, we get a partition of S' into two subsets of equal sums, which in turn gives us a partition of S into two subsets of equal sum. □

4.4 Compatibility of TEF1 and Efficiency

In traditional fair division, many papers have focused on the existence and computation of fair and efficient allocations for goods or chores, with a particular emphasis on simultaneously achieving EF1 and PO [Barman et al., 2018a, Caragiannis et al., 2019].

Observe that for any $\mathcal{A}$ that is PO, any partial allocation $\mathcal{A}^t$ for $t \leq [T]$ is necessarily PO as well. We demonstrate that PO is incompatible with TEF1 in this setting, even under very strong assumptions (of two agents and two types of items), as illustrated by the following result.

Proposition 4.4.1. *For any $n \geq 2$, a* TEF1 *and* PO *allocation for goods or chores may not exist, even when there are two types of items.*

Proof. We first prove the result for goods. Consider an instance with two agents and four goods $O = \{g_1, g_2, g_3, g_4\}$, with the following valuation profile:

$\mathbf{v}$	g_1	g_2	g_3	g_4
1	1.1	1.1	2	2
2	2	2	1.1	1.1

Observe that the first two goods must be allocated to different agents, otherwise TEF1 will be violated after the second good is allocated. Without loss of generality, suppose that agent 1 receives g_1 and agent 2 receives g_2. We have $v_1(g_1) < v_1(\{g_2, g_3, g_4\}) - v_1(g_3)$ and $v_2(g_2) < v_2(\{g_1, g_3, g_4\}) - v_2(g_1)$, thereby showing that EF1 will be violated if g_3 and g_4 are allocated to the same agent.

Thus, in any TEF1 allocation $\mathcal{A}$, agent 1 must receive one good from $\{g_1, g_2\}$ and one good from $\{g_3, g_4\}$. However, observe that every such allocation $\mathcal{A}$ is Pareto-dominated by the allocation where agent 2 receives bundle $\{g_1, g_2\}$ and agent 1 receives bundle $\{g_3, g_4\}$. This proof can be extended to the case of $n \geq 3$ simply by adding dummy agents who have zero value for each good, and observing that they cannot receive any item in a PO allocation. As such, a TEF1 and PO allocation cannot be guaranteed to exist, even when when there are two types of chores.

Next, we prove the result for chores. Consider an instance with $n \geq 2$ agents and $2n$ chores $O = \{c_1, \ldots, c_{2n}\}$, with the following valuation profile:

$\mathbf{v}$	c_1	$\ldots$	c_n	c_{n+1}	$\ldots$	c_{2n}
1	-1.1	$\ldots$	-1.1	-2	$\ldots$	-2
2	-2	$\ldots$	-2	-1.1	$\ldots$	-1.1
3	-2	$\ldots$	-2	-2	$\ldots$	-2
$\vdots$	$\vdots$		$\vdots$	$\vdots$		$\vdots$
n	-2	$\ldots$	-2	-2	$\ldots$	-2

In this instance, agent 1 has value -1.1 for each of the first n chores, and value -2 for the last n chores. Agent 2 has value -2 for the first n chores, and value -1.1 for the last n chores. If $n \geq 3$, then agents $3, \ldots, n$ have value -2 for all chores.

Observe that each agent must receive one of the first n chores to avoid violating TEF1 within the first n rounds. We now show that each agent must also receive one of the final n chores, otherwise TEF1 will be violated. Suppose for contradiction that in the final allocation $\mathcal{A}$, some agent $i \in N$ is allocated at least two chores from $\{c_{n+1}, \ldots, c_{2n}\}$. Then for each $i \in N$, let $c'_i := \arg\min_{c \in A_i} v_i(c)$. We get that

$$v_i(A_i \setminus \{c'_i\}) \leq \begin{cases} -5.1 + 2 = -3.1 & \text{if } i = 1, \\ -4.2 + 2 = -2.2 & \text{if } i = 2, \\ -6 + 2 = -4 & \text{if } i \in \{3, \ldots, n\}. \end{cases} \tag{4.12}$$

By the pigeonhole principle, there exists some other $j \in N \setminus \{i\}$ that receives no chore from $\{c_{n+1}, \dots, c_{2n}\}$, giving us

$$v_i(A_j) = \begin{cases} -1.1 & \text{if } i = 1, \\ -2 & \text{if } i = 2, \\ -2 & \text{if } i \in \{3, \dots, n\}. \end{cases} \tag{4.13}$$

Consequently, agent i would envy agent j even after removing one chore from her own bundle, and TEF1 is violated. Thus, in any TEF1 allocation, each agent must receive exactly one chore from $\{c_1, \dots, c_n\}$ and exactly one chore out of $\{c_{n+1}, \dots, c_{2n}\}$.

However, any such allocation is Pareto-dominated by another allocation where agent 1 receives exactly two chores from $\{c_1, \dots, c_n\}$ and no chores from $\{c_{n+1}, \dots, c_{2n}\}$, and agent 2 receives no chores from $\{c_1, \dots, c_n\}$ and exactly two chores from $\{c_{n+1}, \dots, c_{2n}\}$. As such, a TEF1 and PO allocation cannot be guaranteed to exist, even when there are two types of chores. □

Despite this non-existence result, one may still wish to obtain a TEF1 and PO outcome when the instance admits one. However, the following results show that this is not computationally tractable.

Theorem 4.4.2. *Determining whether there exists a* TEF1 *allocation that is* PO *for goods is* NP*-hard, even when $n = 2$.*

Proof. We reduce from the NP-hard problem 1-IN-3-SAT. An instance of this problem consists of conjunctive normal form F with three literals per clause; it is a yes-instance if there exists a truth assignment to the variables such that each clause has exactly one `True` literal, and a no-instance otherwise.

Consider an instance of 1-IN-3-SAT given by the CNF F which contains n variables $\{x_1, \dots, x_n\}$ and m clauses $\{C_1, \dots, C_m\}$.

We construct an instance $\mathcal{I}$ with two agents and $2n+1$ goods. For each $i \in [n]$, we introduce two goods t_i, f_i. We also introduce an additional good r. Let agents' (identical) valuations be defined as follows:

$$v(g) = \begin{cases} 5^{m+n-i} + \sum_{j\,:\, x_i \in C_j} 5^{m-j}, & \text{if } g = t_i, \\ 5^{m+n-i} + \sum_{j\,:\, \neg x_i \in C_j} 5^{m-j}, & \text{if } g = f_i, \\ \sum_{j \in [m]} 5^{j-1}, & \text{if } g = r. \end{cases}$$

Intuitively, for each variable index $i \in [n]$, we associate with it a unique value 5^{m+n-i}. For each clause index $j \in [m]$, we also associate with it a unique value 5^{m-j}. Note that no two indices (regardless of whether its a variable or clause index) share the same value, hence the uniqueness of the values. Then, the value for each good t_i comprises of the unique value associated with i, and the sum over all unique values of clauses C_j which x_i appears as a *positive literal* in; whereas the value for each good f_i comprises of the unique value associated with i, and the sum over all unique values of clauses C_j which x_i appears as a *negative literal* in. We will utilize this in our analysis later.

Then, we have the set of goods $O = \{t_1, f_1, t_2, f_2, \ldots, t_n, f_n, r\}$. Note that

$$v(O) = v(r) + \sum_{i \in [n]} v(t_i) + \sum_{i \in [n]} v(f_i).$$

Also observe that

$$\sum_{i \in [n]} 5^{m+n-i} = \sum_{i \in [n]} 5^{m+i-1}.$$

Now, as each clause contains exactly three literals,

$$\sum_{i \in [n]} \sum_{j : x_i \in C_j} 5^{m-j} + \sum_{i \in [n]} \sum_{j : \neg x_i \in C_j} 5^{m-j} = 3 \times \sum_{j \in [m]} 5^{j-1}.$$

Then, combining the equations above, we get that

$$v(O) = 2 \times \sum_{i \in [n]} 5^{m+i-1} + 4 \times \sum_{j \in [m]} 5^{j-1}. \tag{4.14}$$

Let the goods appear in the following order:

$$t_1, f_1, t_2, f_2, \ldots, t_n, f_n, r.$$

We first prove the following result.

Lemma 4.4.3. *There exists a truth assignment α such that each clause in F has exactly one* `True` *literal if and only if there exists an allocation $\mathcal{A} = (A_1,\ A_2)$ such that $v(A_1) = v(A_2)$ for instance $\mathcal{I}$.*

Proof. For the 'if' direction, consider an allocation $\mathcal{A}$ such that $v(A_1) = v(A_2)$. Since agents have identical valuations, without loss of generality, let $r \in A_1$. Since $O = A_1 \cup A_2$ and $v(A_1) = v(A_2) = \frac{1}{2}v(O)$, we have that

$$v(A_1 \setminus \{r\}) = \left(\sum_{i \in [n]} 5^{m+i-1} + 2 \times \sum_{j \in [m]} 5^{j-1} \right) - \sum_{j \in [m]} 5^{j-1} = \sum_{i \in [n]} 5^{m+i-1} + \sum_{j \in [m]} 5^{j-1}.$$

Note that this is only possible if for each $i \in [m]$, t_i and f_i are allocated to different agents. The reason is because the only way agent 1 can obtain the $\sum_{i \in [n]} 5^{m+i-1}$ term of the above bundle value is if he is allocated exactly one good from each of $\{t_i, f_i\}$ for all $i \in [n]$.

Then, from the goods that exist in bundle A_1, we can construct an assignment α: for each $i \in [n]$, let $x_i = \texttt{True}$ if $t_i \in A_1$ and $x_i = \texttt{False}$ if $f_i \in A_1$. Then, from the second term in the expression of $v(A_1 \setminus \{r\})$ above, we can observe that each clause must have exactly one `True` literal.

For the 'only if' direction, consider a truth assignment α such that each clause in F has exactly one `True` literal. Then, for each $i \in [n]$, let

$$\ell_i = \begin{cases} t_i & \text{if } x_i = \texttt{True} \text{ under } \alpha, \\ f_i & \text{if } x_i = \texttt{False} \text{ under } \alpha. \end{cases}$$

We construct the allocation $\mathcal{A} = (A_1, A_2)$ where

$$A_1 = \{\ell_1, \dots, \ell_n, r\} \quad \text{and} \quad A_2 = O \setminus A_1.$$

Again, observe that

$$\sum_{i \in [n]} 5^{m+n-i} = \sum_{i \in [n]} 5^{m+i-1}.$$

Then, as each clause has exactly one `True` literal,

$$v(A_1) = \sum_{i \in [n]} 5^{m+i-1} + 2 \times \sum_{j \in [m]} 5^{j-1},$$

and together with (4.14), we get that

$$v(A_2) = v(O) - v(A_1) = v(A_1),$$

as desired. □

Note that for all values of $m, n \geq 1$, and some $\varepsilon < \frac{1}{3}$,

$$5^{m+n} - 2\varepsilon > 5^{m+n-1} + \frac{5^m - 1}{4} = 5^{m+n-1} + \sum_{j \in [m]} 5^{j-1} \geq \max_{g \in O} v(g). \tag{4.15}$$

Now consider another instance $\mathcal{I}'$ that is similar to $\mathcal{I}$, but with an additional four goods o_1, o_2, o_3, o_4. Let the agents' valuations over these four new goods be defined as follows, for some $\varepsilon < \frac{1}{3}$:

v	o_1	o_2	o_3	o_4
1	5^{m+n}	$5^{m+n} - \varepsilon$	$5^{m+n} - \varepsilon$	5^{m+n}
2	$5^{m+n} - \varepsilon$	5^{m+n}	5^{m+n}	$5^{m+n} - \varepsilon$

Then, we have the set of goods $O' = O \cup \{o_1, o_2, o_3, o_4\}$.

Let the goods be in the following order:

$$t_1, f_1, t_2, f_2, \ldots, t_n, f_n, r, o_1, o_2, o_3, o_4.$$

If there is a partial allocation $\mathcal{A}^{2n+1}$ over the first $2n+1$ goods such that $v(A_1^{2n+1}) = v(A_2^{2n+1})$, then by giving o_1, o_4 to agent 1 and o_2, o_3 to agent 2, we obtain an allocation that is TEF1 and PO (note that any allocation for the first $2n+1$ goods will be PO, since agents have identical valuations over them).

However, if there does not exist a partial allocation $\mathcal{A}^{2n+1}$ over the first $2n+1$ goods such that $v(A_1^{2n+1}) = v(A_2^{2n+1})$, then let $\mathcal{A}^{2n+1}$ be any partial allocation of the first $2n+1$ goods that is TEF1 but $v(A_1^{2n+1}) \neq v(A_2^{2n+1})$. We will show that if $v(A_1^{2n+1}) \neq v(A_2^{2n+1})$, any TEF1 allocation of O' cannot be PO.

Note that in order for $\mathcal{A}^{2n+1}$ to be TEF1, we must have that for any agent $i \in \{1, 2\}$,

$$v(A_i^{2n+1}) \geq \frac{v(O) - \max_{g \in O} v(g)}{2}. \tag{4.16}$$

This also means that for any agent $i \in \{1, 2\}$,

$$v(A_i^{2n+1}) \leq v(O) - \frac{v(O) - \max_{g \in O} v(g)}{2} = \frac{v(O) + \max_{g \in O} v(g)}{2}. \tag{4.17}$$

Also observe that since $\min_{g \in O} v(g) > \varepsilon$ and $v(A_1^{2n+1}) \neq v(A_2^{2n+1})$,

$$\left| v(A_1^{2n+1}) - v(A_2^{2n+1}) \right| > \varepsilon. \tag{4.18}$$

We split our analysis into two cases.

Case 1: $v(A_1^{2n+1}) > v(A_2^{2n+1})$. If we give o_1 to agent 1, since by (4.15), $v_2(o_1) > \max_{g \in A_1^{2n+1}} v(g)$, we get that

$$v_2(A_2^{2n+2}) = v(A_2^{2n+1}) < v(A_1^{2n+1}) = v_2(A_1^{2n+2} \setminus \{o_1\}),$$

and agent 2 will still envy agent 1 after dropping o_1 from agent 1's bundle. Thus, we must give o_1 to agent 2.

Next, if we give o_2 to agent 2, then since $v_1(o_1) > \max_{g \in O} v(g)$ and $v_1(o_1) > v_1(o_2)$, we have that

$$\begin{aligned}
v_1(A_1^{2n+3}) &= v(A_1^{2n+1}) \\
&\leq \frac{v(O) + \max_{g \in O} v(g)}{2} \quad \text{(by (4.17))} \\
&< \frac{v(O) - \max_{g \in O} v(g)}{2} + v_1(o_2) \quad \text{(by (4.15))} \\
&\leq v(A_2^{2n+1}) + v_1(o_2) \quad \text{(by (4.16))} \\
&= v_1(A_2^{2n+3} \setminus \{o_1\}),
\end{aligned}$$

and agent 1 will still envy agent 2 after dropping o_1 from agent 2's bundle. Thus, we must give o_2 to agent 1. However, such a partial allocation (and thus $\mathcal{A}$) will fail to be PO, as giving o_1 to agent 1 and o_2 to agent 2 instead will strictly increase the utility of both agents.

Case 2: $v(A_1^{2n+1}) < v(A_2^{2n+1})$. If we give o_1 to agent 2, since by (4.15), $v_1(o_1) > \max_{g \in A_2^{2n+1}} v(g)$, we get that

$$v_1(A_1^{2n+2}) = v(A_1^{2n+1}) < v(A_2^{2n+1}) = v_1(A^{2n+2} \setminus \{o_1\}),$$

and agent 1 will still envy agent 2 after dropping o_1 from agent 2's bundle. Thus, we must give o_1 to agent 1.

Next, if we give o_2 to agent 1, then since $v_2(o_2) > \max_{g \in O} v(g)$ and $v_2(o_2) > v_2(o_1)$, we have that

$$\begin{aligned}
v_2(A_2^{2n+3}) &= v(A_2^{2n+1}) \\
&\leq \frac{v(O) + \max_{g \in O} v(g)}{2} \quad \text{(by (4.17))} \\
&< \frac{v(O) - \max_{g \in O} v(g)}{2} + v_1(o_1) \quad \text{(by (4.15))} \\
&\leq v(A_1^{2n+1}) + v_1(o_1) \quad \text{(by (4.16))} \\
&= v_2(A_1^{2n+3} \setminus \{o_2\}),
\end{aligned}$$

and agent 2 will still envy agent 1 after dropping o_2 from agent 1's bundle. Thus, we must give o_2 to agent 2.

Now, if we give o_3 to agent 2, then since $v_1(o_3) > \max_{g \in O} v(g)$ and $v_1(o_3) = v_1(o_2)$, we have that

$$\begin{aligned}
v_1(A_1^{2n+4}) &= v(A_1^{2n+1}) + v_1(o_1) \\
&< v(A_2^{2n+1}) - \varepsilon + v_1(o_1) \quad \text{(by (4.18))} \\
&= v(A_2^{2n+1}) + v_1(o_2) \\
&= v_1(A_2^{2n+4} \setminus \{o_3\}),
\end{aligned}$$

and agent 1 will still envy agent 2 after dropping o_3 from agent 2's bundle. Thus, we must give o_3 to agent 1.

Finally, if we give o_4 to agent 1, then since $v_2(o_3) > \max_{g \in O} v(g)$ and $v_2(o_3) > v_2(o_1) = v_2(o_4)$, we have that

$$\begin{aligned}
v_2(A_2) &= v(A_2^{2n+1}) + v_2(o_2) \\
&= v(A_2^{2n+1}) + 5^{m+n} \\
&\leq \frac{v(O) + \max_{g \in O} v(g)}{2} + 5^{m+n} \quad \text{(by (4.17))} \\
&< \frac{v(O) - \max_{g \in O} v(g)}{2} + 2 \times 5^{m+n} - 2\varepsilon \quad \text{(by (4.15))} \\
&\leq v(A_1^{2n+1}) + 2 \times 5^{m+n} - 2\varepsilon \quad \text{(by (4.16))} \\
&= v(A_1^{2n+1}) + v_2(\{o_1, o_4\}) \\
&= v_2(A_1 \setminus \{o_3\}),
\end{aligned}$$

and agent 2 will still envy agent 1 after dropping o_3 from agent 1's bundle. Thus, we must give o_4 to agent 2. However, again, this is not PO as giving o_3 to agent 2 and o_4 to agent 1 will strictly increase the utility of both agents.

By exhaustion of cases, we have shown that if $v(A_1^{2n+1}) \neq v(A_2^{2n+1})$, there does not exist a TEF1 and PO allocation over O'. Thus, a TEF1 and PO allocation over O' exists if and only if $v(A_1^{2n+1}) \neq v(A_2^{2n+1})$. By Lemma 4.4.3, this implies that a TEF1 and PO allocation over O' exists if and only if there is a truth assignment α such that each clause in F has exactly one `True` literal. □

Theorem 4.4.4. *Determining whether there exists a* TEF1 *allocation that is* PO *for chores is* NP*-hard, even when* $n = 2$.

Proof. We reduce from the NP-hard problem 1-IN-3-SAT. An instance of this problem consists of conjunctive normal form F with three literals per clause; it is a yes-instance if there exists a truth assignment to the variables such that each clause has exactly one `True` literal, and a no-instance otherwise.

Consider an instance of 1-IN-3-SAT given by the CNF F which contains n variables $\{x_1, \ldots, x_n\}$ and m clauses $\{C_1, \ldots, C_m\}$.

We construct an instance $\mathcal{I}$ with two agents and $2n+1$ chores. For each $i \in [n]$, we introduce two chores t_i, f_i. We also introduce an additional chore r. Let agents' (identical) valuations be defined as follows:

$$v(c) = \begin{cases} -5^{m+n-i} - \sum_{j\,:\,x_i \in C_j} -5^{m-j}, & \text{if } c = t_i, \\ -5^{m+n-i} - \sum_{j\,:\,\neg x_i \in C_j} 5^{m-j}, & \text{if } c = f_i, \\ -\sum_{j \in [m]} 5^{j-1}, & \text{if } c = r. \end{cases}$$

Intuitively, for each variable index $i \in [n]$, we associate with it a unique value -5^{m+n-i}. For each clause index $j \in [m]$, we also associate it with a unique number -5^{m-j}. Note that no two indices (regardless of whether its a variable or clause index) share the same value, hence the term unique value. Then, the value for each chore t_i comprises of the unique value associated with i, and the sum over all unique values of clauses C_j which x_i appears as a *positive literal* in; whereas the value for each chore f_i comprises of the unique value associated with i, and the sum over all unique values of clauses C_j which x_i appears as a *negative literal* in. We will utilize this in our analysis later.

Then, we have that the set of chores $O = \{t_1, f_1, t_2, f_2, \ldots, t_n, f_n, r\}$. Note that

$$v(O) = v(r) + \sum_{i \in [n]} v(t_i) + \sum_{i \in [n]} v(f_i).$$

Also observe that

$$-\sum_{i \in [n]} 5^{m+n-i} = -\sum_{i \in [n]} 5^{m+i-1}.$$

Now, as each clause contains exactly three literals,

$$-\sum_{i \in [n]} \sum_{j:x_i \in C_j} 5^{m-j} - \sum_{i \in [n]} \sum_{j:\neg x_i \in C_j} 5^{m-j} = 3 \times -\sum_{j \in [m]} 5^{j-1}.$$

Then, combining the equations above, we get that

$$v(O) = 2 \times -\sum_{i \in [n]} 5^{m+i-1} + 4 \times -\sum_{j \in [m]} 5^{j-1}. \tag{4.19}$$

Let the chores appear in the following order:

$$t_1, f_1, t_2, f_2, \ldots, t_n, f_n, r.$$

We first prove the following result.

Lemma 4.4.5. *There exists a truth assignment α such that each clause in F has exactly one* `True` *literal if and only if there exists an allocation $\mathcal{A} = (A_1, A_2)$ such that $v(A_1) = v(A_2)$ for instance $\mathcal{I}$.*

Proof. For the 'if' direction, consider an allocation $\mathcal{A}$ such that $v(A_1) = v(A_2)$. Since agents have identical valuations, without loss of generality, let $r \in A_1$. Since $O = A_1 \cup A_2$ and $v(A_1) = v(A_2) = \frac{1}{2}v(O)$, we have that

$$v(A_1\backslash\{r\}) = \left(-\sum_{i\in[n]} 5^{m+i-1} + 2 \times -\sum_{j\in[m]} 5^{j-1}\right)+\sum_{j\in[m]} 5^{j-1} = -\sum_{i\in[n]} 5^{m+i-1}-\sum_{j\in[m]} 5^{j-1}.$$

Note that this is only possible if for each $I \in [m]$, t_i and f_i are allocated to different agents. The reason is because the only way agent 1 can obtain the first term of the above bundle value (less chore r) is if she is allocated exactly one chore from each of $\{t_i, f_i\}$ for each $i \in [n]$.

Then, from the chores that exists in bundle A_1, we can construct an assignment α: for each $i \in [n]$, let $x_i =$ `True` if $t_i \in A_1$ and $x_i =$ `False` if $f_i \in A_1$. Then, from the second term in the expression of $v(A_1 \setminus \{r\})$ above, we can observe that each clause has exactly one `True` literal (because the sum is only obtainable if exactly one literal appears in each clause, and our assignment will cause each these literals to evaluate `True`.

For the 'only if' direction, consider a truth assignment α such that each clause in F has exactly one `True` literal. Then, for each $i \in [n]$, let

$$\ell_i = \begin{cases} t_i & \text{if } x_i = \texttt{True} \text{ under } \alpha, \\ f_i & \text{if } x_i = \texttt{False} \text{ under } \alpha. \end{cases}$$

We construct the allocation $\mathcal{A} = (A_1, A_2)$ where

$$A_1 = \{\ell_1, \ldots, \ell_n, r\} \quad \text{and} \quad A_2 = O \setminus A_1.$$

Again, observe that

$$-\sum_{i\in[n]} 5^{m+n-i} = -\sum_{i\in[n]} 5^{m+i-1}.$$

Then, as each clause has exactly one `True` literal,

$$v(A_1) = -\sum_{i\in[n]} 5^{m+i-1} + 2 \times -\sum_{j\in[m]} 5^{j-1},$$

and together with (4.14), we get that

$$v(A_2) = v(O) - v(A_1) = v(A_1),$$

as desired. □

Note that for all values of $m, n \geq 1$, and some $\varepsilon < \frac{1}{3}$,

$$\frac{-5^{m+n}+2\varepsilon}{2} < -5^{m+n-1} - \frac{5^m-1}{4} = -5^{m+n-1} - \sum_{j\in[m]} 5^{j-1} \leq \min_{c\in O} v(c). \tag{4.20}$$

Now, consider another instance $\mathcal{I}'$ that is similar to $\mathcal{I}$, but with an additional four chores o_1, o_2, o_3, o_4. Let agents' valuations over these four new chores be defined as follows, for some $\varepsilon < \frac{1}{3}$:

$\mathbf{v}$	o_1	o_2	o_3	o_4
1	-5^{m+n}	$-5^{m+n}+\varepsilon$	$-5^{m+n}+\varepsilon$	-5^{m+n}
2	$-5^{m+n}+\varepsilon$	-5^{m+n}	-5^{m+n}	$-5^{m+n}+\varepsilon$

Then, we have the set of chores $O' = O \cup \{o_1, o_2, o_3, o_4\}$.

Let the chores be in the following order:

$$t_1, f_1, t_2, f_2, \ldots, t_n, f_n, r, o_1, o_2, o_3, o_4.$$

If there is a partial allocation $\mathcal{A}^{2n+1}$ over the first $2n+1$ chores such that $v(A_1^{2n+1}) = v(A_2^{2n+1})$, then by giving o_1, o_4 to agent 1 and o_2, o_3 to agent 2, we obtain an allocation that is TEF1 and PO (note that any allocation for the first $2n+1$ chores will be PO, since agents have identical valuations over them).

However, if there does not exist a partial allocation $\mathcal{A}^{2n+1}$ over the first $2n+1$ goods such that $v(A_1^{2n+1}) = v(A_2^{2n+1})$, then let $\mathcal{A}^{2n+1}$ be any partial allocation of the first $2n+1$ goods that is TEF1 but $v(A_1^{2n+1}) \neq v(A_2^{2n+1})$.

Note that in order for $\mathcal{A}^{2n+1}$ to be TEF1, we must have that for any agent $i \in \{1, 2\}$,

$$v(A_i^{2n+1}) - \min_{c\in O} v(c) \geq \frac{v(O)}{2}. \tag{4.21}$$

This also means that for any agent $i \in \{1, 2\}$,

$$v(A_i^{2n+1}) \leq v(O) - \left(\frac{v(O)}{2} + \min_{c\in O} v(c)\right) = \frac{v(O)}{2} - \min_{c\in O} v(c). \tag{4.22}$$

Also observe that since $\min_{g\in O} v(g) > \varepsilon$ and $v(A_1^{2n+1}) \neq v(A_2^{2n+1})$,

$$\left|v(A_1^{2n+1}) - v(A_2^{2n+1})\right| > \varepsilon. \tag{4.23}$$

We split our analysis into two cases.

Case 1: $v(A_1^{2n+1}) > v(A_2^{2n+1})$**.** If we give o_1 to agent 2, since by (4.20), $v_2(o_1) < \min_{c \in A_2^{2n+1}} v(c)$, we get that

$$v_2(A_2^{2n+2} \setminus \{o_1\}) = v(A_2^{2n+1}) < v(A_1^{2n+1}) = v_2(A_1^{2n+2}),$$

and agent 2 will still envy agent 1 after dropping o_1 from his own bundle. Thus, we must give o_1 to agent 1.

Next, if we give o_2 to agent 1, then since $v_1(o_1) < \max_{c \in O} v(c)$ and $v_1(o_1) < v_1(o_2)$, we have that

$$\begin{aligned} v_1(A_1^{2n+3} \setminus \{o_1\}) &= v(A_1^{2n+1}) + v_1(o_2) \\ &\le \frac{v(O)}{2} - \min_{c \in O} v(c) + v_1(o_2) \quad \text{(by (4.22))} \\ &< \frac{v(O)}{2} + \min_{c \in O} v(c) \quad \text{(by (4.20))} \\ &\le v(A_2^{2n+1}) \quad \text{(by (4.21))} \\ &= v_1(A_2^{2n+3}), \end{aligned}$$

and agent 1 will still envy agent 2 after dropping o_2 from her own bundle. Thus, we must give o_2 to agent 2. However, such a partial allocation (and thus $\mathcal{A}$) will fail to be PO, as giving o_1 to agent 2 and o_2 to agent 1 will strictly increase the utility of both agents.

Case 2: $v(A_1^{2n+1}) < v(A_2^{2n+1})$**.** If we give o_1 to agent 1, since by (4.20), $v_1(o_1) \min_{c \in A_1^{2n+1}} v(c)$, we get that

$$v_1(A_1^{2n+2} \setminus \{o_1\}) = v(A_1^{2n+1}) < v(A_2^{2n+1}) = v_1(A^{2n+2}),$$

and agent 1 will still envy agent 2 after dropping o_1 from her own bundle. Thus, we must give o_1 to agent 2.

Next, if we give o_2 to agent 2, then since $v_2(o_2) < \min_{c \in O} v(c)$ and $v_2(o_2) < v_2(o_1)$, we have that

$$\begin{aligned} v_2(A_2^{2n+3} \setminus \{o_2\}) &= v(A_2^{2n+1}) + v_2(o_1) \\ &\le \frac{v(O)}{2} - \min_{c \in O} v(c) + v_2(o_1) \quad \text{(by (4.22))} \\ &< \frac{v(O)}{2} + \min_{c \in O} v(c) \quad \text{(by (4.20))} \\ &\le v(A_1^{2n+1}) \quad \text{(by (4.21))} \\ &= v_2(A_1^{2n+3}), \end{aligned}$$

and agent 2 will still envy agent 1 after dropping o_2 from his own bundle. Thus, we must give o_2 to agent 1.

Now, if we give o_3 to agent 1, then since $v_1(o_3) < \min_{c \in O} v(c)$ and $v_1(o_3) = v_1(o_2)$, we have that

$$\begin{aligned}
v_1(A_1^{2n+4} \setminus \{o_3\}) &= v(A_1^{2n+1}) + v_1(o_2) \\
&< v(A_2^{2n+1}) - \varepsilon + v_1(o_2) \quad \text{(by (4.23))} \\
&= v(A_2^{2n+1}) + v_1(o_1) \\
&= v_1(A_2^{2n+4}),
\end{aligned}$$

and agent 1 will still envy agent 2 after dropping o_3 from her own bundle. Thus, we must give o_3 to agent 2.

Finally, if we give o_4 to agent 2, then since $v_2(o_3) < \min_{c \in O} v(c)$ and $v_2(o_3) < v_2(o_1) = v_2(o_4)$, we have that

$$\begin{aligned}
v_2(A_2 \setminus \{o_3\}) &= v(A_2^{2n+1}) + v_2(\{o_1, o_4\}) \\
&= v(A_2^{2n+1}) - 2 \times 5^{m+n} + 2\varepsilon \\
&\leq \frac{v(O)}{2} - \min_{c \in O} v(c) - 2 \times 5^{m+n} + 2\varepsilon \quad \text{(by (4.22))} \\
&< \frac{v(O)}{2} + \min_{c \in O} v(c) - 5^{m+n} + \varepsilon \quad \text{(by (4.20))} \\
&\leq v(A_1^{2n+1}) + v_2(o_1) \quad \text{(by (4.21))} \\
&= v(A_1^{2n+1}) + v_2(o_1) \\
&= v_2(A_1),
\end{aligned}$$

and agent 2 will still envy agent 1 after dropping o_3 from his own bundle. Thus, we must give o_4 to agent 1. However, again, this is not PO as giving o_3 to agent 1 and o_4 to agent 2 will strictly increase the utility of both agents.

By exhaustion of cases, we have shown that if $v(A_1^{2n+1}) \neq v(A_2^{2n+1})$, there does not exist a TEF1 and PO allocation over O'. Thus, a TEF1 and PO allocation over O' exists if and only if $v(A_1^{2n+1}) \neq v(A_2^{2n+1})$. By Lemma 4.4.5, this implies that a TEF1 and PO allocation over O' exists if and only if there is a truth assignment α such that each clause in F has exactly one `True` literal. □

The proof of the above result essentially imply that even determining whether an instance admits a TEF1 and *utilitarian-maximizing* (i.e., sum of agents' utilities) allocation is computationally intractable, since a utilitarian-welfare maximizing allocation is necessarily PO. In fact, for the case of goods, we can make a stronger statement relating to the general class of *p-mean welfares*, defined as follows.[3]

Definition 4.4.6. Given $p \in (-\infty, 1]$ and an allocation $\mathcal{A} = (A_1, \dots, A_n)$ of goods, the *p-mean welfare* is $\left(\frac{1}{n}\sum_{i \in N} v_i(A_i)^p\right)^{1/p}$.

In the context of fair division, p-mean welfare has been traditionally and well-studied for the setting with goods [Barman et al., 2020, Chaudhury et al., 2021], although it has recently been explored for chores as well [Eckart et al., 2024]. Importantly, p-means welfare captures a spectrum of commonly studied fairness objectives in fair division. For instance, setting $p = 1$ (resp. $p = -\infty$) would correspond to the utilitarian (resp. egalitarian) welfare. Setting $p \to 0$ corresponds to maximizing the geometric mean, which is also known as the Nash welfare [Caragiannis et al., 2019].

Then, from our construction in the proof of Theorem 4.4.2 (for goods), we have that an allocation is TEF1 and PO if and only if it also maximizes the p-mean welfare, for all $p \in (-\infty, 1]$, thereby giving us the following corollary.

Corollary 4.4.7. *For all $p \in (-\infty, 1]$, determining whether there exists a* TEF1 *allocation that maximizes p-mean welfare is* NP*-hard, even when $n = 2$.*

4.5 Multiple Items per Round

We now revisit the setting where multiple items may arrive at each round. While Lemma 4.3.1 reduces this case to the setting where a single item arrives per round, there are restricted variants of our problem that are not preserved by this reduction. We will now consider two such variants: $T = 2$ and repeated allocation.

We begin by showing that when there are two rounds, a TEF1 allocation can be computed efficiently.

Theorem 4.5.1. *When $T = 2$, a* TEF1 *allocation for goods or chores exists and can be computed in polynomial time.*

Proof. Let $\mathcal{A}^1$ and $\mathcal{B} = \mathcal{A}^2 \setminus \mathcal{A}^1$ be the allocations of item sets O_1 and O_2 respectively. Note that while we are in the setting whereby $O_1 = O_2$, we can simply relabel item.

[3]Note that we cannot say the same for chores as when agents' valuations are negative, the p-mean welfare may be ill-defined.

We first address a special case. When allocating chores, for each $t \in \{1, 2\}$ such that $|O_t| < n$ (i.e., there are less chores than agents in either round), add $n - |O_t|$ zero-valued dummy chores to O_t.

To obtain $\mathcal{A}^1$, we allocate the items in the first round in a round-robin fashion, with picking sequence $(1, \dots, n)^*$. That is, agent 1 picks their most preferred item, followed by agent 2, and so on until agent n, after which the sequence restarts. The items arriving in the second round are also allocated in a round-robin fashion to obtain $\mathcal{B}$, but with picking sequence $(n, \dots, 1)^*$. The round-robin algorithm is well-known to satisfy EF1 for both the goods and chores settings [Aziz et al., 2019], so we know that $\mathcal{A}^1$ and $\mathcal{B}$ are EF1. It remains to show that $\mathcal{A}^2 = \mathcal{A}^1 \cup \mathcal{B}$ is EF1.

Consider an arbitrary pair of agents i, j. If $i < j$, then

$$v_i(A_i^1) \geq v_i(A_j^1),$$

because i precedes j in the picking sequence for allocation $\mathcal{A}$. Similarly, if $i > j$, then

$$v_i(B_i) \geq v_i(B_j).$$

Note that these inequalities hold for both goods and chores.

We now prove our result for goods. Consider an arbitrary agent i. Since $\mathcal{A}^1$ and $\mathcal{B}$ are EF1, we know that for any agent $j \neq i$, there exists a good $g_a \in A_j^1$ such that $v_i(A_i^1) \geq v_i(A_j^1 \backslash \{g_a\})$, and there exists a good $g_b \in B_j$ such that $v_i(B_i) \geq v_i(B_j \backslash \{g_b\})$. Therefore for any agent $j < i$, there exists $g_a \in A_j^1$ such that

$$v_i(A_i^2) = v_i(A_i^1 \cup B_i) \geq v_i(A_j^1) - v_i(g_a) + v_i(B_j) = v_i(A_j^2) - v_i(g_a) = v_i(A_j^2 \setminus \{g_a\}).$$

Similarly, for any $j > i$, there exists $g_b \in B_j$ such that $v_i(A_i^2) \geq v_i(A_j^2 \setminus \{g_b\})$.

We next prove our result for chores. Again consider an arbitrary agent i. Due to $\mathcal{A}^1$ and $\mathcal{B}$ satisfying EF1, for any agent $j \neq i$, there exists a chore $c_a \in A_i^1$ such that $v_i(A_i^1 \backslash \{c_a\}) \geq v_i(A_j^1)$, and there exists a chore $c_b \in B_j$ such that $v_i(B_i \backslash \{c_b\}) \geq v_i(B_j)$. Therefore for any $j < i$, there exists $c_a \in A_i^1$ such that

$$v_i(A_i^2 \setminus \{c_a\}) = v_i(A_i^1 \setminus \{c_a\}) + v_i(B_i) \geq v_i(A_j^1) + v_i(B_j) = v_i(A_j^2).$$

Similarly, for any $j > i$, there exists $c_b \in B_i$ such that $v_i(A_i^2 \setminus \{c_b\}) \geq v_i(A_j^2)$. This concludes the proof. □

For the remainder of Section 4.5, we consider the *repeated* setting (also studied by Igarashi et al. [2024] and Caragiannis and Narang [2024]), where the sets $O_1, \dots, O_T$ are identical. Formally, for each $t \in T$ we have $O_t = \{o_1^t, \dots, o_k^t\}$, and $v_i(o_j^t) = v_i(o_j^r)$

for all $t, r \in [T]$ and all $i \in N$, $j \in [k]$. Note that this property of the instance is not preserved by our reduction from many items per round to a singe item per round.

In general, it remains an open question whether a TEF1 allocation exists for this setting. However, we can show that, perhaps surprisingly, it is NP-hard to determine whether there exists a TEF1 allocation that allocates the items in the *same way* at *every round.* We say that an allocation $\mathcal{A}$ is *repetitive* if for each $i \in N$, $j \in [k]$ and all $t, r \in [T]$ we have $o_j^t \in A_i^t \setminus A_i^{t-1}$ if and only if $o_j^r \in A_i^r \setminus A_i^{r-1}$. Then we have the following result.

Theorem 4.5.2. *Determining whether there exists a repetitive allocation $\mathcal{A} = (A_1, \ldots, A_n)$ that is* TEF1 *is* NP*-complete both for goods and for chores. The hardness result holds even if $T = 2$ and agents have identical valuations.*

Proof. It is immediate that this problem is in NP: we can guess a repetitive allocation, and check whether it is TEF1. Both for goods and for chores, we reduce from the NP-hard problem Multiway Number Partitioning [Graham, 1969]. An instance of this problem is given by a positive integer κ and a multiset $S = \{s_1, \ldots, s_\mu\}$ of μ non-negative integers whose sum is κW; it is a yes-instance if S can be partitioned into κ subsets such that the sum of integers in each subset is W, and a no-instance otherwise.

Consider an instance of Multiway Number Partitioning given by a positive integer κ and a multiset $S = \{s_1, \ldots, s_\mu\}$ of μ non-negative integers that sum up to κW.

We first prove the result for goods. We construct an instance with $\kappa + 1$ agents and $\mu + 1$ goods in each round: $O_1 = \{g_1^1, \ldots, g_{\mu+1}^1\}$ and $O_2 = \{g_1^2, \ldots, g_{\mu+1}^2\}$. The agents have an identical valuation function v defined as follows: $v(g_j^1) = v(g_j^2) = s_j$ if $j \in [\mu]$, and $v(g_{\mu+1}^1) = v(g_{\mu+1}^2) = 2W$. We will now prove that there exists a repetitive TEF1 allocation $\mathcal{A}$ if and only if the set S can be partitioned into κ subsets with equal sums (of W each).

For the 'if' direction, consider a κ-way partition $\mathcal{P} = \{P_1, \ldots, P_\kappa\}$ of S with $\sum_{s \in P_i} s = W$ for each $i \in [\kappa]$. We construct allocations $\mathcal{A}^1$ and $\mathcal{A}^2$ by allocating the goods corresponding to the elements of subset P_i to agent i for $i \in [\kappa]$; the goods $g_{\mu+1}^1$ and $g_{\mu+1}^2$ are allocated to agent $\kappa + 1$. Then, in $\mathcal{A}^1$, for each agent $i \in [\kappa]$ we have $v(A_i^1) = \sum_{s \in P_i} s = W$, and $v(A_{\kappa+1}^1) = v(g_{\mu+1}^1) = 2W$. It is easy to verify that $\mathcal{A}^1$ is EF1: no agent $i \in [\kappa]$ envies another agent $j \in [\kappa] \setminus \{i\}$, as they have the same bundle value, and agent i's envy towards agent $\kappa + 1$ can be removed by simply dropping

$g^1_{\mu+1}$ from $A^1_{\kappa+1}$. Also, agent $\kappa+1$ does not envy the first κ agents: she values her bundle at $2W$ and the bundles of $i \in [\kappa]$ at W.

Moreover in $\mathcal{A}^2$ each agent $i \in [\kappa]$ values the bundles $A^2_1, \dots, A^2_\kappa$ at $2W$ and hence does not envy any of the first κ agents; her envy towards $\kappa+1$ can be eliminated by dropping $g^1_{\mu+1}$ from $A^2_{\kappa+1}$. On the other hand, agent $\kappa+1$ values her bundle at $4W$ and all other bundles at $2W$, so she does not envy the first κ agents.

For the 'only if' direction, suppose we have a repetitive allocation $\mathcal{A}^2$ that satisfies TEF1. Since agents have identical valuation functions, we can assume without loss of generality that agent $\kappa+1$ receives goods $g^1_{\mu+1}$ and $g^2_{\mu+1}$ in rounds 1 and 2. Then for agent $i \in [\kappa]$ not to envy $\kappa+1$ in $\mathcal{A}^2$ after we drop one item from A^2_{k+1}, it has to be the case that $v_i(A^2_i) \geq 2W$. As this holds for all $i \in [\kappa]$ and $\sum_{j\in[\mu]} s_j = \kappa W$, this is only possible if there is a κ-way partition of S such that each subset sums up to W.

We now prove the result for the case of chores. We construct a set $S' = \{s'_1, \dots, s'_m\}$ such that for each $j \in [m]$, $s'_j = -K + s_j$ where $K := \max\{s_1, \dots, s_m\}$. Observe that S' contains non-positive integers. Let $W' := \frac{1}{\kappa}\sum_{j\in[m]} s'_j$

Then, we construct an instance with $\kappa+1$ agents and $m+1$ chores in each round: $O_1 = \{c_1, \dots, c_{m+1}\}$ and $O_2 = \{c'_1, \dots, c'_{m+1}\}$, where agents have an identical valuation function v defined as follows:

$$v(c_j) = v(c'_j) = \begin{cases} s'_j, & \text{if } j \leq m, \\ 2W', & \text{if } j = m+1. \end{cases}$$

We will now prove that there exists a repetitive TEF1 allocation $\mathcal{A}$ if and only if the set S can be partitioned into κ subsets with equal sums (of W each).

For the 'if' direction, consider a κ-way partition $\mathcal{P} = \{P_1, \dots, P_\kappa\}$ of S with equal sums (of W each). This means that S' can also be partitioned into κ subsets of equal sums (with the same partition $\mathcal{P}$; let the sum be W'). We construct allocations $\mathcal{A}^1$ and $\mathcal{A}^2$ such that the chores in both rounds are allocated identically, and show that $\mathcal{A}^2$ satisfies TEF1.

For each $i \in \{1, \dots, \kappa\}$, allocate the chores corresponding to the elements of subset P_i to agent i, and the chore c_{m+1} to agent $\kappa+1$. Then, in $\mathcal{A}^1$, for each agent $i \in [\kappa]$, $v(A^1_i) = \sum_{c\in P_i} c = W'$, and $v(A^1_{\kappa+1}) = v(\{c_{m+1}\}) = 2W'$. It is easy to verify that $\mathcal{A}^1$ is TEF1: every pair of agents $i, j \in [\kappa]$ has the same bundle value, and each agent $i \in [\kappa]$ has a higher bundle value than agent $\kappa+1$. Also, agent $\kappa+1$ will not envy any agent $i \in [\kappa]$ after removing chore $c_{m+1} \in A^1_{\kappa+1}$.

Next, we consider $\mathcal{A}^2$. For each agent $i \in [\kappa]$, $v(A^2_i) = 2W'$, and $v(A^2_{\kappa+1}) = 4W'$. We verify that $\mathcal{A}^2$ is TEF1: again, each pair of agents $i, j \in [\kappa]$ has the same bundle value, and each agent $i \in [\kappa]$ has a higher bundle value than agent $\kappa + 1$. Also, agent $\kappa + 1$ will not envy any agent $i \in [\kappa]$ after removing chore $c_{m+1} \in A^2_{\kappa+1}$.

For the 'only if' direction, suppose we have a repetitive allocation $\mathcal{A}^2$ which satisfies TEF1. Since agents have identical valuation functions, without loss of generality, suppose that agent $\kappa+1$ receives chore c_{m+1} under $\mathcal{A}^1$. Then, $v(A^2_{\kappa+1} \setminus \{c_{m+1}\}) \leq 2W'$. In order for $\mathcal{A}^2$ to be TEF1, we must have that $v(A^2_i) \leq 2W'$ for each $i \in [\kappa]$ (so that agent $\kappa + 1$ will not envy any agent $i \in [\kappa]$). This means that for each $i \in [\kappa]$, $v(A^1_i) \leq W'$, but since $\sum_{j \in [m]} s'_j = \kappa W'$, this is only possible if there is a κ-way partition of S' such that each subset has a sum of W' (i.e. there is a κ-way partition of S such that each subset has a sum of W). $\square$

4.6 Conclusion

In this chapter, we studied the informed online fair division of indivisible items, with the goal of achieving TEF1 allocations. For both goods and chores, we showed the existence of TEF1 allocations in four special cases and provided polynomial-time algorithms for each case. Additionally, we showed that determining whether a TEF1 allocation exists for goods is NP-hard, and presented a similar, though slightly weaker, intractability result for chores. We further established the incompatibility between TEF1 and PO, which extends to an incompatibility with p-mean welfare. Finally, we explored the special case of multiple items arriving at each round.

Numerous potential directions remain for future work, including revisiting variants of the standard fair division model. An example include studying the existence (and polynomial-time computability) of allocations satisfying a temporal variant of the weaker PROP1 property (see Chapter 2, which would be implied by EF1. It would also be interesting to extend our results, which hold for the cases of goods and chores separately, to the more general case of mixed manna (see, e.g., Aziz et al. [2019]). In fact, with an appropriate modification of the instance, we can extend Theorem 4.3.2 to show that a TEF1 allocation exists in the mixed manna setting when there are two agents (see Section A.2) in the appendix.

Chapter 5

Characterizing Maximum Nash Welfare for Binary Valuations

5.1 Overview

In the previous chapter, we examined the (in)compatibility between TEF1 and efficiency, specifically, Pareto-optimality (PO). This tension mirrors a well-known and extensively studied interplay between fairness and efficiency in the classical *offline* fair division setting, particularly between EF1 and PO. Perhaps surprisingly, this combination is quite elusive: even allocations that maximize *utilitarian* or *egalitarian* welfare do not, in general, satisfy both EF1 and PO.

A breakthrough result in the offline setting by Caragiannis et al. [2019] showed that under additive valuations, it is always possible to find an allocation that satisfies both EF1 and PO by selecting one that maximizes *Nash welfare*—or the product of the agents' utilities. The remarkable fairness and efficiency of the *maximum Nash welfare (MNW)* rule led the authors to refer to it as the "ultimate solution" for the division of indivisible goods under additive valuations.

Building on this, Yuen and Suksompong [2023] provided a characterization of MNW as the unique rule within the class of *welfarist rules*—rules that select allocations by maximizing some function of the agents' utilities—that guarantees EF1.[1]

Given the strong guarantees offered by MNW in the offline setting, it is natural that researchers have sought analogous results in the online setting. The challenge of achieving both EF1 and PO has become a central question here as well.[2] In

[1]This generalizes an earlier result by Suksompong [2023], who proved a similar characterization within the narrower class of *additive welfarist rules*, where the welfare function is additive across agents. For MNW, this function is the sum of the logarithms of each agent's utility.

[2]Note that we now shift our focus back to the classical EF1 property (which is more widely-studied), not TEF1, which is a strictly stronger requirement that has gained recent attention, and

online settings, ensuring both fairness and efficiency is particularly important—and especially challenging.

Benadè et al. [2024] study this trade-off systematically, exploring the extent to which approximations of EF and PO can be achieved under a range of information settings, from identical agents and i.i.d. valuations to complete uncertainty about future arrivals. Meanwhile, Banerjee et al. [2022] establish a strong impossibility result: even sublinear competitive ratios for Nash welfare are unachievable in the worst case. To mitigate this pessimism, they propose augmenting online algorithms with predictive information and analyze the performance improvements this can bring. Earlier work by Azar et al. [2010] considered an online resource allocation problem where their proposed algorithms were competitive with respect to Nash welfare. However, they required a structural assumption: that each agent's valuations for different items are bounded within a fixed ratio. Under this assumption, they achieve a competitive ratio that is logarithmic in the number of agents, items, and the bounded ratio. Beyond the online arrival model, the EF1+PO combination has also been investigated in other variants of online fair division. For example, Igarashi et al. [2024] study fairness-efficiency trade-offs in repeated allocation scenarios, highlighting the nuanced difficulties of maintaining EF1 and PO over time even in this restricted case.

In this chapter, we aim to strengthen the case for using MNW in the offline fair division setting by providing the first axiomatic characterization of MNW within the space of *all* allocation rules. While our results focus on the offline setting—where all items and agent preferences are known in advance—it is worth noting that this includes the online setting with full information, as discussed in Chapter 3, under the standard (non-cumulative) notions considered there. Although our model differs from the more dynamic and information-constrained setting studied in Chapter 4, our characterization serves as a crucial conceptual foundation for understanding fairness and efficiency in online fair division. In particular, it highlights the axiomatic strengths of MNW, which can guide the design and evaluation of algorithms in the more complex online setting.

Despite its strong normative appeal, MNW also comes with known limitations. For instance, it fails to be strategyproof, meaning agents may benefit from misreporting their preferences [Klaus and Miyagawa, 2002, Halpern et al., 2020]. It also violates resource monotonicity, so adding more goods can paradoxically make an agent worse off [Chakraborty et al., 2021b]. Moreover, computing or even approximating MNW is computationally hard in general [Lee, 2017]. Additionally, since EF1 and PO are

often is studied with full information about the future.

properties defined at the level of individual instances, many rules besides MNW can satisfy them—by, for example, selecting a different EF1 and PO allocation in each instance. These observations suggest that characterizing MNW within the space of all rules is a nontrivial and important task.

An important subclass of additive valuations is the class of *binary* valuations (sometimes referred to as *binary additive* valuations), where each agent's utility for each good is either 0 or 1. Binary valuations can be viewed as approval votes, which have long been studied in the voting literature [Brams and Fishburn, 2007, Kilgour, 2010], and permit very simple elicitation. These valuations have therefore been investigated in numerous fair division papers [Aleksandrov et al., 2015, Darmann and Schauer, 2015, Bouveret and Lemaître, 2016, Barman et al., 2018b, Freeman et al., 2019, Halpern et al., 2020, Hosseini et al., 2020, Kyropoulou et al., 2020, Amanatidis et al., 2021, Suksompong and Teh, 2022].[3] Under binary valuations, the aforementioned drawbacks of MNW disappear: MNW is strategyproof [Bogomolnaia and Moulin, 2004, Halpern et al., 2020], resource-monotone [Suksompong and Teh, 2022], and can be computed in polynomial time [Darmann and Schauer, 2015, Barman et al., 2018b]. Moreover, it coincides with the leximin rule as well as additive welfarist rules with any strictly concave function [Benabbou et al., 2021].

In light of these advantages, one may think that MNW is the indisputable choice when valuations are binary. However, it remains conceivable that there exist other rules which satisfy the same attractive properties (and potentially more). To see why the choice of rule is not obvious even under binary valuations, consider the following example.

Example 5.1.1. Consider two agents $N = \{1, 2\}$ and eight goods $G = \{g_1, \ldots, g_8\}$, where agent 1 approves $g_1, \ldots, g_4$ and agent 2 approves $g_3, \ldots, g_8$. MNW gives $g_1, \ldots, g_4$ to agent 1 and $g_5, \ldots, g_8$ to agent 2. However, the allocation that gives g_1, g_2 to agent 1 and $g_3, \ldots, g_8$ to agent 2 is also envy-free (and therefore EF1) and PO.[4] Moreover, since properties like strategyproofness and resource-monotonicity connect different instances, it is unclear whether they can be used to rule out the latter allocation in this instance.

[3]Binary preferences have also been studied in matching [Bogomolnaia and Moulin, 2004] and auctions [Malik and Mishra, 2021].

[4]Note that under binary valuations, every allocation satisfying PO maximizes utilitarian welfare (i.e., the sum of all agents' utilities), since such an allocation must assign each good to an agent who values it. Hence, utilitarian welfare maximization cannot be used to distinguish among rules that ensure PO.

In Section 5.3, we present a characterization of MNW with respect to five axioms: EF1, strategyproofness, neutrality, minimal completeness, and invariance under disapproving unassigned goods (IDU). Neutrality is a basic property that the agents' utilities should not change if the goods are simply relabeled. Minimal completeness stipulates that all goods positively valued by at least one agent should be allocated and the remaining goods unallocated; a minimally complete version of MNW has been shown to satisfy attractive properties [Halpern et al., 2020, Suksompong and Teh, 2022].[5] IDU states that if an agent no longer approves a good that is not allocated to her, the returned allocation should remain the same. Our characterization holds for any number of agents. We also show that all five properties are necessary—omitting any of them leads to a rule different from MNW that satisfies the remaining four properties. Furthermore, we give a tie-breaking refinement of our characterization for the case of two agents by showing that, in addition to maximizing Nash welfare, the rule must break ties in a consistent manner across all instances with the same total number of goods and the same number of goods approved by at least one agent.

Next, in Section 5.4, we provide an alternative characterization for the special case of two agents, by replacing IDU with non-redundancy (i.e., every allocated good should be given to an agent who values it) and resource-monotonicity. In other words, we show that any rule satisfying EF1, strategyproofness, neutrality, minimal completeness, non-redundancy, and resource-monotonicity must maximize Nash welfare. Again, we establish the independence among axioms as well as a tie-breaking refinement of our characterization.

To the best of our knowledge, these are the first characterizations of MNW within the space of all rules for any domain. Together with work on the guarantees afforded by MNW under binary valuations, our results help cement MNW as the suitable allocation rule for this domain.

5.2 Preliminaries

In this chapter, we assume that v_i is *binary additive*, meaning that $v_i(\{g\}) \in \{0,1\}$ for each $g \in G$ and $v_i(S) = \sum_{g \in S} v_i(\{g\})$ for any bundle $S \subseteq G$. We say that a good g is *unvalued* if $v_i(g) = 0$ for all agents $i \in N$, and *valued* otherwise. We define an *approval profile* as the tuple $\mathcal{P} = (P_1, \ldots, P_n)$ such that for each $i \in N$, we have

[5] On the other hand, if all goods must be allocated, no version of MNW can be strategyproof [Halpern et al., 2020].

$g \in P_i$ if and only if $v_i(g) = 1$. An *instance* $\mathcal{I} = (N, G, \mathcal{P})$ is defined by the set of agents N, the set of goods G, and the approval profile $\mathcal{P}$.

Note that in Section 5.4, G is not necessarily fixed, as extra goods may be added. Also, for the purposes of a characterization, we assume that it is not necessary that $\bigcup_{i \in N} A_i = G$. For each n, an *allocation rule* for n agents is a function $\mathcal{F}$ that maps each instance with n agents (and any set of goods) to an allocation.

We consider the fairness notions of EF1 and PO as defined in Chapter 2.

We also consider a related property called *non-redundancy*, which states that each allocated good must be given to an agent who values it. This property—sometimes referred to as *clean* or *non-wasteful*—has been used in several papers [Babaioff et al., 2021, Benabbou et al., 2021, Barman and Verma, 2022, Suksompong and Teh, 2023, Viswanathan and Zick, 2023, Goko et al., 2024, Montanari et al., 2024].

Definition 5.2.1 (Non-redundancy)**.** An allocation $\mathcal{A}$ is *non-redundant* if $v_i(g) = 1$ for all $i \in N$ and $g \in \bigcup_{i \in N} A_i$.

A slightly different property is minimal completeness [Halpern et al., 2020, Suksompong and Teh, 2022].

Definition 5.2.2 (Minimal completeness)**.** An allocation $\mathcal{A}$ is *minimally complete* if it allocates all valued goods and no unvalued goods.

One can check that while none of PO, non-redundancy, and minimal completeness implies another, the combination of any two implies the third.

We say that an allocation rule $\mathcal{F}$ is EF1 (resp., PO, non-redundant, minimally complete) if for all instances $\mathcal{I}$, the allocation $\mathcal{A} = \mathcal{F}(\mathcal{I})$ is EF1 (resp., PO, non-redundant, minimally complete).

Another important property of allocation rules is *strategyproofness*, which stipulates that no agent should be able to misreport her preference so as to obtain a strictly higher utility for herself.

Definition 5.2.3 (Strategyproofness)**.** An allocation rule $\mathcal{F}$ is *strategyproof* if there do not exist valuation profiles $\mathbf{v}$ and $\mathbf{v}'$ with the following property: If we denote the instances corresponding to $\mathbf{v}$ and $\mathbf{v}'$ by $\mathcal{I}$ and $\mathcal{I}'$, respectively, then there exists $i \in N$ such that

- $v_j = v'_j$ for all $j \in N \setminus \{i\}$;
- $v_i(A'_i) > v_i(A_i)$,

where $\mathcal{A} = \mathcal{F}(\mathcal{I})$ and $\mathcal{A}' = \mathcal{F}(\mathcal{I}')$.

The next property, *neutrality*, captures a different type of fairness in that the agents' utilities should not depend on the labeling of the goods.

Definition 5.2.4 (Neutrality)**.** An allocation rule $\mathcal{F}$ is *neutral* if the following holds: For any two instances $\mathcal{I}$ and $\mathcal{I}'$ such that $\mathcal{I}'$ can be obtained from $\mathcal{I}$ by permuting the labels of the goods, if $\mathcal{F}(\mathcal{I}) = \mathcal{A}$ and $\mathcal{F}(\mathcal{I}') = \mathcal{A}'$, then $v_i(A_i) = v_i(A'_i)$ for each $i \in N$.

Note that the companion property of *anonymity*, which asserts that the agents' utilities should not depend on the labeling of the *agents*, cannot be satisfied along with minimal completeness or PO. Indeed, if there are two agents and one good that is valued by both agents, then one agent must receive utility 1 and the other agent 0.

We now introduce a new property that we call *invariance under disapproving unassigned goods (IDU)*. This property states that if an agent no longer approves a good that is not allocated to her, then the allocation returned by the rule should remain the same. IDU is similar in spirit to *independence of irrelevant alternatives*, which is frequently considered in social choice settings [Arrow, 1950].

Definition 5.2.5 (Invariance under disapproving unassigned goods)**.** An allocation rule $\mathcal{F}$ is *invariant under disapproving unassigned goods (IDU)* if the following holds: For any two instances $\mathcal{I} = (N, G, \mathcal{P})$ and $\mathcal{I}' = (N, G, \mathcal{P}')$ such that $\mathcal{I}$ differs from $\mathcal{I}'$ in that $g \in P_i$ and $g \notin P'_i$ for some good g, if $\mathcal{F}(\mathcal{I}) = \mathcal{A}$ and $g \notin A_i$, then $\mathcal{F}(\mathcal{I}') = \mathcal{F}(\mathcal{I})$.

We observe that IDU and minimal completeness together imply PO, a fact that will be useful later.

Lemma 5.2.6. *Any allocation rule (for any n) satisfying IDU and minimal completeness also satisfies PO.*

Proof. Consider a rule $\mathcal{F}$ satisfying IDU and minimal completeness, and take any instance. Minimal completeness implies that all valued goods are allocated and all unvalued goods are unallocated. Suppose for contradiction that some valued good g is allocated to an agent i who does not value it. Consider a modified instance where all other agents do not value g. By IDU, g should still be allocated to i. However, this contradicts minimal completeness, as g is now unvalued. Hence, every valued good must be allocated to an agent who values it, and PO readily follows. $\square$

One can check that the converse of Lemma 5.2.6 does not hold—PO implies neither minimal completeness nor IDU—and minimal completeness or IDU alone does not imply PO.

We now define the main rule of interest in this chapter.

Definition 5.2.7 (Maximum Nash welfare)**.** Given an instance, an allocation $\mathcal{A}$ is a *maximum Nash welfare (MNW) allocation* if, among the set of allocations in $\Pi_n(G)$, it maximizes the number of agents receiving positive utility and, subject to that, maximizes the product of positive utilities. Formally, let $H(\mathcal{A}) = \{i \in N : v_i(A_i) > 0\}$ and $\mathcal{H} = \arg\max_{\mathcal{A}\in\Pi_n(G)} |H(\mathcal{A})|$. Then, for any instance $\mathcal{I}$, MNW($\mathcal{I}$) $= \arg\max_{\mathcal{A}'\in\mathcal{H}} \prod_{i\in H(\mathcal{A}')} v_i(A'_i)$ denotes the set of all MNW allocations in $\mathcal{I}$, and $\mathcal{A}$ is an MNW allocation if $\mathcal{A} \in$ MNW($\mathcal{I}$).

We say that a rule *maximizes Nash welfare* (equivalently, is an MNW rule) if, given any instance, it always returns an MNW allocation. If there is more than one MNW allocation, the rule is allowed to break ties arbitrarily.

Halpern et al. [2020] proposed MNW$^{\text{tie}}$, which breaks ties for MNW by discarding unvalued goods and maximizing the agents' utilities lexicographically. If we further break any remaining ties according to a lexicographic order over the goods, MNW$^{\text{tie}}$ satisfies EF1, PO, non-redundancy, minimal completeness, strategyproofness, neutrality, and IDU.

5.3 Main Characterization

This section details our main result, which characterizes MNW as the only rule satisfying EF1, strategyproofness, neutrality, minimal completeness, and IDU under binary valuations. We show that this characterization holds for any number of agents $n \geq 2$.

We first state a lemma of Halpern et al. [2020], which provides necessary and sufficient conditions for an allocation to be MNW.[6] A *path* is a sequence of agents together with an allocation such that each agent (except the first) values some good in her predecessor's bundle, and the path is critical if the first agent values her own bundle at least a utility of two more than the last agent values his own bundle. More precisely, given an instance $\mathcal{I} = (N, G, \mathcal{P})$ and an allocation $\mathcal{A}$, a path in the pair $(\mathcal{I}, \mathcal{A})$ is a sequence of agents $(i_1, \ldots, i_k)$ such that $A_{i_\ell} \cap P_{i_{\ell+1}} \neq \emptyset$ for all $\ell \in \{1, \ldots, k-1\}$. A path $(i_1, \ldots, i_k)$ in $(\mathcal{I}, \mathcal{A})$ is *critical* if $v_{i_1}(A_{i_1}) > v_{i_k}(A_{i_k}) + 1$.

[6] This is Lemma 5 in the extended version of their work [Halpern et al., 2020].

Lemma 5.3.1 (Halpern et al. [2020]). *For any* $(\mathcal{I}, \mathcal{A})$ *such that* $\mathcal{A}$ *is PO,* $\mathcal{A} \in \mathrm{MNW}(\mathcal{I})$ *if and only if there is no critical path in* $(\mathcal{I}, \mathcal{A})$.

Then, our characterization result is as follows.

Theorem 5.3.2. *For any fixed* $n \geq 2$*, under binary valuations, any allocation rule that is* EF1*, strategyproof, neutral, minimally complete, and* IDU *maximizes Nash welfare.*

Proof. Let $\mathcal{F}$ be a rule satisfying the five axioms in the theorem statement. Since $\mathcal{F}$ is minimally complete and IDU, by Lemma 5.2.6, we can assume that $\mathcal{F}$ is PO.

Suppose for contradiction that $\mathcal{F}$ does not maximize Nash welfare. By Lemma 5.3.1 and the assumption that $\mathcal{F}$ satisfies PO, there must exist an instance $\mathcal{I}$ such that $(\mathcal{I}, \mathcal{F}(\mathcal{I}))$ admits a critical path. By PO, for an allocation $\mathcal{A}$ returned by $\mathcal{F}$ in any instance, we have $v_i(A_i) = |A_i|$ for all $i \in N$.

Let k be the length of a shortest critical path across all instances, and let $\mathcal{I} = (N, G, \mathcal{P})$ and $\mathcal{A} = \mathcal{F}(\mathcal{I})$ be such an instance and its corresponding allocation under $\mathcal{F}$. Then, we have that $(\mathcal{I}, \mathcal{A})$ admits a critical path $(i_1, \ldots, i_k)$ of length k. Without loss of generality, let $i_\ell = \ell$ for all $\ell \in [k]$.

Since $(1, \ldots, k)$ is a critical path, by definition of a critical path, it holds that

$$|A_1| > |A_k| + 1. \tag{5.1}$$

We split our analysis into two cases.

Case 1: $k \geq 3$**.** By minimal completeness and IDU, we may assume that

(i) for all $i \in [k]$ and $j \in N \setminus [k]$, $P_i \cap A_j = \emptyset$ and $P_j \cap A_i = \emptyset$;

(ii) for all $\ell \in [k] \setminus \{1\}$, $|P_\ell| = |A_\ell| + 1$ and $|P_\ell \cap A_{\ell-1}| = 1$; and

(iii) $P_1 = A_1$.

Indeed, for (i) and (iii), we may let, e.g., each agent $i \in [k]$ disapprove all goods in the bundle A_j for every $j \in N \setminus [k]$; the allocation output by $\mathcal{F}$ does not change due to IDU. Similarly, for (ii), we let each agent $\ell \in [k] \setminus \{1\}$ stop approving all goods in $A_{\ell-1}$ except one; the existence of the latter good follows from the definition of a path. For $\ell \in [k] \setminus \{1\}$, let g_ℓ be the unique good in $P_\ell \cap A_{\ell-1}$.

If $|A_1| - 1 > |A_\ell|$ for some $\ell \in [k-1] \setminus \{1\}$, then $(1, \ldots, \ell)$ is a critical path shorter than $(1, \ldots, k)$, a contradiction. On the other hand, if $|A_1| - 1 < |A_\ell|$, then since

$\mathcal{P}$	$A_1{=}P_1$		$A_2{=}P_2\setminus\{g_2\}$				$A_{k-2}{=}P_{k-2}\setminus\{g_{k-2}\}$		$A_{k-1}{=}P_{k-1}\setminus\{g_{k-1}\}$		$A_k{=}P_k\setminus\{g_k\}$	bundle
		g_2		g_3	$\cdots$	g_{k-2}		g_{k-1}		g_k		size/value
1	$1 \cdots$	1										$r+1$
2		1	$1 \cdots$	1								r
$\vdots$					$\ddots$							$\vdots$
$k-2$						1	$1 \cdots$	1				r
$k-1$								1	$1 \cdots$	1		r
k										1	$1 \cdots 1$	$r-1$

Table 5.1: Profile $\mathcal{P}$ for instance $\mathcal{I}$ in Case 1 of the proof of Theorem 5.3.2. Empty cells indicate a value of 0.

$|A_1| > |A_k| + 1$ (by definition of a critical path), we get $|A_\ell| \geq |A_1| > |A_k| + 1$, which means that $|A_\ell| - 1 > |A_k|$ and so $(\ell, \ldots, k)$ is a critical path shorter than $(1, \ldots, k)$, again a contradiction. Thus, we must have

$$|A_1| - 1 = |A_\ell| \quad \text{for all} \quad \ell \in [k-1] \setminus \{1\}. \tag{5.2}$$

Now, if $|A_k| \geq |A_\ell|$ for some $\ell \in [k-1] \setminus \{1\}$, then we get $|A_1| - 1 = |A_\ell| \leq |A_k|$, contradicting (5.1). Therefore, it must be that $|A_k| < |A_\ell|$, or equivalently,

$$|A_k| + 1 \leq |A_\ell| \quad \text{for all} \quad \ell \in [k-1] \setminus \{1\}.$$

Then, if $|A_k| + 1 < |A_\ell|$ for some $\ell \in [k-1] \setminus \{1\}$, we again have that $(\ell, \ldots, k)$ is a critical path shorter than $(1, \ldots, k)$, a contradiction. Hence,

$$|A_k| + 1 = |A_\ell| \quad \text{for all} \quad \ell \in [k-1] \setminus \{1\}.$$

Combining this with (5.2), we get

$$|A_1| - 1 = |A_2| = \cdots = |A_{k-1}| = |A_k| + 1 := r. \tag{5.3}$$

The valuations and corresponding bundles of agents in the critical path of $(\mathcal{I}, \mathcal{A})$ are illustrated in Table 5.1.

We will show that k cannot be the length of a shortest critical path across all profiles, by constructing a profile with a critical path of length strictly less than k, thereby arriving at a contradiction.

Consider another instance $\mathcal{I}' = (N, G, \mathcal{P}')$ with profile $\mathcal{P}' = (P'_1, \ldots, P'_n)$ such that

$$P'_{k-2} = P_{k-2} \cup \{g_k\} \quad \text{and} \quad P'_\ell = P_\ell \text{ for all } \ell \in N \setminus \{k-2\}.$$

Let $\mathcal{A}' = \mathcal{F}(\mathcal{I}')$ be the corresponding allocation returned by $\mathcal{F}$ on this instance $\mathcal{I}'$. The valuations of agents $\{k-2, k-1, k\}$ in the critical path of $(\mathcal{I}, \mathcal{A})$ under the new instance $\mathcal{I}'$ are illustrated as follows. Empty cells indicate that the agent has a value of 0 for the good, and the valuations of agents $\{1, \dots, k-3\}$ remain the same as in $\mathcal{P}$.

$\mathcal{P}'$	$\cdots$	g_{k-2}			g_{k-1}			g_k			
$\vdots$	$\ddots$										
$k-2$		1	1	$\cdots$	1			**1**			
$k-1$					1	1	$\cdots$	1			
k								1	1	$\cdots$	1

We consider two further cases, based on whether g_k is contained in A'_{k-2}.

Case 1(a): $g_k \in A'_{k-2}$**.** We know that $P'_k = P_k = A_k \cup \{g_k\}$ and agent k is the only agent who values goods in A_k (by (i)). Hence, by PO, we have that $|A'_k| = |A_k| = r-1$.

If $g_{k-2} \in A'_{k-2}$ (when $k \geq 4$) or $g_{k-1} \in A'_{k-2}$, then since agent $k-2$ is the only agent who values goods in $A_{k-2} \setminus \{g_{k-1}\}$, by PO, we have that $|A'_{k-2}| \geq |A_{k-2} \cup \{g_k\} \setminus \{g_{k-1}\}| + 1 \geq r+1$. This means that $(k-2, k)$ forms a critical path of length 2, contradicting the assumption that $k \geq 3$ is the length of a shortest critical path across all profiles. Thus, we must have that $g_{k-2} \notin A'_{k-2}$ (when $k \geq 4$) and $g_{k-1} \notin A'_{k-2}$. If $k = 3$, then $(k-2, k)$ forms a critical path of length 2 regardless, so it must hold that $k \geq 4$. Then, we have $|A'_{k-2}| = r$.

Recall that $|A_1| = r+1$ and $|A_\ell| = r$ for all $\ell \in \{2, \dots, k-3\}$. Since $g_{k-2} \notin A'_{k-2}$, this means that $|A'_1 \cup \cdots \cup A'_{k-3}| = r + 1 + (k-4) \cdot r = (k-3) \cdot r + 1$. Thus, by the pigeonhole principle, there must exist an agent $\ell \in \{1, \dots, k-3\}$ such that $|A'_\ell| = r+1$ (by PO, we cannot have $|A'_\ell| > r+1$). Let ℓ be the largest such index in $\{1, \dots, k-3\}$.

If there exists an agent $\ell' \in \{\ell+1, \dots, k-3\}$ with $|A'_{\ell'}| = r-1$ (by PO, we cannot have $|A'_{\ell'}| < r-1$), then let ℓ' be the smallest such index in $\{\ell+1, \dots, k-3\}$. Consequently, there exists a critical path $(\ell, \ell+1, \dots, \ell')$ with length strictly less than k, contradicting the assumption that k is the length of a shortest critical path across all profiles. Otherwise, all agents $\ell' \in \{\ell+1, \dots, k-3\}$ has $|A'_{\ell'}| = r$. Then, together with the fact that $|A'_{k-2}| = r$, there exists a critical path $(\ell, \ell+1, \dots, k-2, k)$ with length strictly less than k, again giving us a contradiction.

Case 1(b): $g_k \notin A'_{k-2}$**.** IDU implies that

$$\mathcal{A} = \mathcal{A}'. \tag{5.4}$$

Consider another instance $\mathcal{I}'' = (N, G, \mathcal{P}'')$ with profile $\mathcal{P}'' = (P''_1, \dots, P''_n)$ such that

$$P''_k = P'_k \cup \{g_{k-1}\} \quad \text{and} \quad P''_\ell = P'_\ell \text{ for all } \ell \in N \setminus \{k\}.$$

Let $\mathcal{A}'' = \mathcal{F}(\mathcal{I}'')$ be the corresponding allocation returned by $\mathcal{F}$ on this instance $\mathcal{I}''$. The valuations of agents $\{k-2, k-1, k\}$ in the critical path of $(\mathcal{I}, \mathcal{A})$ under the new instance $\mathcal{I}''$ are illustrated as follows. The valuations of agents $\{1, \ldots, k-3\}$ remain the same as in $\mathcal{P}'$ (and $\mathcal{P}$).

$\mathcal{P}''$	$\cdots$	g_{k-2}			g_{k-1}			g_k			
$\vdots$	$\ddots$										
$k-2$		1	1	$\cdots$	1			1			
$k-1$					1	1	$\cdots$	1			
k					**1**			1	1	$\cdots$	1

Note that under instance $\mathcal{I}''$, g_{k-1} and g_k are approved by the same set of agents (namely, $\{k-2, k-1, k\}$).

Strategyproofness for agent k at profile $\mathcal{P}'$ implies that

$$|A''_k \cap P'_k| \leq r - 1; \tag{5.5}$$

otherwise agent k can manipulate from P'_k to P''_k and strictly benefit. Moreover, since agent k is the only agent who approves goods in $A_k = A'_k = P'_k \setminus \{g_k\}$, by PO and (5.5), $A''_k \cap P'_k = A'_k$ (i.e., $g_k \notin A''_k$).

Suppose for contradiction that $A''_k \neq A'_k$. Then, by PO,

$$A''_k = A'_k \cup \{g_{k-1}\}. \tag{5.6}$$

Let π be the permutation of G swapping g_{k-1} and g_k and fixing all other goods. Consider the instance $\mathcal{I}''' = (N, G, \mathcal{P}''')$ with profile $\mathcal{P}''' = \pi(\mathcal{P}')$, and let $\mathcal{A}''' = \mathcal{F}(\mathcal{I}''')$ be the corresponding allocation returned by $\mathcal{F}$ on the instance $\mathcal{I}'''$. The valuations of agents $\{k-2, k-1, k\}$ in the critical path of $(\mathcal{I}, \mathcal{A})$ under the new instance $\mathcal{I}'''$ are illustrated as follows. The valuations of agents $\{1, \ldots, k-3\}$ remain the same as in $\mathcal{P}'$ (and $\mathcal{P}$).

$\mathcal{P}'''$	$\cdots$	g_{k-2}			g_{k-1}			g_k			
$\vdots$	$\ddots$										
$k-2$		1	1	$\cdots$	1			1			
$k-1$					1	1	$\cdots$	1			
k					1				1	$\cdots$	1

Note that $\mathcal{P}'''$ can be derived from $\mathcal{P}''$ by replacing P''_k with $P''_k \setminus \{g_k\}$.

Now, we have that

$$A''_k = A'_k \cup \{g_{k-1}\} = A_k \cup \{g_{k-1}\}, \tag{5.7}$$

where the equalities follow from (5.6) and (5.4), respectively. Since $g_k \notin A_k$, it holds that $g_k \notin A''_k$. Then, by IDU between $\mathcal{P}''$ and $\mathcal{P}'''$, we have $\mathcal{A}''' = \mathcal{A}''$. Combining this with (5.7), we get that

$$A'''_k = A'_k \cup \{g_{k-1}\}. \tag{5.8}$$

On the other hand, since $\mathcal{P}''' = \pi(\mathcal{P}')$, neutrality implies that $|A'''_k| = |A'_k|$, contradicting (5.8). Hence, $A''_k = A'_k$.

By IDU between $\mathcal{P}''$ and $\mathcal{P}'$, we have

$$\mathcal{A}'' = \mathcal{A}'. \tag{5.9}$$

Since $g_{k-1} \in A_{k-2}$ and $\mathcal{A} = \mathcal{A}''$ (by (5.4) and (5.9)), it follows that $g_{k-1} \in A''_{k-2}$. Moreover, since $g_{k-1} \in P''_k$, the path $(1, \ldots, k-2, k)$ is a critical path of length $k-1$ in $(\mathcal{I}'', \mathcal{A}'')$. This contradicts the assumption that k is the length of a shortest critical path across all profiles.

Case 2: $k = 2$**.** From (5.1), we have

$$|A_1| \geq |A_2| + 2. \tag{5.10}$$

By minimal completeness and IDU, we may assume that

(i) for all $i \in \{1, 2\}$ and $j \in N \setminus \{1, 2\}$, $P_i \cap A_j = \emptyset$ and $P_j \cap A_i = \emptyset$;

(ii) $|P_2 \cap A_1| = 1$; and

(iii) $P_1 = A_1$.

The valuations and bundles of agents $\{1, 2\}$ in the critical path of $(\mathcal{I}, \mathcal{A})$ are illustrated as follows.

$$\begin{array}{c|cccc|ccc}
\mathcal{P} & \multicolumn{4}{c}{\overbrace{\qquad\qquad\qquad\qquad}^{A_1=P_1}} & \multicolumn{3}{c}{\overbrace{\qquad\qquad\qquad}^{A_2}} \\
\hline
1 & 1 & \cdots & 1 & 1 & & & \\
2 & & & & 1 & 1 & \cdots & 1 \\
 & & & & \underbrace{\quad}_{P_2 \cap A_1} & & &
\end{array}$$

For $\alpha \in \{0, \ldots, |A_2| + 1\}$, let S_α be some set of α goods in $A_1 \setminus P_2$; such a set exists by (ii) and (5.10). Consider any instance $\mathcal{I}^\alpha = (N, G, \mathcal{P}^\alpha)$ with profile $\mathcal{P}^\alpha = (P_1^\alpha, P_2^\alpha)$ such that $P_1^\alpha = P_1$ and $P_2^\alpha = P_2 \cup S_\alpha$. Let $\mathcal{A}^\alpha = \mathcal{F}(\mathcal{I}^\alpha)$ be the corresponding allocation returned by $\mathcal{F}$ on the instance $\mathcal{I}^\alpha$. The valuations of both agents under $\mathcal{I}^\alpha$ are illustrated as follows. We also include the allocation $\mathcal{A}$ and the set S_α; note that $\mathcal{A}$ does *not* denote the allocation returned by $\mathcal{F}$ in this instance.

$$\begin{array}{c|l}
\mathcal{P}^{\alpha} & \\ \hline
1 & \overbrace{\boxed{1 \cdots 1 \quad 1 \cdots 1 \quad 1}}^{A_1} \\
2 & \quad \underbrace{\mathbf{1} \cdots \mathbf{1}}_{S_\alpha} \quad \underbrace{1}_{P_2 \cap A_1} \quad \overbrace{\boxed{1 \cdots 1}}^{A_2}
\end{array}$$

Since agent 2 is the only agent who values goods in A_2, by PO, we have $A_2 \subseteq A_2^\alpha$. We will prove by induction that $|A_2^\alpha| = |A_2|$ for all $\alpha \in [|A_2| + 1]$. The base case $\alpha = 0$ holds trivially. We establish the induction step via the following lemma.

Lemma 5.3.3. *Suppose that for any instance $\mathcal{I}^\kappa = (N, G, \mathcal{P}^\kappa)$ with S_κ for some $\kappa \in [|A_2|]$, we have $|A_2^\kappa| = |A_2|$. Then, for any instance $\mathcal{I}^{\kappa+1} = (N, G, \mathcal{P}^{\kappa+1})$ with $S_{\kappa+1}$, we also have $|A_2^{\kappa+1}| = |A_2|$.*

Proof. Assume that $|A_2^\kappa| = |A_2|$ for any instance $\mathcal{I}^\kappa$ with S_κ. Consider the instance $\mathcal{I}^{\kappa+1} = (N, G, \mathcal{P}^{\kappa+1})$ such that $P_1^{\kappa+1} = P_1^\kappa$ and $P_2^{\kappa+1} = P_2^\kappa \cup \{g\}$ for some $g \in A_1^\kappa \setminus P_2^\kappa$. Such a good g exists since $|A_1^\kappa| \geq |A_2| + 2$ by the induction hypothesis and $A_1^\kappa \subseteq P_1$, while $|P_2^\kappa \cap P_1| \leq |A_2| + 1$.

Let $\mathcal{A}^{\kappa+1} = \mathcal{F}(\mathcal{I}^{\kappa+1})$ be the corresponding allocation returned by $\mathcal{F}$ on the instance $\mathcal{I}^{\kappa+1}$. If $g^* \in A_2^{\kappa+1}$ for some good $g^* \in P_2^\kappa \cap A_1^\kappa$, then agent 2 can misreport her preferences from $\mathcal{P}^\kappa$ to $\mathcal{P}^{\kappa+1}$ so as to obtain a strictly higher utility, contradicting the strategyproofness of $\mathcal{F}$. Thus, $g^* \notin A_2^{\kappa+1}$ for all $g^* \in P_2^\kappa \cap A_1^\kappa$.

If $g \notin A_2^{\kappa+1}$, then by the previous paragraph, the induction hypothesis, and PO, we have $A_1^{\kappa+1} = A_1^\kappa$. This implies that $|A_2^{\kappa+1}| = |A_2^\kappa| = |A_2|$, where the latter equality follows from the induction hypothesis, as desired. Suppose now that $g \in A_2^{\kappa+1}$. Consider another instance $\widehat{\mathcal{I}} = (N, G, \widehat{\mathcal{P}})$ with $\widehat{\mathcal{P}}$ such that $\widehat{P}_1 = P_1^{\kappa+1} = P_1^\kappa$ and $\widehat{P}_2 = P_2^{\kappa+1} \setminus \{g^*\}$ for some $g^* \in P_2^\kappa \cap A_1^\kappa$. Let $\widehat{\mathcal{A}} = \mathcal{F}(\widehat{\mathcal{I}})$ be the corresponding allocation returned by $\mathcal{F}$ on the instance $\widehat{\mathcal{I}}$. By IDU, $\widehat{\mathcal{A}} = \mathcal{A}^{\kappa+1}$. In particular, we have

$$|\widehat{A}_2| = |A_2^{\kappa+1}|. \tag{5.11}$$

Note that $\widehat{\mathcal{P}} = \pi(\mathcal{P}^\kappa)$ for some permutation π of G. By neutrality, $|\widehat{A}_2| = |A_2^\kappa| = |A_2|$, where the latter equality follows from the induction hypothesis. Combining this with (5.11), we get $|A_2^{\kappa+1}| = |\widehat{A}_2| = |A_2|$, as desired. $\square$

Thus, by induction, we showed that for any instance $\mathcal{I}^\alpha$ where $\alpha = |A_2| + 1$, we have $|A_2^\alpha| = |A_2|$. However, observe that $|P_2^\alpha| = 2(|A_2|+1)$ while $|A_2^\alpha| = |A_2|$, so EF1 is violated. This yields the desired contradiction. $\square$

To complement Theorem 5.3.2, we demonstrate that all five properties are necessary for the characterization—in other words, if we drop any of them, there exists an allocation rule that satisfies the remaining four properties but does not maximize Nash welfare.

Proposition 5.3.4. *Under binary valuations, given only four of the properties in {EF1, strategyproofness, neutrality, minimal completeness, IDU}, there exists an allocation rule for $n = 2$ that satisfies the four properties but does not maximize Nash welfare.*

Proof. We show that for each of the five properties (EF1, minimal completeness, neutrality, IDU, strategyproofness), there exists an allocation rule for $n = 2$ that does not maximize Nash welfare and fails only that property among the five. We include an example demonstrating the violation of each property (and MNW) by the proposed rule.

- **EF1**: A rule that discards all unvalued goods, allocates goods valued by both agents to agent 1, and allocates each of the remaining goods to the agent who uniquely values the good.

  > To see that the rule is not EF1, consider an instance with $m = 2$ goods that are valued by both agents. Then, the rule will allocate both goods to agent 1, violating EF1. The same example shows that the rule returns an allocation that is not MNW.

- **Minimal completeness**: The MNW$^{\text{tie}}$ rule favoring agent 1, with one exception: if $m = 1$, then always discard the good.

  > To see that the rule is not minimally complete, observe that it does not allocate a valued good when $m = 1$ and both agents value the good. This allocation is also not MNW.

- **Neutrality**: The MNW$^{\text{tie}}$ rule favoring agent 1, with the exception that when $m = 2$ and the profile is either of the following $\mathcal{P}_1$ or $\mathcal{P}_2$, it returns the indicated allocation.

$\mathcal{P}_1$	g_1	g_2
1	1	1
2	0	1

and

$\mathcal{P}_2$	g_1	g_2
1	1	1
2	1	1

To see that the rule is not neutral, consider the following profile:

$\mathcal{P}$	g_1	g_2
1	1	1
2	1	0

Then, the $\mathrm{MNW}^{\mathrm{tie}}$ rule favoring agent 1 will return the above indicated allocation, giving both agents 1 and 2 a utility of 1 each. However, $\mathcal{P} = \pi(\mathcal{P}_1)$ for some permutation π of G, and by neutrality the utility that each agent gets in both instances should be equal, which is not the case. The allocation returned by the rule for $\mathcal{P}_1$ is not MNW.

- **IDU**: The $\mathrm{MNW}^{\mathrm{tie}}$ rule favoring agent 2, with the exception that when $m = 1$ and only agent 1 values the good, allocate it to agent 2.

To see that the rule is not IDU, consider the profile where $m = 1$ and the good is unvalued. Then, $\mathrm{MNW}^{\mathrm{tie}}$ favoring agent 2 will discard the good. However, IDU states that the allocation should remain the same as when only agent 1 values the good, and thus the rule does not satisfy it. The allocation returned by the rule when only agent 1 values the good is not MNW.

- **Strategyproofness**: The $\mathrm{MNW}^{\mathrm{tie}}$ rule favoring agent 2, with one exception: when $m = 5$, $|P_1| \geq 4$, $|P_2| = 2$, and all goods are valued, return any allocation (A_1, A_2) with $|A_1| = v_1(A_1) = 4$ and $|A_2| = v_2(A_2) = 1$.

To see that the rule is not strategyproof, consider the following two profiles:

$\mathcal{P}_1$	g_1	g_2	g_3	g_4	g_5
1	0	0	1	1	0
2	0	0	0	1	1

and

$\mathcal{P}_2$	g_1	g_2	g_3	g_4	g_5
1	**1**	**1**	1	1	0
2	0	0	0	1	1

For $\mathcal{P}_1$, the $\mathrm{MNW}^{\mathrm{tie}}$ rule favoring agent 2 will return the above indicated allocation, giving agent 1 a utility of 1. For $\mathcal{P}_2$, the rule will return the indicated allocation, thereby giving agent 1 a utility of 2 with respect to her true preference $\mathcal{P}_1$. This allocation is also not MNW.

$\square$

Now, consider any instance with $n = 2$ agents and m goods. Assume that $m = 2r + 1$ is odd and, for each agent, it is possible to obtain an MNW allocation that gives r goods to that agent and $r + 1$ goods to the other agent. A natural question is whether an allocation rule satisfying the five properties in Theorem 5.3.2 needs consistent tie-breaking, i.e., must consistently break ties in favor of a particular agent across all such instances. We show that among all instances with the same number of total goods and the same number of valued goods, any such rule must indeed break ties consistently.

Let $N = \{1, 2\}$. For an allocation rule $\mathcal{F}$, we say that *tie-breaking is relevant* for an instance $\mathcal{I} = (N, G, \mathcal{P})$ if for $\mathcal{F}(\mathcal{I}) = \mathcal{A}$ where $\mathcal{A} \in \text{MNW}(\mathcal{I})$, either

(i) $|A_1| = r + 1$, $|A_2| = r$, and $A_1 \cap P_2 \neq \emptyset$, or

(ii) $|A_1| = r$, $|A_2| = r + 1$, and $A_2 \cap P_1 \neq \emptyset$,

for some $r \in \mathbb{Z}_{\geq 0}$. If (i) holds, we say that $\mathcal{F}$ favors agent 1, while if (ii) holds, we say that $\mathcal{F}$ favors agent 2. Our formal result is stated below.

Theorem 5.3.5. *Let $n = 2$, and let $\mathcal{F}$ be a rule that is minimally complete, neutral, IDU, and maximizes Nash welfare. Then, for each m and each $m_v \in \{1, \ldots, m\}$, there exists an agent $i \in \{1, 2\}$ such that for every instance with m goods, of which m_v are valued, such that tie-breaking is relevant, $\mathcal{F}$ chooses an MNW allocation favoring agent i.*

Proof. Consider any instance $\mathcal{I} = (N, G, \mathcal{P})$ where tie-breaking is relevant, and assume without loss of generality that $\mathcal{F}(\mathcal{I}) = \mathcal{A}$ is an MNW allocation favoring agent 1. Suppose for contradiction that there exists some other instance $\mathcal{I}' = (N, G, \mathcal{P}')$ with the same number of goods m and valued goods $m_v \in \{1, \ldots, m\}$ where tie-breaking is relevant such that $\mathcal{F}(\mathcal{I}') = \mathcal{A}'$ is an MNW allocation favoring agent 2. In both instances, there is the same number of unvalued goods, which are discarded by the rule (by minimal completeness). Let the number of valued goods be $m_v = 2r + 1$. Then, we have that

$$|A_1| = r + 1, \quad |A_2| = r, \quad |A'_1| = r, \quad |A'_2| = r + 1.$$

Consider the instance $\widehat{\mathcal{I}} = (N, G, \widehat{\mathcal{P}})$ with profile $\widehat{\mathcal{P}} = (\widehat{P}_1, \widehat{P}_2)$ such that

- $\widehat{P}_1 = A_1$, and
- $\widehat{P}_2 = A_2 \cup \{\widehat{g}\}$ for some $\widehat{g} \in A_1 \cap P_2$.

The valuations of agents $N = \{1, 2\}$ under this instance are illustrated as follows. Empty cells indicate that the agent has a value of 0 for the good. Unvalued goods are not allocated and therefore omitted.

$$\begin{array}{c|ccccccc}
 & \multicolumn{4}{c}{\overbrace{\qquad\qquad\qquad}^{A_1}} & \multicolumn{3}{c}{\overbrace{\qquad\qquad}^{A_2}} \\
\widehat{\mathcal{P}} & & & & \widehat{g} & & & \\
\hline
1 & 1 & \cdots & 1 & 1 & & & \\
2 & & & & 1 & 1 & \cdots & 1
\end{array}$$

Let $\widehat{\mathcal{A}} = \mathcal{F}(\widehat{\mathcal{P}})$ be the allocation returned by $\mathcal{F}$ on profile $\widehat{\mathcal{P}}$. Since $\mathcal{F}$ is IDU, we have that $\widehat{\mathcal{A}} = \mathcal{A}$. In particular,

$$|\widehat{A}_1| = r + 1 \quad \text{and} \quad |\widehat{A}_2| = r. \tag{5.12}$$

Similarly, consider the instance $\widehat{\mathcal{I}}' = (N, G, \widehat{\mathcal{P}}')$ with profile $\widehat{\mathcal{P}}' = (\widehat{P}'_1, \ldots, \widehat{P}'_n)$ such that

- $\widehat{P}'_1 = A'_1 \cup \{\widehat{g}'\}$ for some $\widehat{g}' \in A'_2 \cap P'_1$, and
- $\widehat{P}'_2 = A'_2$.

The valuations of agents $N = \{1, 2\}$ under this instance are illustrated as follows.

$$\begin{array}{c|ccccccc}
 & \multicolumn{3}{c}{\overbrace{\qquad\qquad}^{A'_1}} & \multicolumn{4}{c}{\overbrace{\qquad\qquad\qquad}^{A'_2}} \\
\widehat{\mathcal{P}}' & & & & \widehat{g}' & & & \\
\hline
1 & 1 & \cdots & 1 & 1 & & & \\
2 & & & & 1 & 1 & \cdots & 1
\end{array}$$

Let $\widehat{\mathcal{A}}' = \mathcal{F}(\widehat{\mathcal{P}}')$ be the allocation returned by $\mathcal{F}$ on profile $\widehat{\mathcal{P}}'$. Since $\mathcal{F}$ is IDU, we have that $\widehat{\mathcal{A}}' = \mathcal{A}'$. In particular,

$$|\widehat{A}'_1| = r \quad \text{and} \quad |\widehat{A}'_2| = r + 1. \tag{5.13}$$

However, since the number of unvalued goods is the same in both instances, neutrality implies that $|\widehat{A}'_k| = |\widehat{A}_k|$ for all $k \in \{1, 2\}$, a contradiction to (5.12) and (5.13). $\square$

5.4 Alternative Characterization for $n = 2$

In this section, we provide an alternative characterization for the setting of two agents, using non-redundancy along with another property called resource-monotonicity in place of IDU. Resource-monotonicity states that when an extra good is added, no agent's utility should decrease.

Definition 5.4.1 (Resource-monotonicity)**.** An allocation rule $\mathcal{F}$ is *resource-monotone* if the following holds: For any two instances $\mathcal{I}$ and $\mathcal{I}'$ such that $\mathcal{I}'$ can be obtained from $\mathcal{I}$ by adding one extra good, if $\mathcal{F}(\mathcal{I}) = \mathcal{A}$ and $\mathcal{F}(\mathcal{I}') = \mathcal{A}'$, then for each $i \in N$, $v_i(A_i) \leq v_i(A'_i)$.

It was previously shown that the MNW^{tie} rule satisfies resource-monotonicity [Suksompong and Teh, 2022].

For $\{i, j\} = \{1, 2\}$ and a non-redundant bundle A_i, let $A_i = A_i^0 \cup A_i^1$ be the partition of A_i into the set of goods that agent i uniquely values (i.e., $A_i^0 = A_i \setminus P_j$) and the set of goods that both agents value (i.e., $A_i^1 = A_i \cap P_j$), respectively. Our characterization result is as follows.

Theorem 5.4.2. *For $n = 2$, under binary valuations, any allocation rule $\mathcal{F}$ that is* EF1*, non-redundant, strategyproof, resource-monotone, neutral, and minimally complete maximizes Nash welfare.*

Proof. Let $N = \{1, 2\}$, and let $\mathcal{F}$ be a rule satisfying the six axioms in the theorem statement. We show that it maximizes Nash welfare by induction on the number of goods $m = |G|$. By non-redundancy, for an allocation $\mathcal{A}$ returned by $\mathcal{F}$ in any instance, we have $v_i(A_i) = |A_i|$ for all $i \in N$.

For the base case $m = 1$, if the good is unvalued, the rule leaves it unallocated (by minimal completeness). If the good is valued, the rule allocates it to some agent who values it (by minimal completeness and non-redundancy). Hence, the rule maximizes Nash welfare.

We introduce some additional notation. For every instance $\mathcal{I} = (N, G, \mathcal{P})$ with $n = 2$ agents, we define the *characteristic tuple* $\mathcal{C}$ of the corresponding profile $\mathcal{P}$, which returns a set of goods and three numbers:

$$\mathcal{C}(\mathcal{P}) = (G, |P_1 \cap P_2|, |P_1 \setminus P_2|, |P_2 \setminus P_1|).$$

Then, for any other instance $\mathcal{I}' = (N, G, \mathcal{P}')$ with corresponding profile $\mathcal{P}'$ that has the same characteristic tuple as $\mathcal{P}$, we have that $\mathcal{P}' = \pi(\mathcal{P})$ for some permutation π of G. Consequently, by neutrality, if $\mathcal{F}(\mathcal{I}) = \mathcal{A}$ and $\mathcal{F}(\mathcal{I}') = \mathcal{A}'$, then $v_i(A_i) = v_i(A_i')$ for each $i \in N$.

We prove the induction step through the following lemma.

Lemma 5.4.3. *If $\mathcal{F}$ maximizes Nash welfare for every instance with κ goods, then $\mathcal{F}$ maximizes Nash welfare for every instance with $\kappa + 1$ goods.*

Proof sketch. Assume that $|G| = \kappa$ and consider any instance $\mathcal{I}^{\kappa+1} = (N, G \cup \{g^*\}, \mathcal{P}^{\kappa+1})$ on $\kappa + 1$ goods. Let $\mathcal{I}^{\kappa} = (N, G, \mathcal{P}^{\kappa})$ be the instance with κ goods such that $P_i^{\kappa} = P_i^{\kappa+1} \cap G$ for all $i \in N$. Let $\mathcal{A} = \mathcal{F}(\mathcal{I}^{\kappa})$ be the allocation returned by $\mathcal{F}$ on the instance $\mathcal{I}^\kappa$. By assumption, $\mathcal{A} \in \text{MNW}(\mathcal{I}^\kappa)$. Let $\mathcal{B} = \mathcal{F}(\mathcal{I}^{\kappa+1})$ be the allocation returned by $\mathcal{F}$ on the instance $\mathcal{I}^{\kappa+1}$. The characteristic tuple of $\mathcal{P}^{\kappa+1}$ is

$$\mathcal{C}(\mathcal{P}^{\kappa+1}) = (G \cup \{g^*\}, |B_1^1| + |B_2^1|, |B_1^0|, |B_2^0|).$$

The valuations of both agents in $\mathcal{I}^{\kappa+1}$ are illustrated as follows, together with the allocation $\mathcal{B}$.

$\mathcal{P}^{\kappa+1}$	$\overbrace{\hphantom{1 \cdots 1}}^{B_1^0}$	$\overbrace{\hphantom{1 \cdots 1}}^{B_1^1}$	$\overbrace{\hphantom{1 \cdots 1}}^{B_2^1}$	$\overbrace{\hphantom{1 \cdots 1}}^{B_2^0}$
1	$1 \cdots 1$	$1 \cdots 1$	$1 \cdots 1$	
2		$1 \cdots 1$	$1 \cdots 1$	$1 \cdots 1$

If $g^* \notin P_1^{\kappa+1} \cup P_2^{\kappa+1}$ (i.e., g^* is unvalued), then by minimal completeness, g^* must remain unallocated. By resource-monotonicity, $|B_1| = |A_1|$ and $|B_2| = |A_2|$, and $\mathcal{F}$ continues to maximize Nash welfare for $\mathcal{I}^{\kappa+1}$.

Assume that $g^* \in P_1^{\kappa+1} \cup P_2^{\kappa+1}$, and suppose for contradiction that $\mathcal{F}$ does not maximize Nash welfare for the instance $\mathcal{I}^{\kappa+1}$, i.e., $\mathcal{B} \notin \text{MNW}(\mathcal{I}^{\kappa+1})$. We say that $\mathcal{A} = (A_1, A_2)$ is *balanced* if $||A_1| - |A_2|| \leq 1$, and *unbalanced* otherwise. We split our analysis into two cases according to whether $\mathcal{A}$ is balanced, and derive a contradiction in each case.

Case 1: $\mathcal{A}$ is unbalanced. This means that $||A_1| - |A_2|| \geq 2$. Without loss of generality, assume that

$$|A_2| \geq |A_1| + 2. \tag{5.14}$$

If $A_2^1 \neq \emptyset$, then $(2, 1)$ is a critical path in $(\mathcal{I}^\kappa, \mathcal{A})$. Moreover, non-redundancy and minimal completeness imply PO, so $\mathcal{A}$ is PO. By Lemma 5.3.1, $\mathcal{A} \notin \text{MNW}(\mathcal{I}^\kappa)$, contradicting our assumption. We may thus assume that $A_2^1 = \emptyset$. This means that

$$|A_2^0| = |A_2| \geq 2. \tag{5.15}$$

If $B_2^1 = \emptyset$, then $\mathcal{B} \in \text{MNW}(\mathcal{I}^{\kappa+1})$, contradicting our assumption. Thus, we assume that $B_2^1 \neq \emptyset$. By resource-monotonicity and minimal completeness, this means that $|B_2^1| = 1$, $B_2^0 = A_2^0$, $|B_1| = |A_1|$, and $|B_2| = |A_2| + 1$.

Consider another instance $\widehat{\mathcal{I}} = (N, G \cup \{g^*\} \setminus \{\widehat{g}\}, \widehat{\mathcal{P}})$ derived from $\mathcal{I}^{\kappa+1}$ by removing some good $\widehat{g} \in B_2^0$. Note that $\widehat{P}_1 = P_1^{\kappa+1}$ and $\widehat{P}_2 = P_2^{\kappa+1} \setminus \{\widehat{g}\}$. By resource-monotonicity, exactly one agent's bundle value should decrease (by 1) when going from $\mathcal{I}^{\kappa+1}$ to $\widehat{\mathcal{I}}$. Let $\widehat{\mathcal{B}} = \mathcal{F}(\widehat{\mathcal{I}})$ be the allocation returned by $\mathcal{F}$ on the instance $\widehat{\mathcal{I}}$. The valuations of both agents in $\widehat{\mathcal{I}}$ are illustrated as follows, together with the goods identified with respect to the allocation $\mathcal{B}$. Note that $\mathcal{B}$ does *not* denote the allocation returned by $\mathcal{F}$ in this instance.

$$
\begin{array}{c|cccc}
\widehat{\mathcal{P}} & \overbrace{\qquad}^{B_1^0} & \overbrace{\qquad}^{B_1^1} & \overbrace{\qquad}^{B_2^1} & \overbrace{\qquad}^{B_2^0\backslash\{\widehat{g}\}} \\
\hline
1 & 1 \ \cdots \ 1 & 1 \ \cdots \ 1 & 1 \ \cdots \ 1 & \\
2 & & 1 \ \cdots \ 1 & 1 \ \cdots \ 1 & 1 \ \cdots \ 1
\end{array}
$$

Recall from (5.14) and (5.15) that

$$|B_2^0| - 1 = |A_2^0| - 1 = |A_2| - 1 \geq |A_1| + 1.$$

Since $\mathcal{F}$ maximizes Nash welfare in all instances with κ goods (in particular, $\widehat{\mathcal{I}}$), we have that $\widehat{\mathcal{B}} \in \mathrm{MNW}(\widehat{\mathcal{I}})$.

Note that $|\widehat{P}_2| = |P_2^{\kappa+1}| - 1 \geq |B_2^0| - 1 \geq |A_1| + 1$. We now show that $|\widehat{P}_1| \geq |A_1| + 1$ as well. Suppose for contradiction that $|\widehat{P}_1| \leq |A_1|$. This means that $|P_1^{\kappa+1}| = |\widehat{P}_1| \leq |A_1| = |B_1|$. Hence, agent 1 receives from $\mathcal{B}$ all the goods that she values in $\mathcal{I}^{\kappa+1}$. It follows that $\mathcal{B} \in \mathrm{MNW}(\mathcal{I}^{\kappa+1})$, contradicting our assumption. Hence, $|\widehat{P}_1| \geq |A_1| + 1$.

The total number of valued goods in instance $\widehat{\mathcal{I}}$ is

$$|B_1| + |B_2| - 1 = |A_1| + (|B_2^1| + |B_2^0|) - 1 \geq |A_1| + 1 + (|B_2^0| - 1) \geq 2(|A_1| + 1).$$

Together with the fact that

$$|\widehat{P}_1| \geq |A_1| + 1 \quad \text{and} \quad |\widehat{P}_2| \geq |A_1| + 1,$$

in order to ensure that $\widehat{\mathcal{B}}$ maximizes Nash welfare, we must have that $|\widehat{B}_1| \geq |A_1| + 1$.

Finally, recall that instance $\mathcal{I}^{\kappa+1}$ can be derived from $\widehat{\mathcal{I}}$ by adding the good $\widehat{g}$, which is only valued by agent 2. By resource-monotonicity, agent 1's bundle value cannot decrease when moving from $\widehat{\mathcal{I}}$ to $\mathcal{I}^{\kappa+1}$. This means that $|B_1| \geq |\widehat{B}_1| \geq |A_1| + 1$. However, this contradicts the fact that $|B_1| = |A_1|$.

Case 2: $\mathcal{A}$ is balanced. Assume that all goods are valued; the proof proceeds similarly if there are unvalued goods. If κ is even, then $|A_1| = |A_2|$. Since $\mathcal{F}$ is resource-monotone and one agent's utility must increase by exactly 1 when moving from $\mathcal{I}^\kappa$ to $\mathcal{I}^{\kappa+1}$, we have that $\mathcal{B} \in \mathrm{MNW}(\mathcal{I}^{\kappa+1})$, contradicting our assumption. Thus, we assume that κ is odd.

Let $\kappa + 1 = 2\ell$, for some $\ell \in \mathbb{N}$. Without loss of generality, assume that $|A_1| = \ell - 1$ and $|A_2| = \ell$. If $\mathcal{B}$ is balanced, then $\mathcal{B} \in \mathrm{MNW}(\mathcal{I}^{\kappa+1})$, contradicting our assumption. Thus, it must be that $\mathcal{B}$ is unbalanced. By resource-monotonicity, this means that $|B_1| = |A_1| = \ell - 1$ and $|B_2| = |A_2| + 1 = \ell + 1$. Note also that

$$|B_2^0| \leq \ell; \tag{5.16}$$

otherwise $\mathcal{B} \in \mathrm{MNW}(\mathcal{I}^{\kappa+1})$, contradicting our assumption.

Consider the instance $\mathcal{I}^* = (N, G \cup \{g^*\}, \mathcal{P}^*)$ derived from $\mathcal{I}^{\kappa+1}$ by letting all goods in B_2^0 be valued by agent 1—that is, $P_1^* = P_1^{\kappa+1} \cup B_2^0 = B_1 \cup B_2$ and $P_2^* = P_2^{\kappa+1}$. Let $\mathcal{B}^* = \mathcal{F}(\mathcal{I}^*)$ be the corresponding allocation returned by $\mathcal{F}$ on the instance $\mathcal{I}^*$. The characteristic tuple of $\mathcal{P}^*$ is

$$\mathcal{C}(\mathcal{P}^*) = (G \cup \{g^*\}, |B_1^1| + |B_2|, |B_1^0|, 0).$$

The valuations of both agents in $\mathcal{I}^*$ are illustrated as follows, together with the goods identified with respect to the allocation $\mathcal{B}$. Note that $\mathcal{B}$ does *not* denote the allocation returned by $\mathcal{F}$ in this instance.

$$\begin{array}{c|cccc}
\mathcal{P}^* & \overbrace{\qquad\quad}^{B_1^0} & \overbrace{\qquad\quad}^{B_1^1} & \overbrace{\qquad\quad}^{B_2^1} & \overbrace{\qquad\quad}^{B_2^0} \\
\hline
1 & 1 \ \cdots \ 1 & 1 \ \cdots \ 1 & 1 \ \cdots \ 1 & \mathbf{1} \ \cdots \ \mathbf{1} \\
2 & & 1 \ \cdots \ 1 & 1 \ \cdots \ 1 & 1 \ \cdots \ 1
\end{array}$$

We have $\ell > |B_1| \geq |B_1^0|$. Also, by EF1, it holds that $|B_1^*| \geq \ell$. Let $S \subseteq G$ be a subset of ℓ goods that contains

(i) all $|B_1^0|$ goods from B_1^0, and

(ii) $\ell - |B_1^0|$ goods from $B_1^* \setminus B_1^0$.

Now, consider another instance $\widetilde{\mathcal{I}} = (N, G \cup \{g^*\}, \widetilde{\mathcal{P}})$ constructed from $\mathcal{I}^*$ by declaring some $|B_2^0|$ goods in $B_1^1 \cup B_2$ to be unvalued by agent 1. We choose any such $|B_2^0|$ goods from the set $(B_1^1 \cup B_2) \setminus S$. Note that $|B_1 \cup B_2| = 2\ell$, and so $|(B_1^1 \cup B_2) \setminus S| = |(B_1 \cup B_2) \setminus S| = 2\ell - \ell = \ell$. Since $|B_2^0| \leq \ell$ by (5.16), such $|B_2^0|$ goods in $(B_1^1 \cup B_2) \setminus S$ exist. Let $\widetilde{\mathcal{B}} = \mathcal{F}(\widetilde{\mathcal{I}})$ be the corresponding allocation returned by $\mathcal{F}$ on the instance $\widetilde{\mathcal{I}}$.

The characteristic tuple of $\widetilde{\mathcal{P}}$ is

$$\mathcal{C}(\widetilde{\mathcal{P}}) = (G \cup \{g^*\}, |B_1^1| + |B_2^1|, |B_1^0|, |B_2^0|).$$

The valuations of both agents in $\widetilde{\mathcal{I}}$ are illustrated as follows, together with the goods identified with respect to the bundles B_1^0 and $B_1^1 \cup B_2$ and the set S. Note that $\mathcal{B}$ does *not* denote the allocation returned by $\mathcal{F}$ in this instance, and the order of goods in $B_1^1 \cup B_2$ may differ from that illustrated in $\mathcal{P}^*$ above.

$$\begin{array}{c|cccc}
\widetilde{\mathcal{P}} & \overbrace{\qquad\quad}^{B_1^0} & \multicolumn{3}{c}{\overbrace{\qquad\qquad\qquad\qquad\qquad\qquad}^{B_1^1 \cup B_2}} \\
\hline
1 & 1 \ \cdots \ 1 & 1 \ \cdots \ 1 & 1 \ \cdots \ 1 & \mathbf{0} \ \cdots \ \mathbf{0} \\
2 & & 1 \ \cdots \ 1 & 1 \ \cdots \ 1 & 1 \ \cdots \ 1 \\
 & \multicolumn{2}{c}{\underbrace{\qquad\qquad\qquad\qquad}_{S}} & & \underbrace{\qquad\quad}_{|B_2^0| \text{ goods}}
\end{array}$$

Since $\mathcal{C}(\widetilde{\mathcal{P}}) = \mathcal{C}(\mathcal{P}^{\kappa+1})$, by neutrality, we must have that

$$|\widetilde{B}_1| = |B_1| = \ell - 1 \quad \text{and} \quad |\widetilde{B}_2| = |B_2| = \ell + 1.$$

Moreover, recall that $|B_1^*| \geq \ell$. Comparing instance $\widetilde{\mathcal{I}}$ to $\mathcal{I}^*$, we observe that the rule is not strategyproof—agent 1 can misreport her valuations and obtain a strictly higher utility, giving us a contradiction. □

Lemma 5.4.3 and induction implies Theorem 5.4.2. □

Next, we present a result showing that any rule satisfying five of the six properties (we do not require non-redundancy here) and maximizing Nash welfare must consistently break ties between instances with the same number of valued goods where tie-breaking is relevant. As an immediate corollary, the same must hold for any rule satisfying all six properties, by Theorem 5.4.2. Note that because of resource-monotonicity, this tie-breaking is irrespective of the total number of goods. This is in contrast to Theorem 5.3.5, where the instances that must have consistent tie-breaking also need to have the same number of goods (in addition to having the same number of valued goods).

Theorem 5.4.4. *Let $n = 2$, and let $\mathcal{F}$ be a rule that is* EF1*, minimally complete, resource-monotone, strategyproof, neutral, and maximizes Nash welfare. Then, for each m_v, there exists an agent $i \in \{1, 2\}$ such that for every instance with m_v valued goods for which tie-breaking is relevant, $\mathcal{F}$ chooses an MNW allocation favoring agent i.*

Proof. Due to minimal completeness and resource-monotonicity, each agent's bundle value should remain the same upon adding or removing unvalued goods. Thus, without loss of generality, we can assume that all goods are valued in every instance that we consider.

Suppose for contradiction that there exists an instance with m valued goods where tie-breaking is relevant such that $\mathcal{F}$ returns an MNW allocation favoring agent 1, and another instance such that $\mathcal{F}$ returns an MNW allocation favoring agent 2. As tie-breaking is never relevant for an even number of valued goods, we may let $m = 2k+1$. Consider an instance $\mathcal{I}^*$ with m valued goods, where all goods are valued by both agents, and let $\mathcal{F}(\mathcal{I}^*) = \mathcal{A}^*$. Since $\mathcal{A}^*$ is EF1, one agent must receive a utility of $k+1$ and the other agent a utility of k. Without loss of generality, assume that

$$|A_1^*| = k \quad \text{and} \quad |A_2^*| = k + 1.$$

Let $\mathcal{I}$ be an instance with m valued goods where tie-breaking is relevant such that $\mathcal{F}(\mathcal{I}) = \mathcal{A}$ is an MNW allocation favoring agent 1. We have

$$|A_1| = k + 1 \quad \text{and} \quad |A_2| = k.$$

Since tie-breaking is relevant in this instance, $|A_1^1| \geq 1$ (the notation A_1^1 is defined before Theorem 5.4.2). This means that

$$|A_1^0| \leq k. \tag{5.17}$$

Next, consider the modified instance $\widehat{\mathcal{I}} = (N, G, \widehat{\mathcal{P}})$ derived from $\mathcal{I}$ by letting agent 1 value all goods in A_2^0, i.e., $\widehat{P}_1 = A_1 \cup A_2$ and $\widehat{P}_2 = P_2$. Let $\widehat{\mathcal{A}} = \mathcal{F}(\widehat{\mathcal{I}})$. If $|\widehat{A}_1| = k$, agent 1 can strictly benefit by misreporting her valuations from $\widehat{\mathcal{P}}$ to $\mathcal{P}$, violating strategyproofness. Hence,

$$|\widehat{A}_1| = k + 1 \quad \text{and} \quad |\widehat{A}_2| = k.$$

Finally, consider the instance derived from $\mathcal{I}^*$ by letting agent 2 have value 0 for some $|A_1^0|$ goods in A_1^*; the existence of such goods is guaranteed by (5.17). Then, by neutrality with $\widehat{\mathcal{I}}$, agent 1 should get a utility of $k+1$ and agent 2 a utility of k. However, comparing this instance to $\mathcal{I}^*$, agent 2 can obtain a strictly higher utility (with respect to the goods that she values in this instance) by misreporting her valuations. This violates strategyproofness, a contradiction. □

Next, we show that all six properties are necessary for our characterization. In particular, if we were to drop any one of EF1, non-redundancy, strategyproofness, neutrality, or minimal completeness, the allocation rule may no longer maximize Nash welfare. On the other hand, if we were to drop resource-monotonicity, then the allocation rule may fail to break ties consistently as mandated by Theorem 5.4.4.

Proposition 5.4.5. *Under binary valuations, given only five of the properties in {EF1, non-redundancy, strategyproofness, resource-monotonicity, neutrality, minimal completeness}, there exists an allocation rule for $n = 2$ that satisfies the five properties but either does not maximize Nash welfare or fails the tie-breaking requirement set out in Theorem 5.4.4.*

Proof. We show that for each of the six properties (EF1, non-redundancy, minimal completeness, neutrality, resource-monotonicity, strategyproofness), there exists an allocation rule for $n = 2$ that does not maximize Nash welfare (or in the case of resource-monotonicity, fails the tie-breaking requirement set out in Theorem 5.4.4) and fails only that property among the six. We include an example demonstrating the violation of each property (and MNW) by the proposed rule.

- **EF1**: A rule that discards all unvalued goods, allocates each good valued by only one agent to that agent, and allocates goods valued by both agents to agent 1.

> To see that the rule is not EF1, consider an instance with $m = 2$ goods that are valued by both agents. Then, the rule will allocate both goods to agent 1, violating EF1. The same example shows that the rule returns an allocation that is not MNW.

- **Non-redundancy**: The MNW^{tie} rule favoring agent 1, with one exception: if $m = 1$ and the good is valued only by agent 2, allocate it to agent 1.

> To see that the rule fails non-redundancy, observe that when $m = 1$ and the good is only valued by agent 2, agent 1 receives a good that she does not value. The allocation returned by the rule in this case is not MNW.

- **Minimal completeness**: The MNW^{tie} rule favoring agent 1, with one exception: if $m = 1$ and both agents value the good, discard it.

> To see that the rule is not minimally complete, observe that it does not allocate a valued good when $m = 1$ and both agents value the good. This allocation is also not MNW.

- **Neutrality**: The MNW^{tie} rule favoring agent 1, with the exception that for all instances with two valued goods (denote these goods by g_i and g_j, where $i < j$) and any number of unvalued goods:
  - (i) if both agents value g_i and g_j, then allocate g_i to agent 2 and g_j to agent 1;
  - (ii) if agent 1 values both g_i and g_j, whereas agent 2 values only g_j, then allocate both g_i and g_j to agent 1.

To see that the rule is not neutral, consider the following three profiles (with no unvalued goods):

$\mathcal{P}_1$	g_1	g_2
1	1	1
2	0	1

and

$\mathcal{P}_2$	g_1	g_2
1	1	1
2	1	1

and

$\mathcal{P}_3$	g_1	g_2
1	1	1
2	1	0

The rule will return the above indicated allocations, using (ii) for $\mathcal{P}_1$, (i) for $\mathcal{P}_2$, and the MNW$^{\text{tie}}$ rule favoring agent 1 for $\mathcal{P}_3$. However, $\mathcal{P}_3 = \pi(\mathcal{P}_1)$ for some permutation π of G, so by neutrality the utility that each agent gets in both of these instances should be equal, which is not the case. The allocation returned by the rule for $\mathcal{P}_1$ is not MNW.

- **Resource-monotonicity**: The MNW$^{\text{tie}}$ rule favoring agent 1, with one exception: if $m = 2$ and one good is valued by both agents while the other good is unvalued, allocate the valued good to agent 2 and discard the unvalued good.

To see that the rule is not resource-monotone, consider the instance where $m = 1$ and the good is valued by both agents. Then, the MNW$^{\text{tie}}$ rule favoring agent 1 will allocate this good to agent 1. When an unvalued good is added, agent 1's bundle becomes empty, violating resource-monotonicity. Observe that these two instances have the same number of valued goods. However, tie-breaking is in favor of agent 1 in the instance with $m = 1$ but in favor of agent 2 in the instance with $m = 2$. This violates the tie-breaking requirement set out by Theorem 5.4.4.

- **Strategyproofness**: The MNW$^{\text{tie}}$ rule favoring agent 1, with the exception that for all instances with two valued goods (denote these goods by g_i and g_j) and any number of unvalued goods, such that agent 1 values both g_i and g_j and agent 2 values only one of them, allocate both valued goods to agent 1.

To see that the rule is not strategyproof, consider the following two profiles (with no unvalued goods):

$\mathcal{P}_1$	g_1	g_2
1	1	1
2	0	1

and

$\mathcal{P}_2$	g_1	g_2
1	1	1
2	1	1

The rule will return the above indicated allocations. In $\mathcal{P}_1$, agent 2 can manipulate by reporting 1 on g_1 and increase her utility from 0 to 1. The allocation returned by the rule for $\mathcal{P}_1$ is not MNW.

□

5.5 Conclusion

In this chapter, we have presented characterizations of the popular maximum Nash welfare (MNW) rule for binary valuations in the offline fair division setting. Our main result shows that for any number of agents, any rule satisfying EF1, strategyproofness, neutrality, minimal completeness, and IDU must maximize Nash welfare. We also provide an alternative characterization for the case of two agents, by replacing IDU with non-redundancy and resource-monotonicity. To the best of our knowledge, these are the first characterizations of MNW within the space of all rules for any meaningful valuation class.

Besides extending Theorem 5.4.2 to $n > 2$ or obtaining characterizations using other sets of axioms, a natural future direction is to characterize MNW among all rules for additive valuations. As discussed in Section 5.1, this appears to be a challenging task, as several properties fulfilled by MNW in the binary setting are violated by the rule in the additive setting. More broadly, while characterizations are commonplace in the social choice literature, they are still few and far between in fair division despite the recent surge of interest in the area.

By fully characterizing MNW in the offline binary setting, this chapter deepens our theoretical understanding of a foundational fairness objective and provides tools and insights that may inform the design of fair and efficient algorithms for online settings.

Part II

Temporal Voting

Chapter 6

Temporal Elections: Welfare, Manipulation, and Proportionality

6.1 Overview

In the first part of this thesis, we studied the allocation of items that are assigned *privately* to agents—that is, each item is allocated exclusively to one agent and cannot be shared. In this chapter, which begins the second part of the thesis, we turn our attention to a sequential decision-making model, *temporal voting*. This can also be interpreted as the selection of *public* goods—items or candidates that are shared among agents.

Consider a group of friends planning their itinerary for a two-week post-graduation trip across Europe. They have selected their activities, but still need to decide on their choice of meals for each day. As popular restaurants typically require reservations, everyone is asked to declare their preferences upfront before the trip commences.

Suppose that 55% of the group members prefer Asian cuisine, 25% prefer European cuisine, 10% prefer Oceanic cuisine, and the remaining 10% prefer South American cuisine. If they need to schedule 40 meals, it would be fair to select European restaurants for 10 of these meals, and plan 22 visits to Asian restaurants, with the remaining 8 meals split equally between Oceanic and South American establishments. However, if the friends were to adopt plurality voting to decide on each meal, Asian cuisine would be chosen for every meal, and, as a result, 45% of the group will be perpetually unhappy. A natural question is then: what would be an appropriate notion of *satisfaction*, and can we (efficiently) select an outcome that offers high satisfaction?

As the group moves from city to city, the set of available restaurants changes. Even within the same town, one may have different preferences for lunch and dinner, opting for a heavier meal option at lunch and a lighter one at dinner. As both preferences and the set of alternatives may evolve over time, traditional multiwinner voting models[1] [Elkind et al., 2017, Faliszewski et al., 2017, Lackner and Skowron, 2023] do not fully capture this setting.

This problem fits within the *temporal elections* framework, a model where a sequence of decisions is made, and outcomes are evaluated with respect to agents' temporal preferences. It was first introduced as *perpetual voting* by Lackner [2020]; see the survey of Elkind et al. [2024d] for a discussion of subsequent work.

We consider the *offline* variant of this model where preferences are provided upfront. That is, at each timestep, each agent has a set of approved alternatives, and the goal is to select a single alternative per timestep.

While this model has been considered in prior work [Bulteau et al., 2021, Chandak et al., 2024, Elkind et al., 2025c], earlier papers focus on fairness and proportionality notions, with only a few looking into welfare objectives and strategic considerations. Against this background, in this work we focus on the task of maximizing two classic welfare objectives: the utilitarian welfare (the sum of agents' utilities) and the egalitarian welfare (the minimum utility of any agent), both in isolation and in combination with *strategyproofness* (meaning that no agent can obtain a strictly better outcome by misreporting their preferences) and proportionality (ensuring that each agent receives at least a fair share of their maximum possible utility) axioms. We refer to these objectives (as well as, overloading notation, to outcomes that maximize them) as Util and Egal, respectively.

In this chapter, we investigate the Util and Egal objectives from three perspectives: the computational complexity of welfare maximization, compatibility with strategyproofness, and trade-offs with proportionality.

In Section 6.3, we observe that a Util outcome can be computed in polynomial time, whereas computing an Egal outcome is NP-hard even under strong restrictions on the input. To mitigate the hardness result for Egal maximization, we analyze the parameterized complexity of this problem with respect to several natural parameters, and provide an ILP-based approximation algorithm.

In Section 6.4, we show that, while a mechanism that outputs a Util outcome is strategyproof, any deterministic mechanism for Egal fails non-obvious manipulability (NOM), which is a relaxation of strategyproofness. On the positive side, if each agent has a non-empty approval set at each timestep, the mechanism that selects

[1]Multiwinner voting is a collective decision-making framework where a group of voters selects a subset of candidates—or committee of winners—from a larger set of available options. These models capture a wide variety of settings, from parliamentary elections in democratic systems to product placement in online shopping platforms.

among EGAL outcomes using leximin tie-breaking satisfies NOM; however, even in this special case the EGAL objective admits no deterministic strategyproof mechanism in general.

In Section 6.5, we focus on proportionality. We show that, while a simple greedy algorithm can return a proportional (PROP) outcome, it is NP-hard to determine if there exists a PROP outcome that also maximizes UTIL, even when each agent has a non-empty approval set at each timestep. We also provide upper and lower bounds on the (strong) price of proportionality with respect to both UTIL and EGAL. To the best of our knowledge, our work is the first to investigate the price of proportionality in temporal elections. Towards the end of the chapter, we discuss how to extend our results to the general class of p-mean welfare objectives.

6.2 Preliminaries

We do not require each agent to approve at least one project at each timestep; however, we do require that each agent approves at least one project at some timestep, i.e., for each $i \in N$ there exists a $t \in [\ell]$ with $s_{i,t} \neq \varnothing$; indeed, if this condition fails for some $i \in N$, we can simply delete i, as it can never be satisfied. If $s_{i,t} \neq \varnothing$ for all $i \in N$ and $t \in [\ell]$, we say that the agents have *complete preferences (CP)*. We believe that the CP setting captures many real-life applications of our model: for instance, in our motivating example, having no particular opinion on any cuisine would be more reasonably interpreted as approving all options rather than having an empty approval set. We also consider the more restricted setting where $|s_{i,t}| = 1$ for all $i \in N$ and $t \in [\ell]$; in this case, we say that the agents have *unique preferences (UP)*.

6.3 Welfare Maximization

We first focus on welfare maximization; subsequently, we will explore combining welfare objectives with other desiderata. The two objectives we consider are defined as follows.

Definition 6.3.1 (Social Welfare)**.** Given an outcome $\mathbf{o}$, its *utilitarian social welfare* is defined as $\text{UTIL}(\mathbf{o}) = \sum_{i \in N} u_i(\mathbf{o})$ and its *egalitarian social welfare* is defined as $\text{EGAL}(\mathbf{o}) = \min_{i \in N} u_i(\mathbf{o})$. We refer to outcomes that maximize the utilitarian/egalitarian welfare as UTIL/EGAL outcomes, respectively.

A Util outcome can be found in polynomial time: at each timestep, we select a project that has the highest number of approvals. However, if 51% of the population approves project p (and nothing else) at each timestep, while 49% of the population approves project q (and nothing else) at each timestep, the outcome $(p, \ldots, p)$ is the unique Util outcome, but close to half of the population will obtain utility 0 from it. Thus, it is important to consider desiderata other than Util, such as, e.g., egalitarian welfare. However, while computing a Util outcome is computationally feasible, this is not the case for Egal outcomes.

The decision problem associated with computing Egal outcomes, which we denote by Egal-Dec, is defined as follows.

Maximizing Egalitarian Welfare (Egal-Dec):
Input: An instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$ and a parameter $\lambda \in \mathbb{Z}^+$.
Question: Is there an outcome $\mathbf{o}$ with $\text{Egal}(\mathbf{o}) \geq \lambda$?

The following result shows that Egal-Dec is NP-complete, even in the UP setting and even when the goal is to guarantee each agent a utility of 1.

Theorem 6.3.2. Egal-Dec *is* NP*-complete, even in the UP setting with* $\lambda = 1$.

Proof. It is immediate that Egal-Dec is in NP: given a candidate outcome, we can easily check if it provides utility of at least λ to each agent.

To prove hardness, we reduce from the classic Vertex Cover problem. An instance of this problem is a pair (G, k), where $G = (V, E)$ is an undirected graph and k is a positive integer; it is a yes-instance if there is a subset $V' \subseteq V$ of size at most k such that for every edge $\{v, v'\} \in E$ we have $v \in V'$ or $v' \in V'$, and a no-instance otherwise.

Given an instance (G, k) of Vertex Cover with $G = (V, E)$, $V = \{v_1, \ldots, v_s\}$, $|E| = r$, we proceed as follows. We let $N = [r + s - k]$, where the first r agents in N are edge agents (i.e., one agent for each edge of G), and the remaining $s - k$ agents are dummy agents (where k is the target size of the vertex cover, so $k \leq s$), and set $n = |N| = r + s - k$, $\ell = s$.

Let $P^0 = \{q_1, \ldots, q_n\}$, $P^* = \{p_1, \ldots, p_s\}$, and set $P = P^0 \cup P^*$. For each $t \in [\ell]$, we set the approval set of the edge agent i at timestep t to $\{p_t\}$ if i is an edge agent that corresponds to an edge incident to v_t; if i is a dummy agent or an edge agent who corresponds to an edge not incident to v_t, we set $s_{i,t} = \{q_i\}$. By construction, the agents have unique preferences. Let $\lambda = 1$.

Suppose G admits a vertex cover V' of size k. We then construct an outcome $\mathbf{o} = (o_1, \ldots, o_\ell)$ as follows. For each $v_j \in V'$ we set $o_j = p_j$. The remaining $s - k$ timesteps are assigned arbitrarily to the $s - k$ dummy agents; if a dummy agent i receives timestep t, we set $o_t = q_i$. By construction, the utility of every dummy agent is 1. We claim that all edge agents have positive utility as well. Indeed, consider an agent that corresponds to an edge $\{v_j, v_t\}$. Since V' is a vertex cover, we have $o_j = p_j$ or $o_t = p_t$ (or both), so the utility of this agent is at least 1.

Conversely, consider an outcome $\mathbf{o}$ with $u_i(\mathbf{o}) \geq 1$ for all $i \in N$. A dummy agent i can only be satisfied if we set $o_t = q_i$ for some $t \in [\ell]$, and this provides zero utility to agents other than i. Thus, $\mathbf{o}$ has to allocate at least $s - k$ timesteps to satisfying the dummy agents. Hence, $\mathbf{o}$ uses at most k timesteps to provide utility of 1 to each edge agent. We will use this fact to construct a set of vertices V' that forms a vertex cover of size at most k for G. Let T be the set of timesteps in $[\ell]$ that are not allocated to dummy agents; we have $|T| \leq k$. We initialize $V' = \varnothing$. Then, for each timestep $t \in T$, if $o_t = p_j$ for some $p_j \in P^*$ we place v_j in V', and if $o_t = q_j$ for some $j \in [r]$, we place one of the endpoints of the edge that corresponds to agent j in V'. The resulting set V' has size at most $|T| \leq k$; moreover, as it 'covers' each edge agent, it has to form a vertex cover for G. $\square$

Theorem 6.3.2 effectively rules out the possibility of maximizing the egalitarian welfare even in simple settings. Therefore, it what follows, we consider the complexity of egalitarian welfare maximization from the parameterized complexity perspective (Section 6.3.1) and from the approximation algorithms perspective (Section 6.3.2).

6.3.1 Parameterized Complexity of Egalitarian Welfare

The EGAL-DEC problem has four natural parameters: the number of agents n, the number of projects m, the number of timesteps ℓ, and the utility guarantee λ. We will now consider these parameters one by one.

We first show that EGAL-DEC is in FPT with respect to n. Our proof is based on integer linear programming. Specifically, we show how to encode EGAL-DEC as an integer linear program (ILP) whose number of variables depends on n only; our claim then follows from Lenstra's classic result Lenstra Jr [1983]. To accomplish this, we classify the projects and timesteps into 'types', so that the number of types is (doubly) exponential in n, but does not depend on m or ℓ.

Theorem 6.3.3. EGAL-DEC *is* FPT *with respect to n.*

Proof. As a preprocessing step, we create ℓ copies of each project. That is, we replace a project p with projects $p^1, \dots, p^\ell$ and modify the approval vectors: for each $i \in N$, $t \in [\ell]$ and $p \in s_{i,t}$, we place p^t in $s_{i,t}$ and remove p. This does not change the nature of our problem, since in our setting there is no dependence between timesteps, and a project's label can be re-used arbitrarily between timesteps. For the modified instance, it holds that for each project p there is at most one timestep $t \in [\ell]$ such that $p \in \cup_{i \in N} s_{i,t}$. Then, we define the *type* of a project as the set of voters who approve it: the type of p is $\pi(p) = \{i \in N : p \in \cup_{t \in [\ell]} s_{i,t}\}$. Because of the preprocessing step, for each project p there is a timestep $t \in [\ell]$ such that $p \in s_{i,t}$ for all $i \in \pi(p)$, and $p \notin s_{i,t'}$ for all $i \in N$, $t' \in [\ell] \setminus \{t\}$. Let $\mathcal{P} \subseteq 2^N$ be the set of all project types; by construction, we have $|\mathcal{P}| \le 2^n$.

In the same way, we can classify timesteps by the types of projects present in them: the type of a timestep t is defined as $\tau(t) = \{\pi(p) : p \in \cup_{i \in N} s_{i,t}\}$. Let $\mathcal{T} \subseteq 2^{\mathcal{P}}$ be the set of all timestep types; by construction, we have $|\mathcal{T}| \le 2^{|\mathcal{P}|} \le 2^{2^n}$.

We are now ready to construct our ILP. For each $\tau \in \mathcal{T}$, let z_τ be the number of timesteps of type τ; these quantities can be computed from the input. For each $\tau \in \mathcal{T}$ and $\pi \in \tau$, we introduce an integer variable $x_{\tau,\pi}$ representing the number of timesteps of type τ in which a project of type π was chosen.

The ILP is defined as follows:

$$\text{maximize } \lambda \quad \text{subject to:}$$

(1) $\sum_{\pi \in \tau} x_{\tau,\pi} \le z_\tau$ for each $\tau \in \mathcal{T}$;

(2) $\sum_{\tau \in \mathcal{T}} \sum_{\pi \in \mathcal{P} : i \in \pi} x_{\tau,\pi} \ge \lambda$ for each $i \in N$;

(3) $x_{\tau,\pi} \ge 0$ for each $\tau \in \mathcal{T}$ and $\pi \in \mathcal{P}$.

The first constraint requires that we select at most one project per timestep, whereas the second constraint ensures that each agent has utility of at least λ from the outcome.

There are at most $\mathcal{O}(2^{n+2^n})$ variables in the ILP, so the classic result of Lenstra Jr [1983] implies that our problem is FPT with respect to n. □

For the number of agents m, we can show that EGAL-DEC is NP-hard even for $m = 2$; this hardness result holds even if $\lambda = 1$, but our reduction produces instances where $s_{i,t} = \varnothing$ for some $i \in N$, $t \in [\ell]$, i.e., it does *not* show that EGAL-DEC is NP-hard in the CP setting (and hence Theorem 6.3.4 below does not imply Theorem 6.3.2). Our argument is similar in spirit to the proof of Theorem 2 of Deltl et al. [2023].

Theorem 6.3.4. EGAL-DEC *is* NP*-complete even if* $m = 2$*,* $\lambda = 1$*.*

Proof. We reduce from the classical 3-SAT problem. Consider a formula F with α variables $x_1, \dots, x_\alpha$, and κ clauses $M_1, \dots, M_\kappa$. Each clause is a disjunction of at most three variables or their negations. F is satisfiable if there exists some assignment of Boolean values to variables such that the conjunction of all κ clauses evaluates to True, and not satisfiable otherwise.

To reduce 3-SAT to our problem, we introduce κ agents, a set of projects $P = \{\tau, \varphi\}$, and α timesteps; we set $\lambda = 1$. For each $i \in [\kappa]$ and $t \in [\alpha]$, we define $s_{i,t}$ as follows.

$$s_{i,t} = \begin{cases} \{\tau\} & \text{if } x_t \text{ is in } M_i \\ \{\varphi\} & \text{if } \neg x_t \text{ is in } M_i \\ \varnothing & \text{otherwise} \end{cases}$$

We claim that F is satisfiable if and only if there exists an outcome $\mathbf{o}$ that satisfies $u_i(\mathbf{o}) \geq 1$ for all $i \in [\kappa]$.

For the 'if' direction, an outcome $\mathbf{o} = (o_1, \dots, o_\alpha)$ can be interpreted a Boolean assignment to the variables in F: for each $t \in [\alpha]$ we set x_t to True if $o_t = \tau$ and to False if $o_t = \varphi$. Then for each $i \in [\kappa]$ her utility $u_i(\mathbf{o})$ is the number of literals in M_i that are set to True by this assignment. Hence, if for all $i \in [\kappa]$ it holds that $u_i(\mathbf{o}) \geq 1$, then each clause will evaluate to True, and hence F is satisfiable.

For the 'only if' direction, given a satisfying Boolean assignment for F, we construct an outcome $\mathbf{o} = (o_1, \dots, o_\alpha)$ by setting $o_t = \tau$ if this assignment sets x_t to True and $o_t = \varphi$ otherwise. We then have $u_i(\mathbf{o}) \geq 1$ for all $i \in [\kappa]$. □

Next, we show that EGAL-DEC is in XP with respect to ℓ. Our argument is based on exhaustive search.

Theorem 6.3.5. EGAL-DEC *is in* XP *with respect to* ℓ*.*

Proof. Observe that there are m^ℓ possible outcomes. We can thus iterate through all outcomes; we output 'yes' if we find an outcome that provides utility λ to all agents, and 'no' otherwise. □

Both Theorem 6.3.3 and Theorem 6.3.5 provide algorithms for EGAL-DEC, which is a decision problem. However, both algorithms can be modified to output an EGAL outcome.

We complement our XP result for ℓ by showing that EGAL-DEC is W[2]-hard with respect to the number of timesteps. This indicates that an FPT (in ℓ) algorithm does not exist unless FPT = W[2], and hence the XP result of Theorem 6.3.5 is tight.

Theorem 6.3.6. EGAL-DEC *is* W[2]*-hard with respect to ℓ, even in the CP setting with $\lambda = 1$.*

Proof. We reduce from the DOMINATING SET (DS) problem. An instance of DS consists of a graph $G = (V, E)$ and an integer κ; it is a yes-instance if there exists a subset $D \subseteq V$ such that $|D| \leq \kappa$ and every vertex of G is either in D or has a neighbor in D, and a no-instance otherwise. DS is known to be W[2]-complete with respect to the parameter κ [Niedermeier, 2006].

Given an instance (G, κ) of DOMINATING SET with $G = (V, E)$, $V = \{v_1, \dots, v_n\}$, set $N = [n]$, $P = \{p_1, \dots, p_n\}$, $\ell = \kappa$. Then for each $i \in N$ and $t \in [\ell]$ let $s_{i,t} = \{p_j : i = j \text{ or } \{v_i, v_j\} \in E\}$. We claim that G admits a dominating set D with $|D| \leq \kappa$ if and only if there exists an outcome $\mathbf{o}$ such that $u_i(\mathbf{o}) \geq 1$ for all agents $i \in N$.

For the 'if' direction, consider an outcome $\mathbf{o} = (p_{j_1}, \dots, p_{j_\kappa})$ that provides positive utility to all agents, and set $D = \{v_{j_1}, \dots, v_{j_\kappa}\}$. Then D is a dominating set of size at most κ. Indeed, consider a vertex $v_i \in V$. Since voter i approves p_{j_t} for some $j_t \in T$, we have $v_{j_t} \in D$ and $i = j_t$ or $\{v_i, v_{j_t}\} \in E$. Note that if there are projects chosen more than once, we simply have $|D| < \kappa$.

For the 'only if' direction, observe that a dominating set $D = \{v_{j_1}, \dots, v_{j_s}\}$ with $s \leq \kappa$ can be mapped to an outcome $\mathbf{o} = (p_{j_1}, \dots, p_{j_s}, p_1, \dots, p_1)$ (with p_1 selected in the last $\kappa - s$ timesteps). As any vertex of G is either in D, or has a neighbor in D, we have $u_i(\mathbf{o}) \geq 1$ for each agent $i \in N$. □

Finally, we consider parameter λ. By Theorems 6.3.2 and 6.3.4, EGAL-DEC is hard even for $\lambda = 1$; this result holds even in the UP setting (Theorem 6.3.2) or if $m = 2$ (Theorem 6.3.4). However, if both all approval sets are non-empty (which is a weaker condition than UP) and $m = 2$, we obtain a positive result.

Theorem 6.3.7. EGAL-DEC *is in* XP *with respect to λ in the* CP *setting with $m = 2$.*

Proof. Suppose first that $\ell \leq \lambda \cdot \lceil \log_2 n \rceil \leq \lambda \cdot (\log_2 n + 1)$. We can then enumerate all $m^\ell \leq (2n)^\lambda$ possible outcomes, compute the egalitarian welfare for each outcome, and return 'yes' if there exists an outcome in which the utility of each agent is at least λ. Therefore, from now we will assume that $\ell > \lambda \cdot \lceil \log_2 n \rceil$. We will now argue that in this case we can greedily construct an outcome which guarantees utility of at least λ to each agent, i.e., our instance is a yes-instance of EGAL-DEC.

We split $[\ell]$ into λ consecutive blocks $T_1, \dots, T_\lambda$ of length at least $\lceil \log_2 n \rceil$ each. It suffices to argue that for each block T_j we can assign projects to timesteps in T_j so that each agent derives positive utility from at least one timestep in T_j.

Consider the block T_1. We start by setting $N^* = N$, and proceed in $|T_1|$ steps. Let $P = \{p, q\}$. At each step t, $t = 1, \ldots, |T_1|$, if at least half of the agents in N^* approve p at timestep $t \in T_1$, we set $o_t = p$ and remove all agents i with $p \in s_{i,t}$ from N^*; otherwise, we set $o_t = q$ and remove all agents i with $q \in s_{i,t}$ from N^*. Note that an agent is removed from N^* only after we ensure that she derives positive utility from at least one timestep in T_1. Moreover, since we are in the CP setting, at each timestep t at least one project in P is supported by at least half of the remaining agents, so at each step we reduce the size of N^* by at least a factor of 2. It follows that after $\lceil \log_2 n \rceil \leq |T_1|$ steps the set N^* is empty, i.e., each agent in N derives positive utility from some timestep in T_1; we can then assign projects to remaining timesteps in T_1 arbitrarily. By repeating this procedure for $T_2, \ldots, T_\lambda$, we construct an outcome $\mathbf{o}$ with $u_i(\mathbf{o}) \geq \lambda$ for each $i \in N$; moreover, our procedure runs in polynomial time. □

6.3.2 Approximation of Egalitarian Welfare

Theorem 6.3.2 shows that EGAL-DEC is NP-complete even when $\lambda = 1$. This implies that the problem of computing the EGAL welfare is inapproximable: an approximation algorithm would be able to detect whether a given instance admits an outcome with positive egalitarian social welfare. However, suppose we redefine each agent's utility function as $u'_i(\mathbf{o}) = 1 + u_i(\mathbf{o})$; this captures, e.g., settings where there is a timestep in which all agents approve the same project. We will now show that we can obtain a $\frac{1}{4 \log n}$-approximation to the optimal egalitarian welfare with respect to the utility profile $(u'_1, \ldots, u'_n)$.

Theorem 6.3.8. *There is a polynomial-time randomized algorithm that, for any $\varepsilon > 0$, given an instance $(P, N, \ell, (\mathbf{s}_i)_{i \in N})$, with probability $1 - \varepsilon$ outputs an outcome $\mathbf{o}$ whose egalitarian social welfare is at least $\frac{1}{4 \ln n}$ of the optimal egalitarian social welfare with respect to the modified utility functions $(u'_1, \ldots, u'_n)$.*

Proof. First, we construct a polynomial-size integer program for finding outcomes whose egalitarian welfare with respect to the modified utilities is at least a given quantity η. For each $p \in P$ and $t \in [\ell]$, we define a variable $x_{pt} \in \{0, 1\}$: $x_{pt} = 1$ encodes that p is selected at time t. Our constraints require that (1) for each $t \in [\ell]$, at most one project can be chosen at timestep t: $\sum\limits_{p \in P} x_{pt} \leq 1$, and (2) each agent $i \in N$ approves the outcome in at least $\eta - 1$ timesteps, so her modified utility is at least η: $\sum\limits_{t=1}^{\ell} \sum\limits_{p \in s_{i,t}} x_{pt} + 1 \geq \eta$. We then maximize η subject to these constraints. By relaxing the 0-1 variables x_{pt} to take values in $\mathbb{R}_+$, we obtain the following LP relaxation:

$$\begin{aligned}
\text{LP-E}{\small\text{GAL}} : \max \eta \qquad & \\
\sum_{p \in P} x_{pt} \leq 1 \qquad & \text{for all } t \in [\ell] \\
\sum_{t=1}^{\ell} \sum_{p \in s_{i,t}} x_{pt} \geq \eta - 1 \qquad & \text{for all } i \in N \\
x_{pt} \geq 0 \qquad & \text{for all } p \in P, t \in [\ell].
\end{aligned}$$

Let η^* be the optimal (fractional) value of LP-EGAL, and let $(\{x^*_{pt}\}_{p \in P, t \in [\ell]}, \eta^*)$ be the associated fractional solution. Let η' be the optimal egalitarian welfare with respect to $u'_1, \ldots, u'_n$ in our instance. Then η' together with an encoding of the outcome that provides this welfare forms a feasible solution to LP-EGAL, and hence $\eta' \leq \eta^*$.

For every outcome $\mathbf{o}$ we have $u'_i(\mathbf{o}) = u_i(\mathbf{o}) + 1 \geq 1$ for all $i \in N$. Hence, if $\eta^* \leq 4 \ln n$, it holds that every outcome $\mathbf{o}$ is a $\frac{1}{4 \ln n}$-approximation. Thus, we can output an arbitrary outcome in this case. Therefore, from now we assume that $\eta^* > 4 \ln n$.

To transform $\{x^*_{pt}\}_{p \in P, t \in [\ell]}$ into a feasible integer solution, we use randomized rounding: for each $t \in [\ell]$ we set $o_t = p$ with probability x^*_{pt}. These choices are independent across timesteps. For each $i \in N$ and $t \in [\ell]$, define a Bernoulli random variable Z^t_i that indicates whether agent i approves the project randomly selected at timestep t. Then, for each agent $i \in N$ let $Z_i = \sum_{t=1}^{\ell} Z^t_i$, so that the utility of agent i is given by $u'_i = Z_i + 1$. Then, the expected value of Z^t_i is

$$\mathbb{E}[Z^t_i] = \sum_{p \in s_{i,t}} x^*_{pt}.$$

By linearity of expectation, we derive

$$\mathbb{E}[Z_i] = \sum_{t=1}^{\ell} \mathbb{E}[Z^t_i] = \sum_{t=1}^{\ell} \sum_{p \in s_{i,t}} x^*_{pt} \geq \eta^* - 1.$$

Applying the multiplicative Chernoff bound [Alon and Spencer, 2016], we obtain

$$\mathbb{P}\{u'_i \leq \eta^*(1-\delta)\} \leq \exp\left(\frac{-\eta^* \delta^2}{2}\right) \text{ for every } \delta > 0.$$

Recall that $\eta^* > 4 \ln n$. Thus, by letting $\delta = \frac{4}{5}$, we have

$$\mathbb{P}\left\{u'_i \leq \frac{\eta^*}{5}\right\} \leq \exp\left(-\frac{32 \ln n}{25}\right) = n^{-\frac{32}{25}}.$$

Finally, by applying the union bound, we get

$$\mathbb{P}\left\{u'_i \geq \frac{\eta^*}{5}\text{for all } i \in N\right\} \geq 1 - n \cdot n^{-\frac{32}{25}} \qquad = 1 - n^{-\frac{7}{25}} > 0.$$

Consequently, with positive probability we obtain an integer solution whose egalitarian welfare (with respect to the modified utilities) is at least $\frac{1}{5}$ of the optimal egalitarian welfare. Using probability amplification techniques, we can find some such solution with probability $1 - \varepsilon$. It remains to observe that $\frac{1}{5} > \frac{1}{4 \ln n}$ when $n > 3$. $\square$

6.4 Strategyproofness and Non-Obvious Manipulability

An important consideration in the context of collective decision-making is *strategyproofness*: no agent should be able to increase their utility by misreporting their preferences. It is formally defined as follows. Note that agent i's utility function u_i is computed with respect to her (truthful) approval vector $\mathbf{s}_i$.

Definition 6.4.1 (Strategyproofness)**.** For each $i \in N$, let $\mathcal{S}_{-i}$ denote the list of all approval vectors except that of agent i: $\mathcal{S}_{-i} = (\mathbf{s}_1, \dots, \mathbf{s}_{i-1}, \mathbf{s}_{i+1}, \dots, \mathbf{s}_n)$. A mechanism $\mathcal{M}$ is *strategyproof* (SP) if for each instance $(P, N, \ell, (\mathbf{s}_i)_{i \in N})$, each agent $i \in N$ and each approval vector $\mathbf{B}_i$ it holds that $u_i(\mathcal{M}(\mathcal{S}_{-i}, \mathbf{s}_i)) \geq u_i(\mathcal{M}(\mathcal{S}_{-i}, \mathbf{B}_i))$.

Consider the mechanism that outputs a UTIL outcome by choosing a project with the highest number of approvals at each timestep (breaking ties lexicographically); we will refer to this mechanism as GREEDYUTIL. We observe that this mechanism is strategyproof.

Theorem 6.4.2. GREEDYUTIL *is strategyproof.*

Proof. Consider an agent $i \in N$ and a timestep $t \in [\ell]$. Since GREEDYUTIL makes a decision for each timestep independently of agents' reports regarding other timesteps, it suffices to argue that i cannot increase her utility at timestep t by reporting an approval set $s \neq s_{i,t}$. This follows directly from the fact that Approval Voting is strategyproof in the single-winner setting (see, e.g., Brams and Fishburn [1983]); for completeness, we provide a direct proof.

Suppose that when i truthfully reports $s_{i,t}$ at timestep t, GREEDYUTIL set $o_t = p$. If $p \in s_{i,t}$, agent i cannot benefit from misreporting at timestep t, so assume that $p \notin s_{i,t}$. Then for every project $q \in s_{i,t}$ it holds that either q gets fewer approvals than p at timestep t, or p and q receive the same number of approvals, but p precedes q in the tie-breaking order. Agent i cannot change this by modifying her report, so she cannot increase her utility at timestep t. As this holds for every $t \in [\ell]$, the proof is complete. □

In contrast, no deterministic mechanism that maximizes egalitarian welfare can be strategyproof. Intuitively, this is because agents have an incentive to not report their approval for projects that are popular among other agents.

Proposition 6.4.3. *Let $\mathcal{M}$ be a deterministic mechanism that always outputs an* EGAL *outcome. Then $\mathcal{M}$ is not strategyproof, even in the* UP *setting.*

Proof. Consider an instance with $P = \{p_1, p_2, p_3, p_4\}$, $n = 3$, $\ell = 2$, and the approval sets $\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3$ such that $s_{1,1} = s_{2,1} = s_{3,1} = \{p_1\}$ and $s_{i,2} = \{p_i\}$ for each $i \in \{1, 2, 3\}$.

Consider an outcome $\mathbf{o} = (o_1, o_2)$. If $o_1 \neq p_1$, at most one agent receives positive utility from $\mathbf{o}$, so the egalitarian welfare of $\mathbf{o}$ is 0. In contrast, if $o_1 = p_1$, the egalitarian welfare of $\mathbf{o}$ is at least 1. Thus, when agents report truthfully, $\mathcal{M}$ outputs an outcome $\mathbf{o}^* = (o_1^*, o_2^*)$ with $o_1^* = p_1$.

Assume without loss of generality that $o_2^* = p_2$. Then $u_1(\mathbf{o}^*) = 1$. Now, suppose that agent 1 misreports their approval vector as $\mathbf{s}'_1 = (s'_{11}, s'_{12})$, where $s'_{11} = \{p_4\}$, $s'_{12} = \{p_1\}$.

In this case, the only EGAL outcome for the reported preferences is $\mathbf{o}' = (p_1, p_1)$, so $\mathcal{M}$ is forced to output $\mathbf{o}'$. Moreover, agent 1's utility (with respect to their true preferences) from $\mathbf{o}'$ is $u_1(\mathbf{o}') = 2 > u_1(\mathbf{o}^*)$, i.e., agent 1 has an incentive to misreport. □

Having ruled out compatibility of EGAL and strategyproofness, we consider a relaxation of strategyproofness known as *non-obvious manipulability.* It was introduced by Troyan and Morrill [2020], and has been studied in the single-round multiwinner voting literature [Arribillaga and Bonifacio, 2024, Aziz and Lam, 2021]. It is formally defined as follows.

Definition 6.4.4 (Non-Obvious Manipulability)**.** A mechanism $\mathcal{M}$ is *not obviously manipulable (NOM)* if for every instance $(P, N, \ell, (\mathbf{s}_i)_{i \in N})$, each agent $i \in N$, and each approval vector $\mathbf{B}_i$ that i may report, the following conditions hold:

$$\min_{\int_{-i} \in \Sigma^{n-1}_{P,\ell}} u_i(\mathcal{M}(\mathcal{S}_{-i}, \mathbf{s}_i)) \geq \min_{\mathcal{S}_{-i} \in \Sigma^{n-1}_{P,\ell}} u_i(\mathcal{M}(\mathcal{S}_{-i}, \mathbf{B}_i))$$
$$\max_{\mathcal{S}_{-i} \in \Sigma^{n-1}_{P,\ell}} u_i(\mathcal{M}(\mathcal{S}_{-i}, \mathbf{s}_i)) \geq \max_{\mathcal{S}_{-i} \in \Sigma^{n-1}_{P,\ell}} u_i(\mathcal{M}(\mathcal{S}_{-i}, \mathbf{B}_i)),$$

where $\Sigma_{P,\ell}^{n-1}$ denotes the space of all $(n-1)$-voter profiles where each voter expresses her approvals of projects in P over ℓ steps.

Intuitively, a mechanism is NOM if an agent cannot increase her worst-case utility or her best-case utility (with respect to her true utility function) by misreporting. Clearly, strategyproofness implies NOM: if a mechanism is strategyproof, no agent can increase her utility in *any* case by misreporting.

While NOM is a much weaker condition than strategyproofness, it turns out that it is still incompatible with EGAL.

Proposition 6.4.5. *Let $\mathcal{M}$ be a deterministic mechanism that always outputs an* EGAL *outcome. Then $\mathcal{M}$ is not NOM.*

Proof. We will prove that an agent can increase her worst-case utility, and hence the mechanism fails NOM.

Fix $P = \{p_1, p_2\}$, $n = \ell = 2$. Consider first an instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$ with $s_{1,1} = s_{2,1} = \varnothing$, $s_{1,2} = \{p_1\}$, and $s_{2,2} = \{p_2\}$. Let $\mathbf{o} = (o_1, o_2)$ be the output of $\mathcal{M}$ on this instance; assume without loss of generality that $o_2 = p_1$.

Now, consider an instance $\mathcal{I}' = (P, N, \ell, (\mathbf{s}'_i)_{i \in N})$ with $s'_{11} = \{p_1, p_2\}$, $s'_{21} = \varnothing$, $s'_{12} = \{p_1\}$, and $s'_{22} = \{p_2\}$. For $\mathbf{o}' = (o'_1, o'_2)$ to be an EGAL outcome for this instance, it has to provide positive utility to both agents; this is only possible if $o'_2 = p_2$. Thus, the utility of agent 1 from the outcome selected by $\mathcal{M}$ is 1.

However, we will now argue that if the first agent misreports her approval vector as $(\varnothing, \{p_1\})$, she is guaranteed utility 2 no matter what the second agent reports, i.e., her worst-case utility is 2.

Indeed, suppose agent 2 reports $(\varnothing, \{p_2\})$, in which case the agents have the same approval vectors as in $\mathcal{I}$. We assumed that on $\mathcal{I}$ mechanism $\mathcal{M}$ selects p_1 at timestep 2 (and one of p_1, p_2 at timestep 1), so the utility of agent 1 (with respect to her true preferences in $\mathcal{I}'$) is 2. On the other hand, if agent 2 reports $(\varnothing, \{p_1\})$ or $(\varnothing, \{p_1, p_2\})$, then $\mathcal{M}$ selects p_1 at timestep 2 (and one of p_1, p_2 at timestep 1), as this is the only way to guarantee positive utility to both agents. Similarly, if $s_{2,1} \neq \varnothing$, there is a way to provide positive utility to both agents, by selecting a project from $s_{2,1}$ at the first timestep, and p_1 at the second timestep. However, for agent 1 to obtain positive utility it is necessary to select p_1 at the second timestep, so the mechanism is forced to do so. That is, if agent 1 reports $(\varnothing, \{p_1\})$, $\mathcal{M}$ selects an outcome (o_1^*, o_2^*) with $o_2^* = p_1$, and this outcome provides utility 2 to agent 1 according to their preferences in $\mathcal{I}'$. □

However, we obtain a positive result for the CP setting. Let $\mathcal{M}_{\text{lex}}$ be the mechanism that outputs an EGAL outcome, breaking ties in favor of agents with lower indices.

Formally, we define a partial order $\succ_{\text{lex}}$ on the set $\Pi(\mathcal{I})$ of possible outcomes for a given instance $\mathcal{I}$ as follows: (1) if $\text{EGAL}(\mathbf{o}) > \text{EGAL}(\mathbf{o}')$, then $\mathbf{o} \succ_{\text{lex}} \mathbf{o}'$; (2) if $\text{EGAL}(\mathbf{o}) = \text{EGAL}(\mathbf{o}')$ and there is an $i \in N$ such that $u_{i'}(\mathbf{o}) = u_{i'}(\mathbf{o}')$ for $i' < i$ and $u_i(\mathbf{o}) > u_i(\mathbf{o}')$ then $\mathbf{o} \succ_{\text{lex}} \mathbf{o}'$. Note that two outcomes are incomparable under $\succ_{\text{lex}}$ if and only if they provide the same utility to all agents; we say that such outcomes are *utility-equivalent*, and complete $\succ_{\text{lex}}$ to a total order $\succ$ on $\Pi(\mathcal{I})$ arbitrarily. $\mathcal{M}_{\text{lex}}$ outputs an outcome $\mathbf{o}$ with $\mathbf{o} \succ \mathbf{o}'$ for all $\mathbf{o}' \in \Pi(\mathcal{I}) \setminus \{\mathbf{o}\}$.

Theorem 6.4.6. *$\mathcal{M}_{lex}$ is* NOM *in the* CP *setting.*

Proof. In the CP setting, the best-case utility of each agent when they report truthfully is ℓ: this is achieved, e.g., if all other agents have the same preferences. Thus, it remains to establish that under $\mathcal{M}_{\text{lex}}$ no agent can improve their worst-case utility by misreporting.

Let $\mathbf{s}_i$ be the true approval vector of agent i, and let $\mathbf{B}_i$ be another approval vector that i may report. Consider a minimum-length sequence of elementary operations that transforms $\mathbf{s}_i$ into $\mathbf{B}_i$ by first adding approvals in $B_{it} \setminus s_{i,t}$, $t \in [\ell]$, one by one, and then removing approvals in $s_{i,t} \setminus B_{it}$, $t \in [\ell]$, one by one. Let $\mathbf{X}^0, \mathbf{X}^1, \dots, \mathbf{X}^k, \dots, \mathbf{X}^{\gamma+1}$ be the resulting sequence of approval vectors, with $\mathbf{X}^0 = \mathbf{s}_i$, $\mathbf{X}^{\gamma+1} = \mathbf{B}_i$. Note that, since we add approvals first, all entries of each approval vector in this sequence are non-empty subsets of P, i.e., we remain in the CP setting. Suppose this sequence starts with k additions, so that $\mathbf{X}^s$ is obtained from $\mathbf{X}^{s-1}$ by adding a single approval if $s \le k$ and by deleting a single approval if $s > k$.

We will first show that, for any fixed list $\mathcal{S}_{-i}$ of other agents' approval vectors, reporting $\mathbf{X}^k$ instead of $\mathbf{X}^0 = \mathbf{s}_i$ does not increase i's utility; this implies that reporting $\mathbf{X}^k$ instead of $\mathbf{X}^0$ does not increase i's worst-case utility. Then, we will show that for all $s = k+1, \dots, \gamma+1$ reporting $\mathbf{X}^s$ instead of $\mathbf{X}^{s-1}$ does not increase i's worst-case utility either. This implies that reporting $\mathbf{B}_i$ instead of $\mathbf{s}_i$ does not increase i's worst-case utility.

Fix a list $\mathcal{S}_{-i}$ of other agents' approval vectors, and let $\mathcal{S} = (\mathcal{S}_{-i}, \mathbf{s}_i)$, $\mathcal{S}' = (\mathcal{S}_{-i}, \mathbf{X}^k)$. Set $\mathcal{M}_{\text{lex}}(\mathcal{S}) = \mathbf{o}$, $\mathcal{M}_{\text{lex}}(\mathcal{S}') = \mathbf{o}'$. If $\mathbf{o}'$ and $\mathbf{o}$ are utility-equivalent at $\mathcal{S}$, we are done, as this means that i does not benefit from reporting $\mathbf{X}^k$ instead of $\mathbf{s}_i$; hence, assume that this is not the case. Let η be the egalitarian welfare of $\mathbf{o}'$ with respect to the reported utilities at $\mathcal{S}'$, and let η_i be the utility of i at $\mathbf{o}'$ according to

$\mathbf{X}^k$; note that $\eta_i \geq \eta$. Moreover, since $\mathbf{X}^k$ is obtained from $\mathbf{X}^0$ by adding approvals, it holds that $\eta_i \geq u_i(\mathbf{o}')$. By choosing $\mathbf{o}'$ at $\mathcal{S}$, the mechanism can guarantee utility η to all agents other than i, and $u_i(\mathbf{o}') \leq \eta_i$ to i. If $u_i(\mathbf{o}') \leq \eta$, the egalitarian welfare of choosing $\mathbf{o}'$ at $\mathcal{S}$ is $u_i(\mathbf{o}')$, so under any EGAL outcome at $\mathcal{S}$ (in particular, under the outcome chosen by $\mathcal{M}_{\text{lex}}$) the utility of i is at least $u_i(\mathbf{o}')$; hence, in this case we are done.

Now, suppose $u_i(\mathbf{o}') > \eta$. Then the egalitarian welfare of $\mathbf{o}'$ at $\mathcal{S}$ is η, so the egalitarian welfare of $\mathbf{o}$ at $\mathcal{S}$ is at least η. Moreover, it cannot be strictly higher than η, because then the egalitarian welfare of $\mathbf{o}$ at $\mathcal{S}'$ according to the reported utilities would be strictly higher than η as well, a contradiction with $\mathcal{M}_{\text{lex}}$ outputting $\mathbf{o}'$ on $\mathcal{S}'$. Thus, $\mathbf{o}'$ and $\mathbf{o}$ provide the same egalitarian welfare at $\mathcal{S}$. As we assumed that $\mathbf{o}'$ and $\mathbf{o}$ are not utility-equivalent, it has to be the case that $\mathcal{M}_{\text{lex}}$ chooses $\mathbf{o}$ over $\mathbf{o}'$ at $\mathcal{S}$ because of condition (2) in the definition of $\succ_{\text{lex}}$. Let j be the smallest index such that $u_j(\mathbf{o}) > u_j(\mathbf{o}')$. If $j \geq i$, we are done, as this means that $u_i(\mathbf{o}) = u_i(\mathbf{o}')$, so i does not benefit from reporting $\mathbf{X}^k$ instead of $\mathbf{s}_i$. Now, suppose that $j < i$. Note that $\mathbf{o}'$ and $\mathbf{o}$ provide the same egalitarian welfare at $\mathcal{S}'$, and agents $1, \ldots, j$ have the same preferences in $\mathcal{S}'$ and $\mathcal{S}$. Hence, $\mathcal{M}_{\text{lex}}$ should choose $\mathbf{o}$ over $\mathbf{o}'$ at $\mathcal{S}'$, a contradiction with the choice of $\mathbf{o}'$.

We will now consider $s > k$ and argue that for every $\mathcal{S}_{-i}$ there is an $\mathcal{S}'_{-i}$ such that $u_i(\mathcal{M}_{\text{lex}}(\mathcal{S}_{-i}, \mathbf{X}^{s-1})) \geq u_i(\mathcal{M}_{\text{lex}}(\mathcal{S}'_{-i}, \mathbf{X}^{s}))$. Suppose that $\mathbf{X}^s$ is obtained from $\mathbf{X}^{s-1}$ by deleting a project p at timestep t. Let $\mathbf{o} = \mathcal{M}_{\text{lex}}(\mathcal{S}_{-i}, \mathbf{X}^{s-1})$. If $o_t \neq p$ then the outcome $\mathcal{M}_{\text{lex}}(\mathcal{S}_{-i}, \mathbf{X}^{s})$ is utility-equivalent to $\mathbf{o}$. Otherwise, consider a project $p' \neq p$ approved by agent i at timestep t according to $\mathbf{X}^s$. We construct $\mathcal{S}'_{-i}$ by swapping all other agents' approvals for p and p' at timestep t. Consider an outcome $\mathbf{o}'$ that selects p' a timestep t and coincides with $\mathbf{o}$ at all other timesteps.

For each agent $j \in N$ it holds that the utility of j from $\mathbf{o}$ at $(\mathcal{S}_{-i}, \mathbf{X}^{s-1})$ is the same as her utility from $\mathbf{o}'$ at $(\mathcal{S}'_{-i}, \mathbf{X}^{s})$. We claim that this implies that when we execute $\mathcal{M}_{\text{lex}}$ on $(\mathcal{S}'_{-i}, \mathbf{X}^{s})$, it chooses an outcome that is utility-equivalent to $\mathbf{o}'$. Indeed, suppose there is an outcome $\widetilde{\mathbf{o}'}$ such that at $(\mathcal{S}'_{-i}, \mathbf{X}^{s})$ it holds that $\widetilde{\mathbf{o}'} \succ_{\text{lex}} \mathbf{o}'$. Consider at outcome $\widetilde{\mathbf{o}}$ that is identical to $\widetilde{\mathbf{o}'}$ except that if $\widetilde{\mathbf{o}'}$ chooses p' at t then $\widetilde{\mathbf{o}}$ chooses p at t. Then $\widetilde{\mathbf{o}} \succ_{\text{lex}} \mathbf{o}$, a contradiction with our choice of $\mathbf{o}$.

On the other hand, as agent i approves p, we have $u_i(\mathbf{o}) \geq u_i(\mathbf{o}')$. Hence, for each $\mathcal{S}_{-i}$ there is an $\mathcal{S}'_{-i}$ such that $u_i(\mathcal{M}_{\text{lex}}(\mathcal{S}_{-i}, \mathbf{X}^{s-1})) \geq u_i(\mathcal{M}_{\text{lex}}(\mathcal{S}'_{-i}, \mathbf{X}^{s}))$, i.e., reporting $\mathbf{X}^s$ instead of $\mathbf{X}^{s-1}$ does not increase i's worst-case utility. As this holds for all $s > k$, the proof is complete. □

6.5 Proportionality

Another property that may be desirable in the context of temporal voting (and has been considered by others in similar settings [Conitzer et al., 2017, Elkind et al., 2022]) is *proportionality* (PROP).

Definition 6.5.1 (Proportionality)**.** Given an instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i\in N})$, for each $i \in N$ let $\mu_i = |\{t \in [\ell] : s_{i,t} \neq \varnothing\}|$. We say that an outcome $\mathbf{o}$ is *proportional (PROP)* for $\mathcal{I}$ if for all $i \in N$ it holds that $u_i(\mathbf{o}) \geq \lfloor \frac{\mu_i}{n} \rfloor$.

We note that proportionality is often understood as guaranteeing each agent at least $1/n$-th of her maximum utility, which would correspond to using $\frac{\mu_i}{n}$ instead of $\lfloor \frac{\mu_i}{n} \rfloor$ in the right-hand side of our definition [Conitzer et al., 2017, Elkind et al., 2022, Igarashi et al., 2024]. However, the requirement that $u_i(\mathbf{o}) \geq \frac{\mu_i}{n}$ may be impossible to satisfy: e.g. if $N = \{1, 2\}$, $\ell = 3$, $P = \{p_1, p_2\}$ and for $i = 1, 2$ agent i approves project p_i at each timestep, we cannot simultaneously guarantee utility $3/2$ to both agents. Moreover, the proof of Theorem 6.3.2 can be used to show that deciding whether a given instance admits an outcome $\mathbf{o}$ with $u_i(\mathbf{o}) \geq \frac{\mu_i}{n}$ for all $i \in N$ is NP-complete.

Proposition 6.5.2. *Given an instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i\in N})$, it is* NP*-complete to decide whether there exists an outcome $\mathbf{o}$ such that $u_i(\mathbf{o}) \geq \mu_i/n$ for each $i \in N$. The hardness result holds even in the UP setting.*

Proof. The proof of Theorem 6.3.2 proceeds by a reduction from Vertex Cover. We can assume without loss of generality that the input instance $(G = (V, E), k)$ of Vertex Cover in that reduction satisfies $|V| \geq 3$ (as otherwise the problem is trivial) and that $|E| > |V|$; if the latter condition does not hold, we can modify our instance by adding a $2|V|$-vertex clique that is disjoint from the rest of the graph and increasing k by $2|V| - 1$ (as we need $2|V| - 1$ vertices to cover the clique), so that the modified graph has $3|V|$ vertices and at least $2|V|(2|V| - 1)/2 \geq 5|V| > 3|V|$ edges.

The reduction produces an instance of Egal-Dec with $n = |E| + |V| - k$ agents and $\ell = |V|$ timesteps, where all agents have unique preferences, so that $\mu_i = \ell$ for each $i \in N$. As $|E| > |V| \geq k$, we have $n > \ell$ and hence $0 < \mu_i < n$ for each $i \in N$. Consequently, an outcome $\mathbf{o}$ satisfies $u_i(\mathbf{o}) \geq \mu_i/n$ for each $i \in N$ if and only if $u_i(\mathbf{o}) \geq 1$ for each $i \in N$, and the proof of Theorem 6.3.2 shows that the latter condition is satisfied if and only if we start with a yes-instance of Vertex Cover. □

In contrast, our definition of proportionality can be satisfied by a simple greedy algorithm. This follows from a similar result obtained by Conitzer et al. [2017] in the setting of public decision-making.

Theorem 6.5.3. *A* PROP *outcome always exists and can be computed by a polynomial-time greedy algorithm.*

Proof. We describe a polynomial-time greedy algorithm for obtaining a PROP outcome.

We start by reordering the agents so that $\mu_1 \leq \mu_2 \leq \cdots \leq \mu_n$, and let $T = [\ell]$. We then process the agents in N one by one. For each $i \in N$ we pick a subset of timesteps $T_i \subseteq T$ with $|T_i| = \lfloor \frac{\mu_i}{n} \rfloor$ and $s_{i,t} \neq \varnothing$ for each $t \in T_i$. Then for each $t \in T_i$ we set $o_t = \{p\}$ for some $p \in s_{i,t}$; we then remove T_i from T. If T is still non-empty after all agents have been processed, we select the projects for timesteps in T arbitrarily; e.g., for each $t \in T$ we can select a project that receives the maximum number of approvals at t.

Clearly, if we manage to find an appropriate set of timesteps T_i for each $i \in N$, we obtain a proportional outcome. Thus, it remains to argue that such a set of timesteps can always be found. For an agent i, the number of projects selected before her turn is

$$\begin{aligned}\sum_{j \in [i-1]} \left\lfloor \frac{\mu_j}{n} \right\rfloor \leq \sum_{j \in [i-1]} \left\lfloor \frac{\mu_i}{n} \right\rfloor &= (i-1)\left\lfloor \frac{\mu_i}{n} \right\rfloor \\ &\leq (n-1)\left\lfloor \frac{\mu_i}{n} \right\rfloor \leq \frac{(n-1)\mu_i}{n}.\end{aligned}$$

As $\mu_i - \frac{(n-1)\mu_i}{n} = \frac{\mu_i}{n} \geq \lfloor \frac{\mu_i}{n} \rfloor$, it is indeed the case that we can find a set T_i of appropriate size. □

We note that PROP can be seen as a specialization of the *proportional justified representation* axiom for temporal voting [Chandak et al., 2024, Elkind et al., 2025c] to voter groups of size 1; this offers additional justification for our definition. Hence, the existence of PROP outcomes (and polynomial-time algorithms for computing them) also follows from Theorem 4.1 in the work of Chandak et al. [2024].

As finding *some* PROP outcome is not hard, one may wish to select the "best" PROP outcome. A natural criterion would be to pick a PROP outcome with the maximum utilitarian or egalitarian welfare. In particular, it would be useful to have an algorithm that can determine if there exists a PROP outcome that is also Util or Egal.

However, the proof of Proposition 6.5.2 implies that selecting a PROP outcome with maximum egalitarian welfare (among all outcomes) is computationally intractable. Our next result shows that combining proportionality with utilitarian welfare is hard, too, even though both finding a PROP outcome and finding a UTIL outcome is easy. It also implies that finding a utilitarian welfare-maximizing outcome among all PROP outcomes is NP-hard.

Theorem 6.5.4. *Determining if there exists a* PROP *outcome that is* UTIL *is* NP-*complete, even in the* CP *setting.*

Proof. To see that this problem is is NP, recall that we can compute the maximum utilitarian welfare for a given instance; thus, we can check if a given outcome is UTIL and PROP.

By Theorem 6.3.2, in the CP setting it is NP-complete even to determine if there is an outcome $\mathbf{o}$ such that $u_i(\mathbf{o}) \geq 1$ for all agents $i \in N$. Given an instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$ with complete preferences, we construct a new instance $\mathcal{I}'$ with complete preferences such that there is an outcome for $\mathcal{I}'$ that is proportional and maximizes utilitarian welfare if and only if there is an outcome for the original instance $\mathcal{I}$ that offers positive utility to all agents.

Note that, in the CP setting, if $\ell \geq n$, then there always exists an outcome $\mathbf{o}$ such that $u_i(\mathbf{o}) \geq 1$ for all $i \in N$. Hence, assume that $\ell < n$. We construct an instance $\mathcal{I}'$ with a set of agents $N' = [2n]$, $\ell' = 2n$ timesteps, and a set of projects $P' = P \cup \{q_1, \ldots, q_{n+1}\}$. For each $i = n+1, \ldots, 2n$ and each $t \in [\ell]$ let $P'_{it} = \{p \in P : n_{pt} \leq 2n - i\}$. We define the approval sets s'_{it} for $\mathcal{I}'$ as follows:

$$s'_{it} = \begin{cases} s_{i,t} & \text{if } i \leq n \text{ and } t \leq \ell \\ \{q_{i-n}\} \cup P'_{it} & \text{if } n < i \leq 2n \text{ and } t \leq \ell \\ \{q_i\} & \text{if } i \leq n \text{ and } \ell < t \leq 2n \\ \{q_{n+1}\} & \text{if } n < i \leq 2n \text{ and } \ell < t \leq 2n \end{cases}$$

Since all agents in $\mathcal{I}$ have CP preferences, this is also the case for $\mathcal{I}'$. Further, consider an arbitrary project $p \in P$. At timestep $t \in [\ell]$ this project is approved by n_{pt} agents in N as well as by each agent $i \in N' \setminus N$ such that $n < i \leq 2n - n_{pt}$, i.e., by $n - n_{pt}$ additional agents. Thus, in total, at timestep $t \in [\ell]$ each project $p \in P$ receives n approvals, whereas each project in $P' \setminus P$ receives at most one approval. On the other hand, in each of the last $2n - \ell$ timesteps q_{n+1} receives n approvals, whereas every other project receives at most one approval. It follows that an outcome $\mathbf{o}'$ for $\mathcal{I}'$ maximizes UTIL if and only if $o'_t \in P$ for each $t \in [\ell]$ and $o'_t = q_{n+1}$ for each $t > \ell$.

Moreover, since in $\mathcal{I}'$ the agents have complete preferences, $\mathbf{o}'$ is proportional if and only if each agent's utility is at least $\lfloor \frac{\ell'}{2n} \rfloor = 1$.

We now claim that there is an outcome $\mathbf{o}'$ that is proportional and maximizes UTIL for $\mathcal{I}'$ if and only if there is an outcome $\mathbf{o}$ for $\mathcal{I}$ such that $u_i(\mathbf{o}) \geq 1$ for all agents $i \in N$.

For the 'if' direction, suppose there is an outcome $\mathbf{o}$ for $\mathcal{I}$ such that $u_i(\mathbf{o}) \geq 1$ for all agents $i \in N$. We construct outcome $\mathbf{o}'$ by setting $o'_t = o_t$ for $t \in [\ell]$ and $o'_t = q_{n+1}$ for $t = \ell + 1, \ldots, 2n$. Our characterization of the UTIL outcomes implies that $\mathbf{o}'$ maximizes the utilitarian welfare. Moreover, for each agent $i \in [n]$ we have $u_i(\mathbf{o}') \geq 1$, as $\mathbf{o}'$ coincides with $\mathbf{o}$ for the first ℓ timesteps. For agents $i = n+1, \ldots, 2n$, we have $u_i(\mathbf{o}') \geq \ell$, as these agents approve the project q_{n+1} for the last $2n - \ell \geq \ell$ timesteps. Hence, $\mathbf{o}'$ achieves proportionality and maximizes UTIL.

For the 'only if' direction, suppose there exists an outcome $\mathbf{o}'$ for $\mathcal{I}'$ that achieves proportionality and maximizes UTIL. As $\mathbf{o}'$ achieves proportionality, we have $u_i(\mathbf{o}') \geq 1$ for all $i \in N$. Furthermore, as $\mathbf{o}'$ maximizes UTIL, we have $o'_t = q_{n+1}$ for $t > \ell$, so no agent in N derives positive utility from the last $2n - \ell$ timesteps. Hence, each agent $i \in N$ derives positive utility from one of the first ℓ timesteps. We construct the outcome $\mathbf{o}$ by setting $o_t = o'_t$ for $t \in [\ell]$. It then holds that $u_i(\mathbf{o}) \geq 1$ for all agents $i \in N$. □

We have established that simultaneously achieving optimal welfare and proportionality is computationally hard. A natural challenge, then, is to quantify the impact of proportionality on welfare. To address this challenge, we use the concepts of the *price of fairness* and the *strong price of fairness* [Bei et al., 2021, Bertsimas et al., 2011], which have been formulated for several proportionality guarantees in the single-round multiwinner voting literature [Brill and Peters, 2024, Elkind et al., 2024a, Lackner and Skowron, 2020]. As we focus on PROP in this paper, we instantiate the definition of the (strong) price of fairness accordingly, as follows.

Given a problem instance $\mathcal{I}$, let $\Pi_{\text{PROP}}(\mathcal{I}) \subseteq \Pi(\mathcal{I})$ denote the set of all proportional outcomes for $\mathcal{I}$.

Furthermore, given a welfare objective $W \in \{\text{EGAL}, \text{UTIL}\}$, let W-OPT$(\mathcal{I})$ denote the maximum W-welfare over all outcomes in $\Pi(\mathcal{I})$.

Definition 6.5.5 (Price of Proportionality)**.** For a welfare objective W, the *price of proportionality (PoPROP$_W$) for an instance* $\mathcal{I}$ is the ratio between the maximum W-welfare of an outcome for $\mathcal{I}$ and the maximum W-welfare of an outcome for $\mathcal{I}$ that satisfies *PROP*; the *price of proportionality for W (PoPROP$_W$)* is the supremum of *PoPROP*$_W(\mathcal{I})$ over all instances $\mathcal{I}$:

$$\begin{aligned} \textit{PoPROP}_W(\mathcal{I}) &= \frac{W\text{-}\textit{OPT}(\mathcal{I})}{\max_{\mathbf{o}\in\Pi_{PROP}(\mathcal{I})} W(\mathbf{o})}; \\ \textit{PoPROP}_W &= \sup_{\mathcal{I}} \textit{PoPROP}_W(\mathcal{I}). \end{aligned}$$

Definition 6.5.6 (Strong Price of Proportionality)**.** For a welfare objective W, the *strong price of proportionality (s-PoPROP$_W$) for an instance $\mathcal{I}$* is the ratio between the maximum W-welfare of an outcome for $\mathcal{I}$ and the minimum W-welfare of an outcome for $\mathcal{I}$ that satisfies *PROP*; the *strong price of proportionality for W (s-PoPROP$_W$)* is the supremum of *s-PoPROP*$(\mathcal{I})$ over all instances $\mathcal{I}$:

$$\begin{aligned} \textit{s-PoPROP}_W(\mathcal{I}) &= \frac{W\text{-}\textit{OPT}(\mathcal{I})}{\min_{\mathbf{o}\in\Pi_{PROP}(\mathcal{I})} W(\mathbf{o})}; \\ \textit{s-PoPROP}_W &= \sup_{\mathcal{I}} \textit{s-PoPROP}_W(\mathcal{I}). \end{aligned}$$

We first observe that requiring proportionality has no impact on egalitarian welfare: any outcome $\mathbf{o}$ can be transformed (in polynomial time) into a proportional outcome $\mathbf{o}'$ so that the egalitarian welfare of $\mathbf{o}'$ is at least as high as that of $\mathbf{o}$.

Proposition 6.5.7. *Given an outcome $\mathbf{o}$, we can construct in polynomial time another outcome $\mathbf{o}'$ such that $\mathbf{o}'$ is proportional and* $\textsc{Egal}(\mathbf{o}') \geq \textsc{Egal}(\mathbf{o})$.

Proof. Consider an outcome $\mathbf{o}$. If $\mathbf{o}$ satisfies PROP, we are done, so assume that this is not the case. Let $\lambda = \textsc{Egal}(\mathbf{o})$. Reorder the agents in N so that $\mu_i \leq \mu_{i'}$ for $1 \leq i < i' \leq n$. Since $\mathbf{o}$ fails PROP, there exists an $i \in N$ such that $u_i(\mathbf{o}) < \lfloor \frac{\mu_i}{n} \rfloor$; let q be the smallest value of i for which this is the case.

We will construct an outcome $\mathbf{o}'$ that is proportional and whose egalitarian welfare is at least as high as that of $\mathbf{o}$, i.e., $\mathbf{o}'$ satisfies $u_i(\mathbf{o}') \geq \max\{\lambda, \lfloor \frac{\mu_i}{n} \rfloor\}$ for all $i \in N$. To this end, we proceed in two stages.

Note first that for each agent $i \in [q-1]$ we have $u_i(\mathbf{o}) \geq \lfloor \frac{\mu_i}{n} \rfloor$ (by our choice of q) and $u_i(\mathbf{o}) \geq \lambda$ (by the definition of λ). Thus, for each $i \in [q-1]$ there are at least $\max\{\lambda, \lfloor \frac{\mu_i}{n} \rfloor\}$ timesteps $t \in [\ell]$ in which i approves o_t. During the first stage, we ask each agent $i \in [q-1]$ to mark $\max\{\lambda, \lfloor \frac{\mu_i}{n} \rfloor\}$ timesteps $t \in [\ell]$ with $o_t \in s_{i,t}$; multiple agents are allowed to mark the same timestep. We have $\lambda \leq u_q(\mathbf{o}) < \lfloor \frac{\mu_q}{n} \rfloor$ (by our choice of λ and q) and $\lfloor \frac{\mu_i}{n} \rfloor \leq \lfloor \frac{\mu_q}{n} \rfloor$ for $i < q$, so the total number of timesteps marked during the first stage does not exceed $(q-1)\lfloor \frac{\mu_q}{n} \rfloor$. Then, for each marked timestep t we set $o'_t = o_t$.

During the second stage, we proceed greedily, just as in the proof of Theorem 6.5.3. That is, for each $i = q, \ldots, n$ we ask agent i to mark $\lfloor \frac{\mu_i}{n} \rfloor$ previously unmarked timesteps t with $s_{i,t} \neq \varnothing$; for each such timestep t we pick a project $p \in s_{i,t}$ and set $o'_t = p$. To see why each agent i with $i \geq q$ can find $\lfloor \frac{\mu_i}{n} \rfloor$ suitable timesteps, note that her predecessors have marked at most

$$(q-1)\left\lfloor \frac{\mu_q}{n} \right\rfloor + \sum_{i'=q}^{i-1} \left\lfloor \frac{\mu_{i'}}{n} \right\rfloor \leq (i-1)\left\lfloor \frac{\mu_i}{n} \right\rfloor \leq \mu_i - \left\lfloor \frac{\mu_i}{n} \right\rfloor$$

timesteps, so at least $\lfloor \frac{\mu_i}{n} \rfloor$ of the μ_i timesteps in which agent i approves some projects remain available to agent i.

By construction, for each $i < q$ we have $u_i(\mathbf{o}') \geq \max\{\lambda, \lfloor \frac{\mu_i}{n} \rfloor\}$, and for $i \geq q$ we have $u_i(\mathbf{o}') \geq \lfloor \frac{\mu_i}{n} \rfloor \geq \lfloor \frac{\mu_q}{n} \rfloor > u_q(\mathbf{o}) \geq \lambda$. Hence, $\mathbf{o}'$ satisfies PROP and guarantees utility at least λ to all agents. □

By applying Proposition 6.5.7 to an outcome $\mathbf{o}$ that maximizes the egalitarian welfare, we obtain the following corollary.

Corollary 6.5.8. $\mathrm{PoPROP}_{\textsc{Egal}} = 1$.

In contrast, for utilitarian welfare, the price of proportionality scales as $\sqrt{n}$, even in the CP setting.

Theorem 6.5.9. *In the* CP *setting,* $\mathrm{PoPROP}_{\textsc{Util}} = \frac{n}{2\sqrt{n}-1}$ *and hence* $\Theta(\sqrt{n})$. *The lower bound applies even in the* UP *setting.*

Proof. We first establish the lower bound. Given an integer $k \geq 2$, we construct an instance $\mathcal{I}$ with $n = k^2$ agents, a set of projects $P = \{p_0, p_1, \ldots, p_{n-k}\}$, and $\ell = n$. The agents have static preferences: their approval sets for all timesteps $t \in [\ell]$ are defined by

$$s_{i,t} = \begin{cases} \{p_i\} & \text{if } i \leq n-k \\ \{p_0\} & \text{otherwise.} \end{cases}$$

Note that in this instance agents have unique preferences. The (unique) $\textsc{Util}$ outcome chooses project p_0 for all timesteps. As k agents approve p_0, the utilitarian welfare of this outcome is nk. However, in order for an outcome to be proportional, each project must be selected at least once. Hence, for each $\mathbf{o} \in \Pi_{\mathrm{PROP}}(\mathcal{I})$ we have $\textsc{Util}(\mathbf{o}) \leq n - k + k^2$. Then, we obtain

$$\mathrm{PoPROP}_{\textsc{Util}}(\mathcal{I}) \geq \frac{nk}{n-k+k^2} = \frac{n \cdot \sqrt{n}}{2n - \sqrt{n}} = \frac{n}{2\sqrt{n}-1}.$$

To establish the upper bound, given an instance $\mathcal{I}$, we explicitly construct an outcome $\mathbf{o}$ such that $\textsc{Util}(\mathbf{o})$ is at least a $(\frac{2}{\sqrt{n}} - \frac{1}{n})$-fraction of the maximum utilitarian welfare for $\mathcal{I}$.

We first focus on the case $\ell = n$; later, we will explain how to extend our analysis to other values of ℓ. As we consider the CP setting, in this case proportionality means that for every agent $i \in N$ there should be at least one timestep where i approves the selected project.

Recall that n_{pt} denotes the number of agents approving project p at timestep t, and let $k = \max_{(p,t)\in P\times[\ell]} n_{pt}$. We construct an outcome $\mathbf{o}$ in two stages.

In the first stage, we start by setting $T_1 = [\ell]$. Then, for $j = 1, \dots, k$ we pick a pair $(p_j, t_j) \in \arg\max_{(p,t)\in P\times T_j} n_{pt}$, and set $o_{t_j} = p_j$ and $T_{j+1} = T_j \setminus \{t_j\}$. This concludes the first stage.

Note that during the first stage we select projects for $k \geq 1$ timesteps, so $|T_{k+1}| = n - k$ timesteps remain available. Further, at least $k = n_{p_1 t_1}$ agents approve $p_1 = o_{t_1}$ at timestep t_1, so there are at most $n - k$ agents who obtain utility 0 from the partial outcome constructed in the first stage. During the second stage, we allocate each such agent i a distinct timestep $t \in T_{k+1}$, and set o_t to a project in $s_{i,t}$ (recall that the agents have complete preferences, so $s_{i,t} \neq \varnothing$); this is feasible as we have at most $n - k$ agents unsatisfied by stage 1, and $n - k$ available timesteps.

By construction, each agent's utility from $\mathbf{o}$ is at least 1, so $\mathbf{o}$ is proportional. Furthermore, for each $j \in [k]$ project p_j is among the projects that receive the highest number of approvals at timestep t_j, i.e., our selections during the first stage maximize the utilitarian welfare for timesteps $t_1 \dots, t_k$. Let $U = \sum_{j\in[k]} n_{p_j t_j}$ be the utilitarian welfare obtained from these timesteps.

We are now ready to bound $\textsc{Util}(\mathbf{o})$ and the maximum utilitarian welfare in our instance. First, note that $\textsc{Util}(\mathbf{o}) \geq U + (n - k)$: we obtain U from the first stage, and each timestep assigned in the second stage contributes at least 1. Further, let $s = n_{p_k t_k}$, i.e., s is the number of approvals received by project p_k in timestep t_k. Since during the first stage we sequentially selected pairs (p, t) so as to maximize n_{pt}, we have $U \geq sk$. Moreover, for each timestep $t \in T_{k+1}$ (i.e., a timestep for which we did not assign a project during the first stage) we have $\max_{p\in P} n_{pt} \leq s$, so these timesteps can contribute at most $s(n - k)$ to the utilitarian welfare, no matter which projects we select. Thus, the maximum utilitarian welfare for our instance is at most $U + s(n - k)$. Therefore, we have

$$\text{PoPROP}_{\textsc{Util}}(\mathcal{I}) \leq \frac{U + s(n-k)}{U + (n-k)}.$$

Since $U \geq sk$, $s \geq 1$, and the function $\frac{x+A}{x+B}$ is monotonically non-increasing for $A \geq B$, we obtain

$$\mathrm{PoPROP}_{\textsc{Util}}(\mathcal{I}) \leq \frac{sk + s(n-k)}{sk + (n-k)} = \frac{sn}{k(s-1)+n}.$$

Further, since $s = n_{p_k t_k} \leq n_{p_1 t_1} = k$ and $s - 1 \geq 0$, we have

$$\mathrm{PoPROP}_{\textsc{Util}}(\mathcal{I}) \leq \frac{sn}{k(s-1)+n} \leq \frac{sn}{s(s-1)+n}.$$

Applying the AM-GM inequality to s^2 and n, we obtain $s^2 + n \geq 2s\sqrt{n}$, so

$$\mathrm{PoPROP}_{\textsc{Util}}(\mathcal{I}) \leq \frac{sn}{s(s-1)+n} \leq \frac{sn}{2s\sqrt{n}-s} = \frac{n}{2\sqrt{n}-1}.$$

Thus, we have established the desired upper bound on $\mathrm{PoPROP}_{\textsc{Util}}$ for the case $n = \ell$. Now, if $\ell < n$, then $\frac{\mu_i}{n} < 1$ for all $i \in N$, so no outcome violates the proportionality constraint and therefore $\mathrm{PoPROP}_{\textsc{Util}} = 1$ in that case.

Finally, if $\ell > n$, let $q = \lfloor \frac{\ell}{n} \rfloor$ and let $r = \ell - q \cdot \lfloor \frac{\ell}{n} \rfloor$. We split the ℓ timesteps to create q groups $T^1, \dots, T^q$ of n timesteps each; if $r > 0$, we create an extra group T^{q+1} of size r. For each T^j, $j \in [q]$, we use the construction for the case $\ell = n$ to create a partial outcome $\mathbf{o}^j = (o_t)_{t \in T^j}$. For timesteps in T^{q+1}, we construct $\mathbf{o}^{q+1}$ by choosing projects that receive the highest number of approvals at each timestep. We then merge $\mathbf{o}^j$, $j \in [q+1]$, into a single outcome $\mathbf{o}$.

As each agent obtains positive utility from each $\mathbf{o}^j$, $j \in [q]$, the utility of each agent is at least $q = \lfloor \frac{\ell}{n} \rfloor$, so PROP is satisfied. Now, for each $j \in [q+1]$ let α_j be the maximum utilitarian welfare achievable for timesteps in T^j, and let $\beta_j = \textsc{Util}(\mathbf{o}^j)$; if $T^{q+1} = \varnothing$, we set $\alpha_{q+1} = \beta_{q+1} = 0$, with the convention that $\frac{0}{0} = 1$. We have argued that $\mathrm{PoPROP}_{\textsc{Util}} \leq \frac{n}{2\sqrt{n}-1}$ as long as $\ell = n$, so we have $\frac{\alpha_j}{\beta_j} \leq \frac{n}{2\sqrt{n}-1}$ for each $j \in [q]$; also, $\frac{\alpha_{q+1}}{\beta_{q+1}} = 1$ by construction of $\mathbf{o}^{q+1}$. It remains to observe that

$$\mathrm{PoPROP}_{\textsc{Util}}(\mathcal{I}) = \frac{\alpha_1 + \cdots + \alpha_{q+1}}{\beta_1 + \cdots + \beta_{q+1}} \leq \max_{j \in [q+1]} \frac{\alpha_j}{\beta_j} \leq \frac{n}{2\sqrt{n}-1}$$

and

$$\frac{n}{2\sqrt{n}} \leq \frac{n}{2\sqrt{n}-1} \leq \frac{n}{\sqrt{n}},$$

which implies $\mathrm{PoPROP}_{\textsc{Util}} = \Theta(\sqrt{n})$. □

To obtain bounds on s-PoPROP, we first prove a technical lemma.

Lemma 6.5.10. *For every $k, n \in \mathbb{N}$ such that $k \geq n$ it holds that $\lfloor \frac{k}{n} \rfloor \geq \frac{k}{2n-1}$.*

Proof. We proceed by case analysis.

- $n = 1$. In this case we have $\lfloor \frac{k}{n} \rfloor = k = \frac{k}{2n-1}$, so the statement of the lemma is true.

- $n \geq 2$, $n \leq k \leq 2n-1$. In this case we have $\lfloor \frac{k}{n} \rfloor = 1$ and $\frac{k}{2n-1} \leq 1$, so the statement of the lemma is true as well.

- $n \geq 2$, $2n \leq k \leq 3n-1$. We have $3n-1 \leq 4n-2$, so $\frac{k}{2n-1} \leq 2 = \lfloor \frac{k}{n} \rfloor$, which establishes the statement of the lemma for this case.

- $n \geq 2$, $k \geq 3n$. Since $n \geq 2$, we have $2n-1 \geq \frac{3n}{2}$ and hence $\frac{k}{2n-1} \leq \frac{2k}{3n}$. Moreover, $k \geq 3n$ implies $\frac{k}{3n} \geq 1$. Thus, we have

$$\left\lfloor \frac{k}{n} \right\rfloor \geq \frac{k}{n} - 1 \geq \frac{k}{n} - \frac{k}{3n} = \frac{2k}{3n} \geq \frac{k}{2n-1},$$

establishing the statement of the lemma for this case as well.

□

We are now ready to bound the strong price of proportionality.

Theorem 6.5.11. *We have* s-PoPROP$_{\text{UTIL}}$ = s-PoPROP$_{\text{EGAL}}$ $= +\infty$. *However, if for all agents $i \in N$ it holds that $\lfloor \frac{\mu_i}{n} \rfloor \geq 1$, then* s-PoPROP$_{\text{UTIL}}$ = s-PoPROP$_{\text{EGAL}}$ = $2n-1$.

Proof. We will first argue that for general preferences the strong price of proportionality is unbounded, both for UTIL and for EGAL. Consider an instance with $P = \{p, q\}$, $n \geq 2$ agents, and $\ell = 1$. For each $i \in N$, let $s_{i,1} = \{p\}$. Then, both possible outcomes $\mathbf{o} = (p)$ and $\mathbf{o}' = (q)$ satisfy PROP, but only $\mathbf{o} = (p)$ delivers positive utility. Hence, s-PoPROP$_{\text{UTIL}}$ = s-PoPROP$_{\text{EGAL}}$ $= +\infty$.

Next, suppose that $\lfloor \frac{\mu_i}{n} \rfloor \geq 1$ for all agents $i \in N$. We will show that s-PoPROP$_{\text{UTIL}}$ = s-PoPROP$_{\text{EGAL}}$ = $2n-1$.

For the upper bound, consider a PROP outcome $\mathbf{o}$. For each agent $i \in N$ we have $u_i(\mathbf{o}) \geq \lfloor \frac{\mu_i}{n} \rfloor$. Since $\lfloor \frac{\mu_i}{n} \rfloor \geq 1$ implies $\mu_i \geq n$, we can apply Lemma 6.5.10 to μ_i and n to obtain $\lfloor \frac{\mu_i}{n} \rfloor \geq \frac{\mu_i}{2n-1}$, so

$$\text{UTIL}(\mathbf{o}) = \sum_{i \in N} u_i(\mathbf{o}) \geq \frac{1}{2n-1} \sum_{i \in N} \mu_i,$$
$$\text{EGAL}(\mathbf{o}) = \min_{i \in N} u_i(\mathbf{o}) \geq \frac{1}{2n-1} \min_{i \in N} \mu_i.$$

Since the maximum utilitarian and egalitarian welfare do not exceed $\sum_{i \in N} \mu_i$ and $\min_{i \in N} \mu_i$, respectively, this establishes our upper bound.

To prove the lower bound, consider an instance with a set of n agents N, $P = \{p, q\}$, and $\ell = 2n-1$. For each $i \in N$ and $t \in [\ell]$, let $s_{i,t} = \{p\}$. For this instance the maximum utilitarian welfare and the maximum egalitarian welfare are, respectively, $n \cdot (2n-1)$ and $2n-1$, obtained by selecting p at every timestep. On the other hand, we have $1 \leq \frac{\mu_i}{n} < 2$ for all $i \in N$, so $\mathbf{o}'$ with $o_1 = p$, $o_t = q$ for $t = 2, \ldots, \ell$ satisfies PROP. This implies a lower bound of $2n-1$ for both s-PoPROP$_{\text{UTIL}}$ and s-PoPROP$_{\text{EGAL}}$. □

6.6 Extensions

So far, we have focused on UTIL and EGAL; however, some of our results extend to the entire space of welfare measures between these two extremes, namely, the family of *p-mean welfare* objectives. Formally, for each $p \in \mathbb{R}$, the *p-mean welfare* provided by an outcome $\mathbf{o}$ is given by $\left(\frac{1}{n}\sum_{i\in N} u_i(\mathbf{o})^p\right)^{1/p}$. The associated family of decision problems is defined as follows.

p-MEAN WELFARE:
Input: An instance $\mathcal{I} = (N, P, \ell, (\mathbf{s}_i)_{i\in N})$ and a parameter $\lambda \in \mathbb{Z}^+$.
Question: Is there an outcome $\mathbf{o}$ that satisfies $\left(\frac{1}{n}\sum_{i\in N} u_i(\mathbf{o})^p\right)^{1/p} \geq \lambda$?

Note that setting $p = 1$ (respectively, $p = -\infty$) corresponds to the utilitarian (respectively, egalitarian) welfare. Setting $p \to 0$ corresponds to the geometric mean, or Nash welfare, which we denote by NASH (with the corresponding decision problem denoted by NASH-DEC).

It can be verified that many of the computational hardness and impossibility results for EGAL directly translate to similar results for NASH: NASH is obviously manipulable in the general setting and not strategyproof even in the CP setting. Moreover, NASH-DEC is NP-complete even when $m = 2$ and is W[2]-hard with respect to ℓ.

Regarding positive results, the XP algorithm with respect to ℓ is based on enumerating all possible outcomes, and hence it works for all p-mean welfare measures. While our FPT algorithm (with respect to n) for EGAL relies on an ILP that does not extend to other welfare measures due to their non-linearity, a randomized XP algorithm has been recently proposed for a more demanding setting [Elkind et al., 2022]. In our model, we can show that there is a deterministic XP algorithm (with respect to n) for any *p-mean welfare* objective.

Theorem 6.6.1. *There exists a deterministic* XP *algorithm (with respect to n) that maximizes the p-mean welfare.*

Proof. We describe a dynamic programming algorithm that runs in time $\mathcal{O}((\ell+1)^{n+1} \cdot mn)$.

Fix an instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i\in N})$, and let $L = \{0, \dots, \ell\}$. Given an outcome $\mathbf{o}$ for $\mathcal{I}$, a timestep $t \in L$, and an agent $i \in N$, we write $u_i^t(\mathbf{o})$ to denote the utility obtained by agent i from the first t timesteps under $\mathbf{o}$: $u_i^t(\mathbf{o}) = |\{j \in [t] : o_j \in s_{i,j}\}|$. We collect these utilities in a vector $U^t(\mathbf{o}) = (u_i^t(\mathbf{o}))_{i\in N}$. Note that for each $t \in L$ and each $\mathbf{o} \in \Pi(\mathcal{I})$ we have $U^t(\mathbf{o}) \in L^n$.

For each vector $V \in L^n$ and each $t \in L$ let $Q[V, t] = 1$ if there exists an outcome $\mathbf{o} \in \Pi(\mathcal{I})$ such that $U^t(\mathbf{o}) = V$, and let $Q[V, t] = 0$ otherwise. We will now explain how to compute the quantities $(Q[V, t])_{V\in L^n, t\in L}$ using dynamic programming.

For the base case $t = 0$, we have

$$Q[V, 0] = \begin{cases} 1 & \text{if } V = (0, \dots, 0) \\ 0 & \text{otherwise.} \end{cases}$$

Now, suppose $t \geq 1$. Then for each $V \in L^n$ we have $Q[V, t] = 1$ if and only if there is a vector $Y \in L^n$ with $Q[Y, t-1] = 1$ and a project p such that for each $i \in N$ we have either (i) $p \in s_{i,t}$ and $Y_i + 1 = V_i$ or (ii) $p \notin s_{i,t}$ and $Y_i = V_i$. Thus, given the values of $Q[Y, t-1]$ for all $Y \in L^n$, we can compute the values $Q[V, t]$ for all $V \in L^n$; each value can be determined by iterating through all projects and all agents, i.e., in time $\mathcal{O}(mn)$.

As our dynamic programming table has $(\ell + 1)^{n+1}$ entries, it can be filled in time $\mathcal{O}((\ell+1)^{n+1}mn)$. Once it has been filled, we can scan through all vectors $V \in L^n$ such that $Q[V, \ell] = 1$ and find one with the highest p-mean welfare. An outcome $\mathbf{o}$ that provides this welfare can be found using standard dynamic programming techniques. □

However, it remains open whether there is an FPT algorithm with respect to n that maximizes Nash (or other p-mean welfare objectives with $p \neq 0, -\infty$).

6.7 Conclusion

We investigated the problem of maximizing utilitarian and egalitarian welfare for temporal elections. We showed that, while Util outcomes can be computed in polynomial time and can be achieved in a strategyproof manner, Egal is NP-complete and obviously manipulable. To circumvent the NP-hardness of EGAL, we analyzed its parameterized complexity with respect to n, m and ℓ, and provided an approximation algorithm that is based on randomized rounding. We also established the existence of a NOM mechanism for EGAL under a mild constraint on agents' preferences. Finally, we considered proportionality and showed that it is computationally hard to select the 'best' proportional outcome. We also gave upper and lower bounds on the (strong) price of proportionality with respect to both UTIL and EGAL.

We note that our upper bound on the price of proportionality with respect to UTIL only applies to the CP setting. It is easy to see that $\text{PoPROP}_{\text{UTIL}}$ does not exceed n even for general preferences: if there are ℓ' timesteps $t \in [\ell]$ with $\cup_{i \in N} s_{i,t} \neq \varnothing$, the utilitarian welfare does not exceed $\ell' n$, whereas selecting o_t from $\cup_{i \in N} s_{i,t}$ for each $t \in [\ell]$ results in an outcome $\mathbf{o}$ that satisfies $\text{UTIL}(\mathbf{o}) \geq \ell'$. However, obtaining non-trivial upper bounds on $\text{PoPROP}_{\text{UTIL}}$ in the general setting remains an open problem.

Some of our results extend to p-mean objectives other than EGAL and UTIL. However, it remains open which of these objectives admit polynomial-time algorithms.

In our model, agents are assumed to have approval preferences, which can be thought of as *binary utilities* over projects. One can extend this model to arbitrary *cardinal preferences*, by allowing each agent $i \in N$ to have a valuation function $v_i : P \times [\ell] \to \mathbb{R}_{[0,1]}$ instead of an approval set. Cardinal preferences have recently been studied in various social choice settings [Conitzer et al., 2017, Elkind et al., 2023, 2025a, Fain et al., 2018, Freeman et al., 2017]. It would be interesting to investigate whether the positive results for our model extend to the model that allows for arbitrary cardinal preferences.

Chapter 7

Proportional Representation in Temporal Voting

7.1 Overview

In the previous chapter, we looked at a notion of *proportionality* (Section 6.5), which aims to guarantees each agent their "proportional fair share". This concept is typically studied in fair division and public decision-making settings. Here, we shift focus to consider another form of proportionality commonly studied in other collective decision-making scenarios—*multiwinner voting.*Proportionality (or more precisely, *proportional representation*) in this context aims to guarantee that each group of agents should deserve a utility that commensurate (proportionally) with their group size and demands. We begin with a motivating example.

Consider a large corporation that has decided to improve its public image and to give back to the society by engaging in *corporate philanthropy* over the next decade. They will commit a small fraction of their profits towards supporting the efforts of a single charitable organization, to be selected on an annual basis. The management of this corporation decides to ask its customers, staff, and shareholders for input as to which charity organization it should select each year. Furthermore, the charity selected is of strategic importance—it would directly impact the company's corporate image and hence profitability. As such, it is important for the company to ensure that the selection is *representative* of what its customers, staff, and shareholders care about and that it would create maximum impact for the charity organization they choose to support.

It is natural to view this problem through the lens of multiwinner voting [Faliszewski et al., 2017, Elkind et al., 2017, Lackner and Skowron, 2023]: indeed, the corporation's goal is to select a fixed-size subset of *candidates* (in this case, charities) while respecting the preferences of *agents* (in this case, the customers, staff, and shareholders). In this context, several notions of representation and fairness have been proposed over the past decade, spanning across proportional representation [Elkind et al., 2017, Aziz and Lee, 2020], diversity [Bredereck et al., 2018, Celis et al., 2018], and excellence, amongst others [Lackner and Skowron, 2023]. Perhaps the most prominent among these is the concept of *justified representation (JR)* and its variants (such as proportional justified representation (PJR) and extended justified representation (EJR)), which aim to capture the idea that large cohesive groups of voters should be fairly represented in the final outcome [Aziz et al., 2017, Sánchez-Fernández et al., 2017, Aziz et al., 2018, Peters et al., 2021, Brill and Peters, 2023].

However, the existing notions of fairness do not fully capture the complexity of our setting: traditional multiwinner voting models consider decisions made in a single round, with the entire set of candidates to fund (or candidates) being chosen simultaneously. In contrast, in our model the decisions are made over time, voters' preferences may evolve, and a candidate may be chosen multiple times. This calls for adapting the JR axioms to the temporal setting.

Bulteau et al. [2021] and, subsequently, Chandak et al. [2024] defined temporal analogues of the justified representation axioms and investigated whether existing multiwinner rules with strong axiomatic properties can be adapted to the temporal setting so as to satisfy the new axioms.

We build upon the works of Bulteau et al. [2021] and Chandak et al. [2024], and focus on the complexity of *verifying* whether a given solution satisfies justified representation axioms. This task is important if, e.g., the outcome is fully or partially determined by external considerations, so explicitly using an algorithm to obtain an outcome with strong representation guarantees is not feasible, but representation remains an important concern. In multiwinner voting setting, the verification problem is known to be coNP-hard for PJR and EJR, but polynomial-time solvable for JR. We argue that the existing complexity results do not automatically transfer from the multiwinner setting to the temporal setting. However, we develop new proofs specifically tailored to the temporal setting, and show that in temporal elections all three properties are coNP-hard to verify, even under strong constraints on the structure of the input instance. Our complexity result for JR shows that the temporal setting is strictly harder than the multiwinner setting.

We complement our hardness results by fixed parameter tractability results as well as a polynomial-time algorithm for a natural special case of our model where candidates may join the election over time, but never leave, and voters' preferences over available candidates do not change. We also develop an integer linear programming formulation for the problem of selecting an outcome that provides EJR and satisfies additional linear constraints on voters' utilities, thereby establishing that this problem is fixed-parameter tractable with respect to the number of voters n.

Finally, we (partially) answer an open question of Chandak et al. [2024], by showing that the prominent Greedy Cohesive Rule [Bredereck et al., 2019, Peters et al., 2021] can be adapted to the temporal setting.

7.2 Preliminaries

An important concern in the multiwinner setting is group fairness, i.e., making sure that large groups of voters with similar preferences are represented by the selected committee. The most well-studied group fairness axioms are (in increasing order of strength) *justified representation (JR)* [Aziz et al., 2017], *proportional justified representation (PJR)* [Sánchez-Fernández et al., 2017], and *extended justified representation (EJR)* [Aziz et al., 2017]. We will now formulate these axioms, as well as their extensions to the temporal setting; for the latter, we follow the terminology of Chandak et al. [2024]. Definition 7.2.1 may appear syntactically different from the standard definitions of these notions, but it can easily be shown to be equivalent to them; further, it has the advantage of being easily extensible to the temporal setting.

Definition 7.2.1. For a multiwinner election $(P, N, (s_i)_{i \in N}, k)$ and a group of voters $N' \subseteq N$, we define the *demand* of N' as

$$\alpha^{mw}(N') = \min\{\beta^{mw}(N'), \gamma^{mw}(N')\}, \quad \text{where}$$

$$\beta^{mw}(N') = |\cap_{i \in N'} s_i| \quad \text{and} \quad \gamma^{mw}(N') = \left\lfloor k \cdot \frac{|N'|}{n} \right\rfloor.$$

A committee $W \subseteq P$ provides *justified representation (JR)* if for every $N' \subseteq N$ with $\alpha^{mw}(N') > 0$ there is an $i \in N'$ such that $|s_i \cap W| > 0$; it provides *proportional justified representation (PJR)* if for every $N' \subseteq N$ we have $|(\cup_{i \in N'} s_i) \cap W| \geq \alpha^{mw}(N')$, and it provides *extended justified representation (EJR)* if for every $N' \subseteq N$ there is an $i \in N'$ such that $|s_i \cap W| \geq \alpha^{mw}(N')$.

When extending these notions to the temporal setting, Bulteau et al. [2021] consider JR and PJR (but not EJR), each with three variants—prefixed with "static", "dynamic all-periods-intersection", and "dynamic some-periods-intersection"; where "dynamic some-periods intersection" is the most demanding of these. Chandak et al. [2024] extend this analysis to EJR, and focus on two variants of the axioms: "dynamic all-period intersection", which they call weak JR/PJR/EJR, and "dynamic some-period intersection", which they call JR/PJR/EJR[1]. In what follows, we use the terminology of Chandak et al. [2024].

Just as in the multiwinner setting, for each group of voters N' we determine its demand $\alpha(N')$, which depends both on the size of N' and on the degree of agreement among the group members. The axioms then require that the collective satisfaction of group members is commensurate with the group's demand.

Definition 7.2.2. Given an election $E = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$ and a group of voters $N' \subseteq N$, we define the *agreement* of N' as the number of rounds in which all members of N' approve a common candidate:

$$\beta(N') = |\{t \in [\ell] : \bigcap_{i \in N'} s_{it} \neq \varnothing\}|.$$

We define the *demand* of a group of voters N' as

$$\alpha(N') = \left\lfloor \beta(N') \cdot \frac{|N'|}{n} \right\rfloor.$$

That is, if voters in N' agree in β rounds, they can demand a fraction of these rounds that is proportional to $|N'|$.

We now proceed to define temporal extensions of JR, PJR, and EJR, starting with their stronger versions

Definition 7.2.3 (Justified Representation)**.** An outcome $\mathbf{o}$ provides *justified representation (JR)* for a temporal election $E = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$ if for every group of voters $N' \subseteq N$ with $\alpha(N') > 0$ we have $sat_i(\mathbf{o}) > 0$ for some $i \in N'$.

Definition 7.2.4 (Proportional Justified Representation)**.** An outcome $\mathbf{o}$ provides *proportional justified representation (PJR)* for a temporal election $E = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$ if for every group of voters $N' \subseteq N$ it holds that $|\{t \in [\ell] : o_t \in \bigcup_{i \in N'} s_{it}\}| \geq \alpha(N')$.

Definition 7.2.5 (Extended Justified Representation)**.** An outcome $\mathbf{o}$ provides *extended justified representation (s-EJR)* for a temporal election $E = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$ if for every group of voters $N' \subseteq N$ there exists a voter $i \in N'$ with $sat_i(\mathbf{o}) \geq \alpha(N')$.

[1] The conference version of their paper uses "JR/PJR/EJR" for the weaker version and "strong JR/PJR/EJR" for the stronger version; we use the terminology from the arXiv version of their paper.

Note that EJR implies PJR, and PJR implies JR.

It is instructive to compare Definitions 7.2.3–7.2.5 to Definition 7.2.1. In particular, one may wonder why we did not define the demand of a group N' in the temporal setting as $\overline{\alpha}(N') = \min\{\beta(N'), \ell \cdot \frac{|N'|}{n}\}$, thereby decoupling the constraints on the size of the group and the level of agreement. Note that $\beta(N') \leq \ell$ for all $N' \subseteq N$ and hence $\overline{\alpha}(N') \geq \alpha(N')$ for all N'. However, the following example shows that if we were to use this definition of demand, even the JR axiom would be impossible to satisfy (whereas Chandak et al. [2024] show that every temporal election admits an outcome that provides EJR); see also the discussion in Section 5 of the paper by Chandak et al. [2024].

Example 7.2.6. Let $N = \{1, \ldots, 6\}$, and let $\mathcal{T}$ be the set of all size-2 subsets of N (so $|\mathcal{T}| = 15$). Consider an election with voter set N, $P = \{x_T\}_{T\in\mathcal{T}} \cup \{y_j\}_{j=1,\ldots,6}$, and $\ell = 3$, where the voters' approval sets have the following structure:

$$s_{i1} = \{x_T : i \in T\}, \quad s_{i2} = s_{i3} = \{y_i\} \quad \text{for all } i \in N.$$

For each group of voters N' with $|N'| = 2$ we have $\beta(N') = 1$, as both voters in N' approve $x_{N'}$ in the first round. Moreover, $\ell \cdot \frac{|N'|}{n} = 1$, so $\overline{\alpha}(N') = 1$. Hence, if we were to replace $\alpha(N')$ with $\overline{\alpha}(N')$ in the definition of JR, we would have to ensure that for every pair of voters N' at least one voter in N' obtains positive satisfaction. However, there is no outcome $\mathbf{o}$ that accomplishes this: o_1 is approved by at most two voters, and o_2 and o_3 are approved by at most one voter each, so for every outcome $\mathbf{o}$ there are two voters whose satisfaction is 0. That is, the modified definition of JR (and hence PJR and EJR) is unsatisfiable.

On the other hand, we have $\beta(N') = 3$ if $|N'| = 1$, $\beta(N') = 1$ if $|N'| = 2$ and $\beta(N') = 0$ if $|N'| \geq 3$, and hence $\alpha(N') = 0$ for all $N' \subseteq N$. Thus, every outcome $\mathbf{o}$ provides EJR (and hence PJR and JR).

The axioms introduced so far apply to groups of voters with positive agreement. Bulteau et al. [2021] and Chandak et al. [2024] also consider weaker axioms, which only apply to groups of voters that agree in all rounds.

Definition 7.2.7 (Weak Justified Representation/Proportional Justified Representation/Extended Justified Representation)**.** Consider an outcome $\mathbf{o}$ for a temporal election $E = (P, N, \ell, (\mathbf{s}_i)_{i\in N})$. We say that $\mathbf{o}$ provides:

- *weak justified representation* (w-JR) if for every group of voters $N' \subseteq N$ with $\beta(N') = \ell$, $\alpha(N') > 0$ it holds that $sat_i(\mathbf{o}) > 0$ for some $i \in N'$.

- *weak proportional justified representation* (w-PJR) if for every group of voters $N' \subseteq N$ with $\beta(N') = \ell$ it holds that $|\{t \in [\ell] : o_t \in \bigcup_{i \in N'} s_{it}\}| \geq \alpha(N')$.
- *weak extended justified representation* (w-EJR) if for every group of voters $N' \subseteq N$ with $\beta(N') = \ell$ there exists a voter $i \in N'$ with $sat_i(\mathbf{o}) \geq \alpha(N')$.

By construction, the weak JR/PJR/EJR axioms are less demanding than their regular counterparts: e.g., if $\ell = 10$ and all voters agree on a candidate in each of the first 9 rounds, but each voter approves a different candidate in the last round, these axioms are not binding. Indeed, they are somewhat easier to satisfy: e.g., Chandak et al. [2024] show that a natural adaptation of the Method of Equal Shares [Peters and Skowron, 2020] satisfies w-EJR, but not EJR. Therefore, in our work we consider both families of axioms.

7.3 Hardness Proofs

In multiwinner elections, outcomes that provide EJR (and thus JR and PJR) can be computed in polynomial time [Aziz et al., 2018, Peters and Skowron, 2020]. In contrast, the problem of verifying if a given outcome provides JR/PJR/EJR is considerably more challenging: while this problem is polynomial-time solvable for JR, it is coNP-hard for PJR and EJR [Aziz et al., 2017, 2018].

In this section, we show that in the temporal setting the verification problem is coNP-hard even for (w-)JR. These results extend to (w-)PJR and (w-)EJR.

We note that the hardness results for PJR and EJR in the multiwinner setting do not transfer immediately to the temporal setting. This is because the notion of agreement in our model is fundamentally different from the one in the multiwinner model: we focus on the number of rounds in which a group of voters agrees on a candidate, whereas in the multiwinner model the cohesiveness of a group is determined by the number of candidates the group members agree on. Suppose that we naively transform a multiwinner instance $(P, N, (s_i)_{i \in N}, k)$, where k is the number of candidates to be elected, into a temporal instance with k rounds where voters' approvals in each round are given by $(s_i)_{i \in N}$. Consider a group of voters N' that only agrees on a single candidate in the original instance (and therefore can demand at most one spot in the elected committee). In the temporal setting, voters in N' will agree on that candidate in all rounds, and therefore if N' is large, its demand can be substantial. Thus, we have to prove hardness of verifying PJR and EJR from scratch.

Fortunately, the proofs of Theorems 7.3.1 and 7.3.2 apply to all three representation concepts that we consider.

Theorem 7.3.1. *For each of $X \in \{$w-JR, w-PJR, w-EJR$\}$, verifying whether an outcome provides X is* coNP*-complete. The hardness result holds for w-JR and w-PJR even if $|P| = 3$, and for w-EJR even if $|P| = 2$.*

Proof. If an outcome $\mathbf{o}$ does not provide w-JR, this can be witnessed by a group of voters $N' \subseteq N$ with $\beta(N') = \ell$, $\alpha(N') > 0$ such that $sat_i(\mathbf{o}) = 0$ for all $i \in N'$. Hence, our problem is in coNP. Similarly, for w-PJR (respectively, w-EJR) it suffices to exhibit a group of voters N' such that $\beta(N') = \ell$ and $|\{t \in [\ell] : o_t \in \cup_{i \in N'} s_{it}\}| < \alpha(N')$ (respectively, $\beta(N') = \ell$, $sat_i(\mathbf{o}) < \alpha(N')$ for all $i \in N'$).

To prove coNP-hardness for w-JR, we reduce from the NP-hard problem CLIQUE. An instance of CLIQUE is given by a graph $G = (V, E)$ with vertex set V and edge set E, together with a parameter κ; it is a 'yes'-instance if there exists a subset of vertices $V' \subseteq V$ of size κ that forms a clique, i.e., for all $v, v' \in V'$ we have $\{v, v'\} \in E$.

Given an instance (G, κ) of CLIQUE with $G = (V, E)$, where $V = \{v_1, \ldots, v_\nu\}$, we construct an election with a set of voters $N = \{1, \ldots, \nu\} \cup N_0$, where $|N_0| = (\kappa-1)\nu$, a set of candidates $P = \{p_1, p_2, p_3\}$, and $\ell = \nu$ rounds. For each $i \in [\nu]$, the preferences of voter i are given by:

$$s_{it} = \begin{cases} \{p_2\} & \text{if } t = i; \\ \{p_1, p_2\} & \text{if } \{v_i, v_t\} \in E; \\ \{p_1\} & \text{otherwise.} \end{cases}$$

For each $i \in N_0$ and each $t \in [\ell]$ we set $s_{it} = \{p_3\}$.

We claim that a set of voters $N' \subseteq [\nu]$ satisfies $\beta(N') = \ell$ if and only if the set $V' = \{v_i : i \in N'\}$ forms a clique in G. Indeed, let V' be a clique in G, and consider a round $t \in [\ell]$. If $t \in N'$, we have $s_{tt} = \{p_2\}$, $s_{it} = \{p_1, p_2\}$ for all $i \in N' \setminus \{t\}$, i.e., $p_2 \in \cap_{i \in N'} s_{it}$. On the other hand, if $t \notin N'$, for each $i \in N'$ we have $s_{it} = \{p_1, p_2\}$ if $\{v_i, v_t\} \in E$ and $s_{it} = \{p_1\}$ otherwise, so $p_1 \in \cap_{i \in N'} s_{it}$. Thus, for each $t \in [\ell]$ the set $\cap_{i \in N'} s_{it}$ is non-empty. Conversely, if $\{v_i, v_j\} \notin E$ for some $i, j \in N'$, $i \neq j$, then $s_{ij} = \{p_1\}$, whereas $s_{jj} = \{p_2\}$, i.e., $\cap_{i' \in N'} s_{i'j} = \varnothing$ and hence $\beta(N') < \ell$.

We claim that an outcome $\mathbf{o} = (p_3, \ldots, p_3)$ fails to provide w-JR if and only if G contains a clique of size κ.

Indeed, let V' be a size-κ clique in G, and let $N' = \{i : v_i \in V'\}$. We have argued that $\beta(N') = \ell$ and hence $\alpha(V') = \ell \cdot \frac{\kappa}{\kappa \cdot \nu} = 1 > 0$; however, $sat_i(\mathbf{o}) = 0$ for each $i \in N'$. For the converse direction, suppose there exists a subset of voters N' witnessing that

$\mathbf{o}$ fails to provide w-JR. As $sat_i(\mathbf{o}) = 0$ for each $i \in N'$, we have $N' \cap N_0 = \varnothing$, i.e., $N' \subseteq [\nu]$. Moreover, $\beta(N') = \ell$ and hence the set $V' = \{v_i : i \in N'\}$ forms a clique in G. Then $\alpha(N') > 0$ can be re-written as $\ell \cdot \frac{|N'|}{\kappa \cdot \nu} \geq 1$, i.e., $|N'| \geq \kappa$. As $|V'| = |N'|$, the set V' is a clique of size at least κ in G.

The argument extends easily to w-PJR/w-EJR: if G contains a clique of size κ, then, as argued above, $\mathbf{o}$ fails to provide w-JR (and hence it fails to provide w-PJR/w-EJR), whereas if there are no cliques of size κ in G, there is no group of voters $N' \subseteq N$ with $\beta(N') = \ell$, $\alpha(N') > 0$, so w-PJR/w-EJR is trivially satisfied.

We will now provide a (different) proof that verifying whether an outcome provides w-EJR is coNP-hard even if $|P| = 2$. To this end, we reduce from the NP-hard problem Independent Set with Maximum Degree of Three (IS-3) Garey et al. [1976]. An instance of IS-3 is given by a graph $G = (V, E)$ with a vertex set V and an edge set E such that every vertex in V has at most three neighbors, together with a parameter κ; it is a 'yes'-instance if there exists a subset of vertices $V' \subseteq V$ with $|V'| \geq \kappa$ such that for all vertices $u, v \in V'$ it holds that $\{u, v\} \notin E$, and a 'no' instance otherwise. We can assume without loss of generality that $\kappa \leq |V| - 3$: indeed, if $\kappa \geq |V| - 2$, we can solve IS-3 in time $O(|V|^2)$ by checking all size-κ subsets of V.

Given an instance (G, κ) of IS-3 with $G = (V, E)$, where $V = \{v_1, \ldots, v_\nu\}$ and $\kappa \leq \nu - 3$, we construct an election with a set of voters $N = \{1, \ldots, \nu\} \cup N_0$, where $|N_0| = \nu + 1 - \kappa$, a set of candidates $P = \{p, q\}$, and $\ell = 2\nu + 1 - \kappa$ rounds. For each $i \in N$, the preferences of voter i in round t are given by:

$$s_{it} = \begin{cases} \{p\} & \text{if } t > \nu \\ \{q\} & \text{if } i \in [\nu] \text{ and } t = i; \\ \{p\} & \text{if } i, t \in [\nu] \text{ and } \{v_i, v_t\} \in E; \\ \{p, q\} & \text{in all other cases.} \end{cases}$$

Set $\mathbf{o} = (q, \ldots, q)$. We claim that there exists a set of voters $N^* \subseteq [\nu] \cup N_0$ such that $\beta(N^*) = \ell$ but $sat_i(\mathbf{o}) < \alpha(N^*)$ for all $i \in N^*$ if and only if the set $V' = \{v_i : i \in N^* \cap [\nu]\}$ is an independent set of size at least κ in G.

For the 'if' direction, let V' be an independent set of size at least κ in G, and let N' be the set of voters that correspond to the vertices in V'. Set $N^* = N' \cup N_0$. We claim that $\beta(N^*) = \ell$. Indeed, consider two voters $i, i' \in N'$, a voter $j \in N_0$, and a round $t \in [\ell]$; we will now show that $s_{it} \cap s_{i't} \cap s_{jt} \neq \varnothing$. To this end, we consider the following cases:

- $t > \nu$. Then $s_{it} = s_{i't} = s_{jt} = \{p\}$.

- $t = i$. Then $s_{it} = \{q\}$ and $s_{i',t} = s_{jt} = \{p, q\}$;
- $t = i'$. Then $s_{i',t} = \{q\}$ and $s_{i,t} = s_{j,t} = \{p, q\}$;
- $t \in [\nu] \setminus \{i, i'\}$. Then $p \in s_{i,t}$, $p \in s_{i',t}$, $s_{j,t} = \{p, q\}$.

As all three voters approve a common candidate in each round, we have $\beta(N^*) = \ell$. Then, we get

$$\alpha(N^*) = \left\lfloor \beta(N^*) \cdot \frac{|N^*|}{2\nu + 1 - \kappa} \right\rfloor \geq \left\lfloor (2\nu + 1 - \kappa) \cdot \frac{\nu + 1}{2\nu + 1 - \kappa} \right\rfloor = \nu + 1.$$

However, since $s_{i,t} = \{p\}$ for all $i \in N$ and all $t > \nu$, the satisfaction of every voter in N^* from outcome $\mathbf{o}$ does not exceed ν. Thus, we obtain $sat_i(\mathbf{o}) < \alpha(N^*)$ for all $i \in N^*$.

For the 'only if' direction, suppose there exists a set of voters $N^* \subseteq [\nu] \cup N_0$ such that $\beta(N^*) = \ell$ but $sat_i(\mathbf{o}) < \alpha(N^*)$ for all $i \in N^*$. Let $N' = N^* \cap [\nu]$, and let $V' = \{v_i : i \in N'\}$. We claim that V' is an independent set. Indeed, if $\{v_i, v_j\} \in E$ for some $i, j \in N'$ then $s_{i,i} = \{q\}$ and $s_{i,j} = \{p\}$, which is a contradiction with $\beta(N^*) = \ell$. It remains to argue that $|V'| \geq \kappa$. To this end, we note that $n = \ell = 2\nu + 1 - \kappa$ and $\beta(N^*) = \ell$ implies that $\alpha(N^*) = |N^*|$, and consider two cases.

- $N^* \subseteq [\nu]$. In this case, $N' = N^*$. Since every vertex in V has at most three neighbors, for each $i \in N'$ we have $sat_i(\mathbf{o}) \geq \nu - 3$. On the other hand, by our assumption, $sat_i(\mathbf{o}) < \alpha(N^*) = |N^*|$ for all $i \in N^*$. Therefore, V' is an independent set of size $|N^*| > \nu - 3 \geq \kappa$.
- $N^* \cap N_0 \neq \varnothing$. Then, consider an agent $i \in N^* \cap N_0$. For each $i \in N_0$ we have $sat_i(\mathbf{o}) = \nu$ and hence $|N^*| = \alpha(N^*) > \nu$. Since $|N_0| = \nu + 1 - \kappa$, this implies $|V'| = |N'| = |N^* \setminus N_0| > \kappa - 1$. Thus, $|V'|$ is an independent set of size at least κ.

In both cases, we conclude that V' is an independent set of size at least κ, and hence our input instance of IS-3 was a 'yes'-instance. This completes the proof. □

For stronger versions of our axioms, we obtain a hardness result even for $|P| = 2$.

Theorem 7.3.2. *For each of $X \in \{$JR, PJR, EJR$\}$, verifying whether an outcome provides X is* coNP*-complete. The hardness result holds even if $|P| = 2$.*

Proof. It is easy to see that our problems are in coNP: we can modify the proof of Theorem 7.3.1 by omitting the condition $\beta(N') = \ell$.

To establish coNP-hardness, we give a reduction from the NP-hard problem Maximum Edge Biclique Peeters [2003]. An instance of Maximum Edge Biclique is given by a bipartite graph $G = (L \cup R, E)$ and a parameter κ; it is a 'yes'-instance if and only if G contains a biclique with at least κ edges, i.e., there exist $L' \subseteq L$, $R' \subseteq R$ such that for each $i \in L'$, $j \in R'$ it holds that $\{i, j\} \in E$ and $|L'| \cdot |R'| \geq \kappa$.

We can assume without loss of generality that $\kappa > |L| + |R|$. Indeed, if this is not the case, we can construct a new graph G' by replacing each vertex $v \in L \cup R$ with $\xi = |L| + |R| + 1$ copies $v_1, \dots, v_\xi$ and adding an edge between the i-th copy of v and the j-th copy of u for all $i, j \in [\xi] \times [\xi]$ if and only if $\{v, u\} \in E$. The graph G' has $\xi(|L| + |R|)$ vertices, and it contains a biclique with $\xi^2\kappa$ edges if and only if G contains a biclique with κ edges; our choice of ξ ensures that $\xi^2\kappa > \xi(|L| + |R|)$.

We will give a coNP-hardness proof for JR and then extend it to PJR and EJR. Consider an instance of Maximum Edge Biclique given by a graph $G = (L \cup R, E)$ and a parameter κ, where $L = \{v_1, \dots, v_\nu\}$, $R = \{u_1, \dots, u_\lambda\}$ and $\kappa > \nu + \lambda$. We construct an election with a set of voters $N = \{1, \dots, \nu\} \cup N_0$, where $|N_0| = \kappa - \nu$, a set of candidates $P = \{p, q\}$, and $\ell = \lambda$ rounds. For each $i \in [\nu]$ and each $t \in [\ell]$ we set $s_{i,t} = \{p\}$ if $\{v_i, u_t\} \in E$ and $s_{i,t} = \varnothing$ otherwise; for each $i \in N_0$ and each $t \in [\ell]$ we set $s_{i,t} = \{q\}$. Let $\mathbf{o} = (q, \dots, q)$.

Suppose G contains a biclique with parts $L' \subseteq L$ and $R' \subseteq R$ and at least κ edges, and consider the set of voters $N' = \{i : v_i \in L'\}$ and the set of rounds $T = \{t \in [\ell] : u_t \in R'\}$. Note that for each $i \in N'$, $t \in T$ we have $s_{i,t} = \{p\}$ and hence $\beta(N') \geq |T| = |R'|$. Therefore, we obtain

$$\alpha(N') = \left\lfloor \beta(N') \cdot \frac{|N'|}{|N|} \right\rfloor \geq \left\lfloor \frac{|R'| \cdot |L'|}{|N|} \right\rfloor \geq \frac{\kappa}{\kappa} = 1.$$

Thus, N' has positive demand, but no voter in N' derives positive satisfaction from $\mathbf{o}$, i.e., JR is violated.

Conversely, suppose that there is a group of voters N' with positive demand such that $sat_i(\mathbf{o}) = 0$ for each $i \in N'$. Then $N' \subseteq [\nu]$, as all voters in N_0 approve q in each round. Consider a set of rounds $T \subseteq [\ell]$ given by $T = \{t \in [\ell] : \cap_{i \in N'} s_{i,t} \neq \varnothing\}$; we have $s_{i,t} = \{p\}$ for each $i \in N'$, $t \in T$. Thus, (N', T) corresponds to a biclique in G: indeed, we have $s_{i,t} = \{p\}$ if and only if $\{v_i, u_t\} \in E$. It follows that G contains a biclique with at least $|N'| \cdot |T|$ edges. On the other hand, we have $\beta(N') = |T|$ and hence $\alpha(N') = \lfloor \frac{|N'| \cdot |T|}{\kappa} \rfloor$. Thus, $\alpha(N') \geq 1$ implies $|N'| \cdot |T| \geq \kappa$.

To extend our result to PJR and EJR, we note that if $\mathbf{o}$ provides PJR (respectively, EJR) then it also provides JR, and hence, as argued above, G does not contain a biclique of size κ. Conversely, if $\mathbf{o}$ violates PJR (respectively, EJR), there exists a group of voters N' with a positive demand such that $|\{t \in [\ell] : o_t \in \cup_{i \in N'} s_{i,t}\}| < \alpha(N')$ (respectively, $sat_i(\mathbf{o}) < \alpha(N')$ for all $i \in N'$). Now, the satisfaction of each voter in N_0 is ℓ, and the demand of any group of voters is at most ℓ. Hence, $N' \cap N_0 = \varnothing$, i.e., $N' \subseteq [\nu]$. As argued above, the conditions $N' \subseteq [\nu]$ and $\alpha(N') > 0$ imply that the pair (N', T), where $T = \{t \in [\ell] : \cap_{i \in N'} s_{i,t} \neq \varnothing\}$, corresponds to a biclique in G that has at least κ edges. □

7.4 Tractability Results for Two Projects

The hardness results in Theorems 7.3.1 and 7.3.2 hold even if the number of candidates $|P|$ is small (3 for w-JR/w-PJR and 2 for w-EJR/JR/PJR/EJR). Now, for $|P| = 1$ verification is clearly easy: there is only one possible outcome, and for any group of voters N' this outcome provides satisfaction of at least $\beta(N') \geq \alpha(N')$ to all voters in N'. To complete the complexity classification with respect to $|P|$, it remains to consider the complexity of verifying w-JR/w-PJR when $|P| = 2$.

Proposition 7.4.1. *Given an election $(P, N, \ell, (\mathbf{s}_i)_{i \in N})$ with $|P| = 2$ and an outcome $\mathbf{o}$, we can check in polynomial time whether $\mathbf{o}$ provides w-JR.*

Proof. Assume that $P = \{p, q\}$. Suppose that a group of voters N' witnesses that $\mathbf{o}$ fails to provide w-JR. Then $sat_i(\mathbf{o}) = 0$ for each $i \in N'$, but $\beta(N') = \ell$.

We say that a voter i is *grumpy* if in each round t she approves a single candidate, and this candidate is not o_t, i.e., $|s_{i,t}| = 1$ and $o_t \notin s_{i,t}$ for all $t \in [\ell]$. Let G be the set of all grumpy voters, and note that $\beta(G) = \ell$: in each round t, all grumpy voters approve the (unique) candidate in $P \setminus \{o_t\}$.

Note that $N' \subseteq G$: indeed, if a voter i is not grumpy, either she approves o_t for some $t \in [\ell]$, so $sat_i(\mathbf{o}) > 0$, or she has $s_{i,t} = \varnothing$ for some $t \in [\ell]$, which is not compatible with $\beta(N') = \ell$. Hence, to check whether $\mathbf{o}$ provides w-JR it suffices to verify whether $\alpha(G) = \lfloor \ell \cdot \frac{|G|}{|N|} \rfloor = 0$. □

It is not clear if Proposition 7.4.1 can be extended to w-PJR: to verify w-PJR, it is no longer sufficient to focus on grumpy voters. We conjecture that verifying w-PJR remains hard even for $|P| = 2$ (recall that, by Theorem 7.3.1, this is the case for w-EJR).

Observe that for the election constructed in the proof of Theorem 7.3.1 we have $s_{i,t} \neq \varnothing$ for all $i \in N$, $t \in [\ell]$. In contrast, in the election constructed in the proof of Theorem 7.3.2 the approval sets $s_{i,t}$ may be empty. It turns out that if we require that $s_{i,t} \neq \varnothing$ for all $i \in N$, $t \in [\ell]$ then for $|P| = 2$ the problem of verifying JR becomes easy; under this assumption, if $sat_i(\mathbf{o}) = 0$ for all $i \in N'$ then all voters in N' are grumpy, so it suffices to check if $\alpha(G) = 0$.

Proposition 7.4.2. *Given an election $(P, N, \ell, (\mathbf{s}_i)_{i\in N})$ with $|P| = 2$ and $s_{i,t} \neq \varnothing$ for all $i \in N$, $t \in [\ell]$ and an outcome $\mathbf{o}$, we can check in polynomial time whether $\mathbf{o}$ provides JR.*

Proof. Again we can assume without loss of generality that $P = \{p, q\}$ and, by the same argument as in the proof of Proposition 7.4.1, we can assume $\mathbf{o} = (q, \dots, q)$.

As we assume that $s_{i,t} \neq \varnothing$ for all $i \in N$, $t \in [\ell]$, each voter in N' derives zero satisfaction from $\mathbf{o}$ if and only if $N' \subseteq N_p$, where N_p is defined as in the proof of Proposition 7.4.1, i.e., $N_p = \{i \in N : s_{i,t} = \{p\} \text{ for all } t \in [\ell]\}$. Note that $\beta(N') = \beta(N_p) = \ell$, so, to check if $\mathbf{o}$ provides JR it suffices to verify whether $\alpha(N_p) = \lfloor \ell \cdot \frac{|N_p|}{|N|} \rfloor = 0$. □

Again, it is not clear if Proposition 7.4.2 extends to (w-)PJR and (w-)EJR; we conjecture that the answer is 'no'.

Remark 7.4.3. The proofs of Propositions 7.4.1 and 7.4.2 go through if, instead of requiring $|P| = 2$, we require that in each round there are at most two candidates that receive approvals from the voters, i.e., $|\cup_i s_{i,t}| \leq 2$ for each $t \in [\ell]$.

Proposition 7.4.2 shows that, for $|P| = 2$, requiring each voter to approve at least one candidate in each round reduces the complexity of checking JR considerably. However, this requirement only helps if $|P|$ is bounded. Indeed, we can modify the construction in the proof of Theorem 7.3.2 by creating a set of additional candidates $P^N = \{p^i\}_{i\in N}$ and modifying the preferences so that whenever $s_{i,t} = \varnothing$ in the original construction, we set $s_{i,t} = \{p^i\}$. The new candidates do not change the agreement of any voter group, so the rest of the proof goes through unchanged.

Proposition 7.4.4. *For each of $X \in \{$JR, PJR, EJR$\}$, verifying whether an outcome provides X is* coNP*-complete, even if $s_{i,t} \neq \varnothing$ for all voters $i \in N$ and all rounds $t \in [\ell]$.*

It remains an open question whether verifying that a given outcome provides JR/PJR/EJR remains coNP-complete if all approval sets are non-empty and the size of the set P is a fixed constant greater than 2; we conjecture that this is indeed the case, but we were unable to prove this.

7.5 Parameterized Complexity

We have seen that the verification problem becomes easier if the size of the candidate set $m = |P|$ is very small. We will now consider other natural parameters of our problem, such as the number of voters n and the number of rounds ℓ, and explore the complexity of the verification problem with respect to there parameters.

Proposition 7.5.1. *For each $X \in \{$(w-)JR, (w-)PJR, (w-)EJR$\}$, checking if an outcome provides X is FPT with respect to n.*

Proof. Fix an outcome $\mathbf{o}$. We go through all subsets of voters $N' \subseteq N$. For each subset N' we compute $\beta(N')$, by going through all rounds $t \in [\ell]$ and, for each round t, checking whether $\cap_{i \in N'} s_{i,t} \neq \varnothing$. We then compute $\alpha(N')$ as well as the satisfaction that each voter in N' derives from $\mathbf{o}$. We can then easily verify whether the (w-)EJR (respectively, (w-)PJR, (w-)JR) condition is violated for N'.

The bound on the running time follows immediately: we consider 2^n subsets of voters and perform a polynomial amount of work for each subset. $\square$

The proof of the following proposition, as well as the results in Section 7.6, make use of the following observation.

Observation 7.5.2. *Let X be a multiset of real numbers, and let $r \leq |X|$ be a positive integer. Let Y and Z be two subsets of X with $|Y| = |Z| = r$ such that Z consists of the r smallest elements of X (for some way of breaking ties). Then for each $z \in Z$ we have $z \leq \max_{y \in Y} y$.*

Proposition 7.5.3. *For each $X \in \{$(w-)JR, (w-)PJR, (w-)EJR$\}$, checking if an outcome provides X is FPT with respect to the combined parameter (m, ℓ) and XP with respect to ℓ.*

Proof. Given an outcome $\mathbf{o}$, we proceed as follows. For each possible outcome $\mathbf{o}'$ and each subset $T \subseteq [\ell]$, compute the set $N_{\mathbf{o}',T} = \{i \in N : o'_t \in s_{i,t} \text{ for all } t \in T\}$; this set consists of all voters who approve the outcome $\mathbf{o}'$ in each of the rounds in T. We then sort the voters in $N_{\mathbf{o}',T}$ according to their satisfaction under $\mathbf{o}$ in non-decreasing order, breaking ties lexicographically; for each $r = 1, \dots, |N_{\mathbf{o}',T}|$, let $N^r_{\mathbf{o}',T}$ denote the set that consists of the first r voters in this order.

To verify EJR, for each $r = 1, \dots, |N_{\mathbf{o}',T}|$ we determine whether there is an $i \in N^r_{\mathbf{o}',T}$ such that $sat_i(\mathbf{o}) \geq \lfloor \frac{r}{n} \cdot |T| \rfloor$. We claim that $\mathbf{o}$ provides EJR if and only if it passes these checks for all possible choices of $\mathbf{o}'$, T and r.

Indeed, suppose $\mathbf{o}$ fails this check for some $\mathbf{o}'$, T and r, and let $N' = N^r_{\mathbf{o}',T}$. By construction, $\beta(N') \geq |T|$ and $|N'| = r$, so $\alpha(N') \geq \lfloor \frac{r}{n} \cdot |T| \rfloor$, i.e., N' is a witness that EJR is violated.

Conversely, suppose that $\mathbf{o}$ fails EJR. Then there is a set of voters N' such that $\max_{i \in N'} sat_i(\mathbf{o}) < \alpha(N')$. By definition of $\beta(N')$, there exists a set $T' \subseteq [\ell]$, $|T'| = \beta(N')$, and an outcome $\mathbf{o}'$ such that for each $t \in T'$ and each $i \in N'$ it holds that $o'_t \in s_{i,t}$; note that this implies $N' \subseteq N_{\mathbf{o}',T'}$. Let $r = |N'|$, and consider the set $N^r_{\mathbf{o}',T'}$. Observe that $\beta(N^r_{\mathbf{o}',T'}) \geq |T'| = \beta(N')$ and $|N^r_{\mathbf{o}',T'}| = |N'| = r$, so $\alpha(N^r_{\mathbf{o}',T'}) \geq \alpha(N')$. On the other hand, by applying Observation 7.5.2 to the multisets $\{sat_i(\mathbf{o}) : i \in N'\}$ and $\{sat_i(\mathbf{o}) : i \in N^r_{\mathbf{o}',T'}\}$, we conclude that for each $i \in N^r_{\mathbf{o}',T'}$ it holds that $sat_i(\mathbf{o}) \leq \max_{i \in N'} sat_i(\mathbf{o}) < \alpha(N') \leq \alpha(N^r_{\mathbf{o}',T'})$. That is, $N^r_{\mathbf{o}',T'}$ is also a witness that $\mathbf{o}$ fails to provide EJR. As our algorithm checks $N^r_{\mathbf{o}',T'}$, it will be able to detect that $\mathbf{o}$ fails to provide EJR.

For PJR, instead of checking whether $sat_i(\mathbf{o}) \geq \lfloor \frac{r}{n} \cdot |T| \rfloor$ for some $i \in N^r_{\mathbf{o}',T}$, we check that $|\{t \in [\ell] : o_t \in \cup_{i \in N^r_{\mathbf{o}',T}} s_{i,t}\}| \geq \lfloor \frac{r}{n} \cdot |T| \rfloor$, and for JR we check whether $\lfloor \frac{r}{n} \cdot |T| \rfloor \geq 1$ implies that $sat_i(\mathbf{o}) \geq 1$ for some $i \in N^r_{\mathbf{o}',T}$. For w-JR, w-PJR and w-EJR, the algorithm can be simplified: instead of iterating through all $T \subseteq [\ell]$, it suffices to consider $T = [\ell]$. The rest of the argument goes through without change.

The bound on the runtime follows, as we have m^ℓ possibilities for $\mathbf{o}'$ and 2^ℓ possibilities for T, and we process each pair $(\mathbf{o}', T)$ in polynomial time. □

Our XP result with respect to ℓ is tight, at least for weak versions of the axioms: our next theorem shows that checking whether an outcome provides w-JR, w-PJR, or w-EJR is W[1]-hard with respect to ℓ. This indicates that an FPT (in ℓ) algorithm does not exist unless FPT = W[1].

Theorem 7.5.4. *For each of $X \in$ {w-JR, w-PJR, w-EJR}, verifying whether an outcome provides X is $W[1]$-hard with respect to ℓ.*

Proof. To show W[1]-hardness, we reduce from the *multicolored clique* problem. In this problem, we are given a parameter k and a k-partite graph $G = (V_1 \sqcup \dots \sqcup V_k, E)$ with the goal being to find a clique of size k. It is well known that multicolored clique is W[1]-hard when parameterized by k [Cygan et al., 2015]. We construct an election as follows: Let $|V_1 \sqcup \cdots \sqcup V_k| = n'$ be the number of vertices in G.

In our election instance, we have k rounds, one for each partition of G. At round $i \in [k]$, the candidates are the vertices in V_i together with one dummy candidate. Let each voter corresponding to a vertex in V_i only approve of the candidate corresponding to itself. Let each voter corresponding to a vertex not in V_i approve all candidates corresponding to vertices connected to it. The dummy candidate is approved by no voter.

Finally, to make the instance balanced, we compare $\frac{n'}{k}$ to k. If $\frac{n'}{k} < k$, we add $k^2 - n$ voters that approves of no candidates to the instance. Thus, the k vertices corresponding to a clique would correspond to exactly $\frac{n}{k}$ many voters. If $\frac{n'}{k} > k$ we add $n'' = \lceil \frac{n'-k^2}{(k-1)} \rceil$ voters that each approve of every candidate to the instance. Therefore, the n'' new voters together with k potential clique voters correspond to an almost $\frac{n'+n''}{k}$ fraction of the new instance.

We now claim that the committee consisting of dummy candidates does not satisfy w-JR if and only if there exists a multicolored clique.

First, assume that committee consisting of dummy candidates does not satisfy w-JR. Then there must exist $\frac{n}{k} \geq k$ vertices agreeing in every instance. Based on the way we added candidates, this must include at least k voters corresponding to vertices. These k voters must come from different partitions of the graph, as otherwise they would not agree in at least one round. Further, in the other rounds, they must agree on a vertex and are thus all connected to this vertex. Therefore, they form a clique. Analogously, all the voters correspond to a clique (together with potentially added voters approving everything) also witness a w-JR violation, since they all agree on a candidate during each round and are large enough by construction. □

7.6 Monotonic Preferences

Since the problem of checking whether an outcome provides (w-)JR, (w-)PJR, or (w-)EJR is intractable in general, it is natural to seek a restriction on voters' preferences that may yield positive results. In particular, a natural restriction in this context is monotonicity: once a voter approves a candidate, she continues to approve it in subsequent rounds. Formally, we will say that an election $(P, N, \ell, (\mathbf{s}_i)_{i \in N})$ is *monotonic* if for any two rounds $t, t' \in [\ell]$ with $t < t'$, each $p \in P$ and each $i \in N$ it holds that $p \in s_{i,t}$ implies $p \in s_{i,t'}$. Monotonic elections occur if, e.g., candidates join the candidate pool over time, but never leave, and the voters' preferences over the available candidates do not change. Such preferences can also arise in settings where the candidates improve over time: e.g., job candidates become more experienced. By reversing the time line, this can also model candidates that deteriorate over time (e.g., planning family meals from available ingredients).

We first note that verifying weak JR/PJR/EJR in monotonic elections is easy.

Theorem 7.6.1. *Given a monotonic election $(P, N, \ell, (\mathbf{s}_i)_{i\in N})$ togther with an outcome $\mathbf{o}$, for each of $X \in \{$w-JR, w-PJR, w-EJR$\}$ we can decide in polynomial time whether $\mathbf{o}$ provides X.*

Proof. Suppose that $(P, N, \ell, (\mathbf{s}_i)_{i\in N})$ is monotonic, and consider a group of voters $N' \subseteq N$. Note that $\beta(N') = \ell$ if and only if $\cap_{i\in N'} s_{i,1} \neq \varnothing$: if all voters in N' approve some candidate p in the first round, they also approve p in all subsequent rounds. This enables us to proceed similarly to the proof of Proposition 7.5.3.

Specifically, fix an outcome $\mathbf{o}$, and for each $p \in P$, let $N_p = \{i \in N : p \in s_{i,1}\}$. Order the voters in N_p according to their satisfaction from $\mathbf{o}$ in non-decreasing order, breaking ties lexicographically. For each $z = 1, \ldots, |N_p|$, let N_p^z be the set that consists of the first z voters in this order. Note that $\beta(N_p^z) = \ell$.

We claim that $\mathbf{o}$ provides w-EJR if and only if for each $p \in P$ and each $z = 1, \ldots, |N_p|$ it holds that $sat_i(\mathbf{o}) \geq \alpha(N_p^z)$ for some $i \in N_p^z$. Note that this means that we can verify whether $\mathbf{o}$ provides w-EJR by considering at most mn groups of voters and performing a polynomial amount of computation for each group, i.e., in polynomial time.

Now, if there exists a $p \in P$ and $z \in \{1, \ldots, |N_p|\}$ such that $sat_i(\mathbf{o}) < \alpha(N_p^z)$ for all $i \in N_p^z$, then $\mathbf{o}$ fails to provide w-EJR, as witnessed by N_p^z.

Conversely, if there is a set N' witnessing that $\mathbf{o}$ fails to provide w-EJR, we have $\beta(N') = \ell$ and hence there exists a candidate $p \in P$ such that $p \in \cap_{i\in N'} s_{i,1}$; in particular, this implies $N' \subseteq N_p$. Let $z = |N'|$, and let $u = \max_{i\in N'} sat_i(\mathbf{o})$; since N' witnesses that $\mathbf{o}$ fails to provide w-EJR, we have $u < \alpha(N')$. Since $\beta(N') = \beta(N_p^z)$ and $|N'| = z = |N_p^z|$, we have $\alpha(N') = \alpha(N_p^z)$. Also, by Observation 7.5.2 (applied to the satisfactions of voters in N' and N_p^z), for each $j \in N_p^z$ we have $sat_j(\mathbf{o}) \leq u < \alpha(N') = \alpha(N_p^z)$, i.e., N_p^z is also a witness that $\mathbf{o}$ fails to provide w-EJR. As our algorithm checks N_p^z, it will be able to detect that $\mathbf{o}$ fails to provide w-EJR.

To verify whether $\mathbf{o}$ provides w-PJR or w-JR, we modify the checks that the algorithm performs for each group of voters, just as in the proof of Proposition 7.5.3; we omit the details. $\square$

For strong notions of justified representation, we also obtain easiness-of-verification results, though the proof requires an additional level of complexity.

Theorem 7.6.2. *Given a monotonic election $(P, N, \ell, (\mathbf{s}_i)_{i\in N})$ together with an outcome $\mathbf{o}$, for each of $X \in \{JR, PJR, EJR\}$ we can decide in polynomial time whether $\mathbf{o}$ provides X.*

Proof. Fix a monotonic election $(P, N, \ell, (\mathbf{s}_i)_{i\in N})$ and an outcome $\mathbf{o}$. We can assume without loss of generality that $P = \cup_{i\in N} s_{i,\ell}$: if a candidate receives no approvals in the last round, it receives no approvals at all, and can be removed.

Now, for each $p \in P$ and each $t \in [\ell]$ let $N_{p,t} = \{i \in N : p \in s_{i,t}\}$. Note that all voters in $N_{p,t}$ approve p in each of the rounds $t, t+1, \dots, \ell$, so $\beta(N_{p,t}) \geq \ell - t + 1$. Order the voters in $N_{p,t}$ according to their satisfaction from $\mathbf{o}$ in non-decreasing order, breaking ties lexicographically. For each $z = 1, \dots, |N_{p,t}|$, let $N^z_{p,t}$ be the set that consists of the first z voters in this order.

We claim that $\mathbf{o}$ provides EJR if and only if for each $p \in P$, $t \in [\ell]$ and each $z = 1, \dots, |N_{p,t}|$ it holds that $sat_i(\mathbf{o}) \geq \alpha(N^z_{p,t})$ for some $i \in N^z_{p,t}$. This means that we can verify whether $\mathbf{o}$ provides EJR by considering at most $m\ell n$ groups of voters and performing a polynomial amount of computation for each group, i.e., in polynomial time.

Indeed, if there exist $p \in P$, $t \in [\ell]$ and $z \in \{1, \dots, |N_{p,t}|\}$ such that $sat_i(\mathbf{o}) < \alpha(N^z_{p,t})$ for all $i \in N^z_{p,t}$, then $\mathbf{o}$ fails to provide EJR, as witnessed by $N^z_{p,t}$.

Conversely, suppose there is a set N' witnessing that $\mathbf{o}$ fails to provide EJR. Let t be the first round such that $\cap_{i\in N'} s_{i,t} \neq \varnothing$, and let p be some candidate in $\cap_{i\in N'} s_{i,t}$. Note that $p \in \cap_{i\in N'} s_{i,\tau}$ for all $\tau = t, \dots, n$ and hence $N' \subseteq N_{p,t}$. Moreover, we have $\beta(N') = \ell - t + 1$: the voters in N' agree on p in rounds $t, \dots, \ell$, and, by the choice of t, there is no candidate they all agree on in earlier rounds.

Let $z = |N'|$, and let $u = \max_{i\in N'} sat_i(\mathbf{o})$; since N' witnesses that $\mathbf{o}$ fails to provide EJR, we have $u < \alpha(N')$. Since $\beta(N') = \ell - t + 1 \leq \beta(N^z_{p,t})$ and $|N'| = z = |N^z_{p,t}|$, we have $\alpha(N') \leq \alpha(N^z_{p,t})$. Also, by Observation 7.5.2 (applied to the satisfactions of voters in N' and $N^z_{p,t}$), it follows that for each $j \in N^z_{p,t}$ we have $sat_j(\mathbf{o}) \leq u < \alpha(N') \leq \alpha(N^z_p)$, i.e., $N^z_{p,t}$ is also a witness that $\mathbf{o}$ fails to provide EJR. As our algorithm checks $N^z_{p,t}$, it will be able to detect that $\mathbf{o}$ fails to provide EJR.

To verify whether $\mathbf{o}$ provides PJR or JR, we modify the checks that the algorithm performs for each group of voters, just as in the proof of Proposition 7.5.3; we omit the details. $\square$

The reader may wonder why we did not consider a more general model, where candidates' availability may change from round to round, but voters' preferences over available candidates are static: if voter i approves candidate p in some round where p is available, then i approves p whenever it is available. However, it turns out that the resulting "restricted availability" model is as general as the model we considered throughout the chapter. To see this, note that given an instance $(P, N, \ell, (\mathbf{s}_i)_{i\in N})$ of our standard model, we can modify it by setting $N' = N$, $P' = \{(p, t) : p \in P, t \in [\ell]\}$, $\ell' = \ell$, and for each $i \in N$ and $t \in [\ell]$ setting $s'_{i,t} = \{(p, t) : p \in s_{i,t}\}$. The modified instance is essentially equivalent to the original instance, but every candidate only receives approvals in a single round, so it can be seen as an instance of the restricted availability model.

7.7 Finding EJR Outcomes

So far, we focused on checking whether an outcome provides a representation guarantee. We will now switch gears, and explore the problem of finding a fair outcome.

7.7.1 Greedy Cohesive Rule Provides EJR

Chandak et al. [2024] show that every temporal election admits an outcome that provides EJR, by adapting the PAV rule [Thiele, 1895] and its local search variant ls-PAV [Aziz et al., 2018] from the multiwinner setting to the temporal setting; the local search-based approach results in a rule that is polynomial-time computable. However, they leave it as an open question whether another prominent voting rule, namely, the Greedy Cohesive Rule (GCR) [Bredereck et al., 2019, Peters et al., 2021] can be adapted to the temporal setting so as to provide EJR.

Bulteau et al. [2021] describe an algorithm that is similar in spirit to GCR and constructs a PJR outcome; however, unlike GCR, the algorithm of Bulteau et al. [2021] proceeds in two stages. We will now describe a two-stage procedure that is inspired by the algorithm of Bulteau et al. (but differs from it) and always finds EJR outcomes.

Theorem 7.7.1. *Algorithm 9 always outputs an outcome that provides EJR, and runs in time $O(2^n \cdot poly(n, m, \ell))$.*

Proof. The bound on the running time of the algorithm follows from the fact that the size of $\mathcal{V}$ is $O(2^n)$, and the amount of computation performed for each subset in $\mathcal{V}$ is polynomial in the input size. We now focus on correctness.

Algorithm 9: 2-Stage Greedy Cohesive Rule.

Input: Set of voters $N = \{1, \dots, n\}$, set of candidates $P = \{p_1, \dots, p_m\}$, number of rounds ℓ, voters' approval sets $(\mathbf{s}_1, \dots, \mathbf{s}_n)$;
$\mathbf{o} \leftarrow (p_1, \dots, p_1)$;
$T \leftarrow [\ell]$;
$\mathcal{V} := \{V \subseteq N : \beta(V) > 0\}$;
$\mathcal{V}^+ = \varnothing$;
while $\mathcal{V} \neq \varnothing$ **do**
    Select a set $V \in \arg\max_{V \in \mathcal{V}} \alpha(V)$ with ties broken arbitrarily;
    $\mathcal{V}^+ \leftarrow \mathcal{V}^+ \cup \{V\}$;
    for $V' \in \mathcal{V}$ **do**
        if $V \cap V' \neq \varnothing$ **then**
            $\mathcal{V} \leftarrow \mathcal{V} \setminus \{V'\}$;
Sort $\mathcal{V}^+$ as $V_1, \dots, V_q$ so that $\beta(V_1) \leq \dots \leq \beta(V_q)$;
for $k = 1, \dots, q$ **do**
    Let T' be a subset of $\{t \in T : \cap_{i \in V_k} s_{i,t} \neq \varnothing\}$ of size $\alpha(V_k)$;
    $T \leftarrow T \setminus T'$;
    for $t \in T'$ **do**
        Pick $p \in \cap_{i \in V_k} s_{i,t}$ and set $o_t \leftarrow p$;
return *outcome* $\mathbf{o}$;

Note that all sets in $\mathcal{V}^+$ are pairwise disjoint: for each $k \in [q]$, when we place V_k in $\mathcal{V}^+$, we remove all sets that intersect V_k from $\mathcal{V}$, and therefore no such set can be added to $\mathcal{V}^+$ at a future iteration. We will use this observation to argue that, while processing the sets in $\mathcal{V}^+$, for each set V_k we consider there exist $\alpha(V_k)$ rounds in which all members of V_k agree on at least one candidate. Suppose for a contradiction that this is not the case, and let $k \in [q]$ be the first index such that there are fewer than $\alpha(V_k)$ slots available out of the $\beta(V_k)$ rounds voters in V_k agree on. This means that strictly more than $\beta(V_k) - \alpha(V_k)$ of these slots have been taken up in previous iterations, and hence $\sum_{r=1}^{k} \alpha(V_r) > \beta(V_k)$. Since $\beta(V_k) \geq \beta(V_r)$ for all $r \leq k$, the inequality implies

$$\sum_{r=1}^{k} \frac{|V_r|}{n} \cdot \beta(V_k) \geq \sum_{r=1}^{k} \frac{|V_r|}{n} \cdot \beta(V_r) \geq \sum_{r=1}^{k} \alpha(V_r) > \beta(V_k), \tag{7.1}$$

and hence $\sum_{r=1}^{k} |V_r| > n$, a contradiction with the fact the groups $V_1, \dots, V_{k-1}, V_k$ are pairwise disjoint, and $|V| = n$.

We conclude that for each $V \in \mathcal{V}^+$ it holds that $sat_i(\mathbf{o}) \geq \alpha(V)$ for each $i \in V$. Hence, no set in $\mathcal{V}^+$ can be a witness that EJR is violated. Further, each set $V \in 2^N \setminus \mathcal{V}$

has $\beta(V) = 0$ and hence $\alpha(V) = 0$, so it cannot witness that EJR is violated either. It remains to consider sets in $\mathcal{V} \setminus \mathcal{V}^+$.

Fix a set $V' \in \mathcal{V} \setminus \mathcal{V}^+$, and suppose that it was deleted from $\mathcal{V}$ when some V was placed in $\mathcal{V}^+$ (and hence $\alpha(V') \leq \alpha(V)$). Moreover, when processing $\mathcal{V}^+$, the algorithm ensured that the satisfaction of each voter $i \in V$ is at least $\alpha(V) = \lfloor \frac{\beta(V)\cdot|V|}{n} \rfloor$. As $V \cap V' \neq \varnothing$, this means that the satisfaction of some voter $i \in V'$ is at least $\alpha(V) \geq \alpha(V')$. Thus, V' cannot be a witness that EJR is violated. $\square$

We note that for monotonic elections Algorithm 9 can be modified to run in polynomial time. Indeed, the proof of Theorem 7.6.2 shows that, when verifying representation axioms, it suffices to consider sets of the form $N^z_{p,t}$ (defined in the proof; see the appendix for details). If we modify Algorithm 9 so that initially it only places sets of this form in $\mathcal{V}$, it will only have to consider $O(m\ell n)$ sets, each of them in polynomial time; on the other hand, by Theorem 7.6.2 its output would still provide EJR.

While GCR runs in exponential time, it is a simple rule that can often be used to prove existence of fair outcomes; e.g., in the multiwinner setting, its outputs satisfy the stronger FJR axiom Peters et al. [2021], which is not satisfied by PAV or the Method of Equal Shares (MES). While FJR seems difficult to define for the temporal setting, this remains an interesting direction for future work, and GCR may prove to be a relevant tool. Moreover, one may want to combine JR-like guarantees with additional welfare, coverage, or diversity guarantees; the associated problems are likely to be NP-hard, so GCR may be a useful starting point for an approximation or a fixed-parameter algorithm (that may be more practical than the ILP approach). Finally, compared to PAV and MES, GCR may be easier to adapt to general monotone valuations (this is indeed the case in the multiwinner setting).

7.7.2 An ILP For Finding EJR Outcomes

We will now describe an algorithm for finding EJR outcomes that is based on integer linear programming (ILP). While this algorithm does not run in polynomial time, it is very flexible: e.g., we can easily modify it so as to find an EJR outcome that maximizes the utilitarian social welfare, or provides satisfaction guarantees to individual voters.

Theorem 7.7.2. *There exists an integer linear program (ILP) whose solutions correspond to outcomes that provide EJR; the number of variables and the number of constraints of this ILP are bounded by a function of the number of voters n.*

Proof. Consider an election $E = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$.

We first establish that we can assume $|P| \le 2^n$. Indeed, fix a subset of voters $V \subseteq N$ and a round $t \in [\ell]$. If there are two candidates p, p' such that for every $i \in N$ we have either $p, p' \in s_{i,t}$ or $p, p' \notin s_{i,t}$ then we can remove p' from the approval sets of all voters in N at round t: every outcome $\mathbf{o}$ that provides EJR and satisfies $o_t = p'$ can be modified by setting $o_t = p$ instead, without affecting representation. That is, we can assume that each candidate at round t is uniquely identified by the set of voters who approve her. It follows that we can modify the input election E by replacing P with $P' = \{p_V : V \subseteq N\}$ and modifying the voters' approval sets accordingly: for each $i \in N$, $t \in [\ell]$ we place p_V in $s'_{i,t}$ if and only if the approval set $s_{i,t}$ contains a candidate p with $\{i' : p \in s_{i',t}\} = V$. The modified election $E' = (P', N, \ell, (\mathbf{s}'_i)_{i \in N})$ satisfies $|P'| \le 2^n$, and an outcome $\mathbf{o}'$ of E' that provides EJR can be transformed into an outcome $\mathbf{o}$ of E that provides EJR as follows: for each $t \in [\ell]$, we set $o_t = p$ for some p such that $o'_t = p_V$ and $p \in \cap_{i \in V} s_{i,t}$. Thus, from now on we will assume that $|P| \le 2^n$.

Now, note that each round can be characterized by n voters' approval sets, i.e., a list of length n whose entries are subsets of P. Let $\mathbf{T}$ denote the set of all such lists; we will refer to elements of $\mathbf{T}$ as round *types*. There are 2^m possibilities for each voter's approval set, and hence $|\mathbf{T}| = (2^m)^n$. For each $\tau \in \mathbf{T}$, let $\kappa_\tau \in \mathbb{Z}_{\ge 0}$ be the number of rounds of type τ. Overloading notation, we write $p \in s_{i,\tau}$ if $p \in s_{i,t}$ for a round t of type τ.

Our ILP will have a variable $x_{p,\tau}$ for each $\tau \in \mathbf{T}$ and each $p \in P$, i.e., at most $m \cdot (2^m)^n \le 2^n \cdot (2^{2^n})^n$ variables: the variable $x_{p,\tau}$ indicates how many times we choose candidate p at a round of type τ and takes values in $0, \ldots, \kappa_\tau$. For each $\tau \in \mathbf{T}$, we introduce a constraint

$$\sum_{p \in P} x_{p,\tau} = \kappa_\tau; \tag{7.2}$$

these constraints guarantee that solutions to our ILP encode valid outcomes.

It remains to introduce constraints encoding the EJR axiom. To this end, for each group of voters V we add a constraint saying that at least one voter in V has satisfaction at least $\alpha(V)$.

Observe that the satisfaction of voter i from rounds of type τ can be written as $\sum_{p \in s_{i,\tau}} x_{p,\tau}$; thus, the total satisfaction of i is given by $\sum_{\tau \in \mathbf{T}} \sum_{p \in s_{i,\tau}} x_{p,\tau}$.

Next, for each $V \subseteq N$ with $V \ne \varnothing$ and each $i \in V$, we introduce a variable $\xi_{i,V}$; these variables take values in $\{0, 1\}$ and indicate which voter $i \in V$ receives satisfaction at least $\alpha(V)$. The constraints

$$0 \leq \xi_{i,V} \leq 1, \quad \sum_{i \in V} \xi_{i,V} \geq 1 \qquad \text{for all } V \in 2^N \setminus \{\varnothing\} \tag{7.3}$$

guarantee that for every nonempty subset of voters V we have $\xi_{i,V} = 1$ for at least one voter $i \in V$. Now, we can capture EJR by adding constraints

$$\sum_{\tau \in \mathbf{T}} \sum_{p \in s_{i,\tau}} x_{p,\tau} \geq \alpha(V) \cdot \xi_{i,V} \text{ for all } V \in 2^N \setminus \{\varnothing\}, i \in V. \tag{7.4}$$

Indeed, constraint (7.4) ensures that the satisfaction of at least one voter in V (one with $\xi_{i,V} = 1$) is at least $\alpha(V)$.

We conclude that every feasible solution to the ILP given by the constraints (7.2)–(7.4) encodes an outcome that provides EJR; moreover, both the number of variables and the number of constraints in this ILP can be bounded by functions of n. □

The following corollary illustrates the power of the ILP-based approach.

Corollary 7.7.3. *There is an FPT algorithm with respect to the number of voters that, given an election $E = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$ and a set of integers $\delta_1, \ldots, \delta_n$, decides whether there exists an EJR outcome of E that guarantees satisfaction δ_i to voter i for each $i \in N$, and, if yes, finds an outcome that maximizes the utilitarian social welfare among all outcomes with this property.*

Proof. The ILP in the proof of Theorem 7.7.2 contains an expression encoding each voter's satisfaction. We can modify this ILP by adding constraints saying that the satisfaction of voter i is at least δ_i, and adding a goal function that maximizes the sum of voters' satisfactions. The resulting ILP admits an algorithm that is FPT with respect to n by the classic result of Lenstra Jr [1983]. □

We note that, while Chandak et al. [2024] show that ls-PAV rule can find EJR outcomes in polynomial time, their approach cannot handle additional constraints on voters' welfare; hence Corollary 7.7.3 is not implied by their result.

7.7.3 An Impossibility Result for EJR in the (Semi-)Online Setting

In their analysis, Chandak et al. [2024] distinguish among (1) the *online setting*, where the number of rounds ℓ is not known, voters' preferences are revealed round-by-round, and o_t is selected as soon as all $(s_{i,t})_{i \in N}$ are revealed, (2) the *semi-online setting*, where

ℓ is known, but preferences are revealed round-by-round and an outcome for a round needs to be chosen as soon as the preferences for that round have been revealed, and (3) the *offline setting*, where we select the entire outcome $\mathbf{o}$ given full access to $(\mathbf{s}_i)_{i\in N}$. The PAV rule and its local search variant, the GCR rule and the ILP approach only work in the offline setting. In contrast, Chandak et al. [2024] show that a variant of MES satisfies w-EJR in the semi-online setting, but not in the online setting. They left open the existence of rules satisfying EJR in the semi-online setting. Here, we resolve this open question and show that no rule can satisfy EJR in the semi-online setting (and hence in the online setting).

Proposition 7.7.4. *No rule satisfies EJR in the semi-online setting.*

Proof. Consider an instance with an even number of agents $n = 2k$, $k \geq 4$, candidate set $P = \{p_1, \dots, p_n\}$, and n rounds. We construct the approval sets as follows.

$s_{i,t}$	1	$\dots$	k	$k+1$	$\dots$	n
1	$\{p_1\}$	$\dots$	$\{p_1\}$	$\{p_n\}$	$\dots$	$\{p_n\}$
2	$\{p_2\}$	$\dots$	$\{p_2\}$	$\{p_n\}$	$\dots$	$\{p_n\}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
k	$\{p_k\}$	$\dots$	$\{p_k\}$	$\{p_n\}$	$\dots$	$\{p_n\}$
$k+1$	$\{p_{k+1}\}$	$\dots$	$\{p_{k+1}\}$	$\{p_1\}$	$\dots$	$\{p_1\}$
$k+2$	$\{p_{k+2}\}$	$\dots$	$\{p_{k+2}\}$	$\{p_2\}$	$\dots$	$\{p_2\}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
n	$\{p_n\}$	$\dots$	$\{p_n\}$	$\{p_k\}$	$\dots$	$\{p_k\}$

By symmetry, we can assume without loss of generality that in the first k rounds, we select $o_t = p_t$ for $t \in [k]$. Note that for each $i = k+1, \dots, n$ the group $N' = \{i\}$ satisfies $\beta(N') = n$ and hence $\alpha(N') = 1$. Thus, in order to satisfy w-JR, in the next $n/2$ rounds, we must select each of $p_1, \dots, p_k$ exactly once. However, the resulting outcome $\mathbf{o}$ fails EJR: indeed, $N'' = [k]$ satisfies $\beta(N'') = k$, $\alpha(N'') = k/2 \geq 2$, while for each $i \in N''$ we have $sat_i(\mathbf{o}) = 1$. $\square$

7.8 Conclusion

We have explored the complexity of verifying whether a given outcome of a temporal election satisfies one of the six representation axioms considered by Chandak et al. [2024]. We have obtained coNP-hardness results even for very restricted special cases of this problem: e.g., for strong versions of the axioms verification remains hard even if there are just two candidates. We complement these hardness results with parameterized complexity results and a positive result for a structured setting, where candidates join the pool over time and never leave. We also describe an ILP that can be used to find outcomes that provide EJR and satisfy additional constraints. Finally, we answer an open question of Chandak et al. [2024], by showing that a variant of the Greedy Cohesive Rule provides EJR.

Possible directions for future work (in addition to open problems listed in Section 7.4) include considering stronger variants of proportionality, such as FJR [Peters et al., 2021] or core stability [Aziz et al., 2017], exploring other domain restrictions, and extending the temporal framework to the broader setting of participatory budgeting [Aziz and Lee, 2021, Lackner et al., 2021, Peters et al., 2021].

Chapter 8

Fair Scheduling

8.1 Overview

In this final chapter, we study a special case of the temporal voting model in which outcomes are restricted to *permutations*. This setting captures the essence of a *scheduling* problem (i.e., each project can be chosen at only one timestep). We begin with a motivating example.

In July 2022, NASA's James Webb Space Telescope recorded breathtaking images of the distant cosmic cliffs and the glittering landscape of star birth in fine detail. The stunning images were made possible by the advancement of space exploring technologies, prompting widespread awe both within the scientific community and the world at large. Unsurprisingly, it is clear that no regular institution, let alone individual, could afford such highly specialized equipment for their own (research) endeavours. The new discoveries were funded by a 10 billion-dollar investment from NASA and the USA.

NASA, whose goal is to "expand knowledge for the benefit of humanity", could choose to loan its facilities (not just this telescope, but others as well) to (possibly independent) research institutes to further the exploration of space, maybe with fresh perspectives. In fact, on a smaller scale, in 2008, libraries in the US trialed the idea for loaning out telescopes, pioneered by the New Hampshire Astronomical Society. The movement has proved to be a great success, and currently, over 100 libraries across the country participate in this program. The city of Westminster in the UK also has a similar program.

Now, each institution may have their preferred schedule of equipment rental—some of them may want to see certain specific galaxies or phenomena, and due to natural conditions, they can only view them during certain time periods. Then, the goal of NASA is to come up with a schedule for the loan of their observatories and telescopes. In doing so, multiple goals may be considered. For instance, if maximal utilization is the primary goal, then maximizing *utilitarian* welfare may be desirable, to fully utilize the telescope by filling up its schedule as much as possible. However, NASA may wish to give different institutions the opportunity to gain access to the facility and hopefully bring in new perspectives, and thus, may wish to treat different institutions fairly. Thus, other welfare objectives, such as maximizing *egalitarian* welfare, or fairness notions such as *equitability* and *proportionality* may become more relevant instead.

Other scenarios where different agents' preferences over schedules (i.e., assignment of projects to timesteps) need to be aggregated into a common schedule include scheduling university lectures, conference talks or popular entertainment. Inspired by these applications, in this work, we study the problem faced by a central authority in creating a common schedule in a fair and efficient manner. In particular, we revisit the computational problems associated with achieving outcomes that maximize two welfare notions (utilitarian welfare and egalitarian welfare) or satisfy proportionality. We also study another fairness notion not considered previously—*equitability*.

8.2 Preliminaries

In this setting, we further assume that the number of timesteps is at most the number of projects, i.e., $\ell \leq m$. This is to ensure the validity of a permutation as an outcome. We also require that for any $r, r' \in [\ell]$ with $r \neq r'$, we have $o_r \neq o_{r'}$; that is, each project is built at most once.

We also introduce additional notation that would simplify our analysis. We can represent agents' preferences as graphs: for each agent $i \in N$, let her *approval graph* be G^i, a bipartite graph with parts P and $[\ell]$ that contains an edge $(p_j, r) \in P \times [\ell]$ if and only if $p_j \in s_{i,r}$. We let $G^i_{j,r} = 1$ if $(p_j, r) \in G^i$, and $G^i_{j,r} = 0$ otherwise. Let μ_i be the size of a maximum matching in G^i.

Note that an outcome maximizes agent i's utility if and only if it corresponds to a maximum matching in G^i, i.e., provides her a utility of μ_i.

Preference restrictions In general, our framework allows for agents to approve of any number of projects between 0 and m for each timestep. From this, we can derive three natural restricted cases of our model: (1) limiting the number of approved projects for *each* timestep, (2) limiting the *total* number of approved projects across all timesteps, and (3) limiting the *number of times* a project can be approved overall.

The special case where each agent approves exactly one project for each timestep and each project is approved exactly once by each agent is called the *full permutation* (FP) setting. The setting where for each $i \in N$ each project appears in at most one set $s_{i,r}$ and $|\cup_{r\in[\ell]} s_{i,r}| \leq k$ for some $k \in [m]$ is called the *k-partial permutation* (k-PP) setting.

The goal of this work is to study the algorithmic complexity of identifying good outcomes in our model. There are several ways to define what it means for an outcome to be *good*. In what follows, we formally define the notions that will be considered in the remainder of the chapter.

Perhaps the most straightforward approach is to focus on outcomes that optimize some notion of welfare, as seen in previous chapters.

Definition 8.2.1 (Utilitarian Social Welfare)**.** Given an instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i\in N})$, an outcome $\mathbf{o}$ *maximizes utilitarian social welfare (Max-UTIL)* if for every outcome $\mathbf{o}' \in \Pi(\mathcal{I})$ it holds that $\sum_{i\in N} u_i(\mathbf{o}) \geq \sum_{i\in N} u_i(\mathbf{o}')$.

Definition 8.2.2 (Egalitarian Social Welfare)**.** Given an instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i\in N})$, an outcome $\mathbf{o}$ *maximizes egalitarian social welfare (Max-EGAL)* if for every outcome $\mathbf{o}' \in \Pi(\mathcal{I})$ it holds that $\min_{i\in N} u_i(\mathbf{o}) \geq \min_{i\in N} u_i(\mathbf{o}')$.

Instead of maximizing the welfare, we may want to focus on equity: can we obtain an outcome that guarantees each agent the same utility? To capture this idea, we define the following notion of fairness.

Definition 8.2.3 (Equitability)**.** Given an instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i\in N})$, an outcome $\mathbf{o}$ is *equitable (EQ)* if for all $i, i' \in N$ it holds that $u_i(\mathbf{o}) = u_{i'}(\mathbf{o})$.

Note that an equitable outcome does not always exist (this is also the case for many similar fairness properties in the social choice literature [Brandt et al., 2016]). Consider the simple case of two agents, with approval sets $\mathbf{s}_1 = (p_1, p_2)$ and $\mathbf{s}_2 = (p_2, p_1)$ respectively. Then, any outcome $\mathbf{o}$ will give a utility of 2 to one agent, and 0 to the other—equal treatment of these individuals is not achievable in our model.

We will also consider proportionality, as discussed in previous chapters. Intuitively, this property demands that each agent's utility should be at least as high as her proportional fair share. In our setting, given an instance $\mathcal{I}$, it is natural to define an agent's proportional fair share as her best-case utility $\max_{\mathbf{o}\in\Pi(\mathcal{I})} u_i(\mathbf{o}) = \mu_i$, divided by the number of agents n. Thus, we define proportionality as follows.

Definition 8.2.4 (Proportionality)**.** Given an instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i\in N})$, an outcome $\mathbf{o}$ is *proportional (PROP)* if for all $i \in N$ it holds that $u_i(\mathbf{o}) \geq \frac{\mu_i}{n}$.

8.3 Utilitarian Social Welfare

The utilitarian social welfare is perhaps the most popular optimization target in multi-agent allocation problems. We start by formalizing the problem of computing an outcome that maximizes this measure of welfare.

> UTILITARIAN SOCIAL WELFARE MAXIMIZATION (UTIL):
> **Input**: An instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$, and a parameter $\lambda \in \mathbb{N}$.
> **Question**: Is there an outcome $\mathbf{o}$ such that $\sum_{i \in N} u_i(\mathbf{o}) \geq \lambda$?

We will now show that the problem we just defined admits an efficient algorithm; in fact, we can even compute an outcome that maximizes the utilitarian social welfare in polynomial time.

Theorem 8.3.1. UTIL *is solvable in polynomial time.*

Proof. Given an instance of UTIL, we construct a weighted complete bipartite graph with parts P and ℓ, where the weight of the edge (p_j, r) equals to the number of agents that approve implementing p_j at timestep r: that is, we set $w(p_j, r) = |\{i \in N : p_j \in s_{i,r}\}|$. Then a maximum-weight matching in this graph corresponds to an outcome that maximizes the utilitarian social welfare. It remains to observe that a maximum-weight matching in a bipartite graph can be computed in polynomial time [Schrijver, 2003]. □

8.4 Egalitarian Social Welfare

Next, we consider the complexity of maximizing the egalitarian welfare. Again, we first formulate the associated decision problem.

> EGALITARIAN SOCIAL WELFARE MAXIMIZATION (EGAL):
> **Input**: An instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$, and a parameter $\lambda \in \mathbb{N}$.
> **Question**: Is there an outcome $\mathbf{o}$ such that $u_i(\mathbf{o}) \geq \lambda$ for each $i \in N$?

It turns out that EGAL is NP-complete. In fact, this hardness result holds even in the FP setting.

Our proof proceeds by a reduction from the BINARY CLOSEST STRING PROBLEM (BCSP) [Frances and Litman, 2007, Lanctot et al., 2003]. An instance of this problem consists of ν binary strings of length ρ each, and an integer κ; it is a yes-instance if there exists a binary string y of length ρ such that the Hamming distance (i.e., number of *mismatches*) between y and each of the ν strings is at most κ (equivalently, the number of matches is at least $\rho - \kappa$), and a no-instance otherwise. This problem is known to be NP-complete [Frances and Litman, 2007, Lanctot et al., 2003].

We are now ready to establish the complexity of EGAL.

Theorem 8.4.1. EGAL *is* NP-*complete. The hardness result holds even in the FP setting.*

Proof. It is clear that EGAL is in NP: given an outcome, we can evaluate each agent's utility and compare it to λ. To establish NP-hardness, we give a reduction from BCSP.

Consider an instance of BCSP given by ν binary strings $X = \{x_1, \dots, x_\nu\}$ of length ρ each, and an integer κ. For each $i \in [\nu]$, $j \in [\rho]$, denote the j-th bit of the string x_i by x_{ij}. To create an instance of EGAL, we introduce 2ρ projects $p_1, \dots, p_{2\rho}$ and 2ρ timesteps $1, \dots, 2\rho$. We will encode the bit strings as agents' preferences: for each x_{ij}, if $x_{ij} = 1$, let $s_{i,2j-1} = p_{2j-1}$ and $s_{i,2j} = p_{2j}$; if $x_{ij} = 0$, let $s_{i,2j-1} = p_{2j}$ and $s_{i,2j} = p_{2j-1}$.

Let $\lambda = 2(\rho - \kappa)$. We will now prove that there exists an outcome $\mathbf{o}$ that gives each agent a utility of at least λ if and only if there exists a binary string y of length ρ such that its Hamming distance to each string in X is at most κ (i.e., the number of matches is at least $\rho - \kappa$).

For the 'if' direction, let $y = (y_1, \dots, y_\rho)$ be a string with at most κ mismatches to each of the strings in X. Construct an outcome $\mathbf{o}$ by setting $o_{2j-1} = p_{2j-1}$, $o_{2j} = p_{2j}$ if $y_j = 1$, and $o_{2j-1} = p_{2j}$, $o_{2j} = p_{2j-1}$ if $y_j = 0$. Consider an agent i. For each bit j such that $x_{ij} = y_j$ we have $s_{i,2j-1} = o_{2j-1}$, $s_{i,2j} = o_{2j}$. Thus, the utility of this agent is at least $2(\rho - \kappa) = \lambda$, which is what we wanted to prove.

For the 'only if' direction, suppose there exists an outcome that gives each agent a utility of λ. We will say that an outcome $\mathbf{o}$ is *proper* if for each $j \in [\rho]$ we have $\{o_{2j-1}, o_{2j}\} = \{p_{2j-1}, p_{2j}\}$. We claim that there exists a proper outcome that gives each agent a utility of λ. Indeed, suppose that this is not the case. For each outcome $\mathbf{o}$, let $z(\mathbf{o})$ be the number of timesteps q such that $q \in \{2j-1, 2j\}$ for some $j \in [\rho]$, but $o_q \notin \{p_{2j-1}, p_{2j}\}$. Among all outcomes that give each agent a utility of λ, pick one with the minimal value of $z(\mathbf{o})$; let this outcome be $\mathbf{o}^*$. By our assumption, $\mathbf{o}^*$ is not proper, so $z(\mathbf{o}^*) > 0$. Thus, there exists a timestep q such that $q \in \{2j-1, 2j\}$ for some $j \in [\rho]$, but $o^*_q \notin \{p_{2j-1}, p_{2j}\}$. Then in $\mathbf{o}^*$ there is a project $p \in \{p_{2j-1}, p_{2j}\}$ that is scheduled at timestep $2j' - 1$ or $2j'$ for some $j' \neq j$. Modify $\mathbf{o}^*$ by swapping p with o^*_q. Note that in $\mathbf{o}^*$, no agent derives positive utility from either of these two projects. Hence, this swap cannot decrease any agent's utility, but it decreases $z(\cdot)$ (because the project at timestep q is now one of p_{2j-1}, p_{2j}), a contradiction with our choice of $\mathbf{o}^*$.

Now, fix a proper outcome $\mathbf{o}$ that gives each agent a utility of λ. Let

$$y_j = \begin{cases} 1 & \text{if } o_{2j-1} = p_{2j-1} \text{ and } o_{2j} = p_{2j} \\ 0 & \text{if } o_{2j-1} = p_{2j} \text{ and } o_{2j} = p_{2j-1} \end{cases}$$

Consider a string x_i. We know that agent i's utility from $\mathbf{o}$ is at least $\lambda = 2(\rho - \kappa)$. Note that for each $j \in [\rho]$ we have either (1) $o_{2j-1} = s_{i,2j-1}$, $o_{2j} = s_{i,2j}$ or (2) $o_{2j-1} \neq s_{i,2j-1}$, $o_{2j} \neq s_{i,2j}$. Hence, there can be at most κ indices $j \in [\rho]$ for which condition (2) holds, and therefore there are at most κ mismatches between x_i and y. □

Theorem 8.4.1 indicates that even for FP instances, it is hard to decide whether each agent can be guaranteed a utility of at least λ. This motivates us to consider a less ambitious goal: can EGAL be solved efficiently if λ is bounded by a small constant?

Perhaps surprisingly, even for $\lambda = 1$ and FP instances this is unlikely to be the case: we show that EGAL is as hard as the PERFECT COMPLETE BIPARTITE PROPER RAINBOW MATCHING (PCBP-RM) problem [Perarnau and Serra, 2011], one of the many variants of the RAINBOW MATCHING problem for which an NP-hardness result is conjectured, but has not been established [Aharoni et al., 2017].

An instance of PCBP-RM is given by a complete bipartite graph $K_{\nu,\nu}$ (i.e., there are ν nodes on each side of the graph, and each node on one side is connected to every other node on the opposite side), where edges are properly colored (i.e., if two edges share an endpoint, they have different colors). The goal is to decide whether this instance admits a perfect rainbow matching M, i.e., a matching of size ν in which every edge has a different color.

Theorem 8.4.2. EGAL *is at least as hard as* PCBP-RM*, even when restricted to FP instances with* $\lambda = 1$*.*

Proof. Consider an instance of PCBP-RM with two parts V_1 and V_2, $|V_1| = |V_2| = \nu$, where for all $i, j \in [\nu]$, the i-th vertex in V_1 is connected to the j-th vertex in V_2 via an edge e_{ij}; we denote the color of this edge by $\texttt{color}(e_{ij})$. There are a total of ν unique colors $C = \{c_1, \ldots, c_\nu\}$.

We construct an instance of EGAL that contains ν agents, ν projects, and ν timesteps. For each agent i and timestep r, we set $s_{i,r} = p_j$, where j is the index of the color of the edge e_{ir}, i.e., $\texttt{color}(e_{ir}) = c_j$.

We will now prove that there exists an outcome $\mathbf{o}$ that gives each agent a utility of at least 1 if and only if there exists a perfect rainbow matching M.

For the 'if' direction, let M be a perfect rainbow matching. We create an outcome $\mathbf{o}$ as follows. To determine the timestep for project p_j, we identify an edge of M that has color c_j; if this edge connects agent i and timestep r, we schedule p_j at time r (thereby providing utility 1 to agent i). Since M is a rainbow matching, each project is scheduled exactly once, and any two projects are assigned distinct timesteps. Moreover, as M is a matching, each agent's utility is 1.

For the 'only if' direction, consider an outcome $\mathbf{o}$ that gives each agent a utility of at least 1. Observe that for each $r \in [\nu]$ and every pair of agents i, i', we have $s_{i,r} \neq s_{i',r}$. This means that for each $r \in [\nu]$, there is exactly one agent that receives a utility of 1 from o_r, i.e., the utility of each agent is exactly 1. We construct the matching M as follows: if agent i receives utility from the project scheduled at r, we add an edge from the i-th vertex of V_1 to the r-th vertex of V_2 to M. Note that M is a matching: each vertex in V_1 is matched, as each agent receives utiity 1 from some project, and each vertex in V_2 is matched, as each timestep provides positive utility to at most one agent. Moreover, M is a rainbow matching, as each project is scheduled exactly once. $\square$

On a more positive note, for $\lambda = 1$ in the γ-PP setting we can characterize the complexity of EGAL with respect to the parameter γ. We do so by establishing a tight relationship between this problem and the k-SAT problem. An instance of k-SAT consists of ν Boolean variables and ρ clauses, where each clause has at most k literals; it is a yes-instance if there exists an assignment of Boolean values to the variables such that at least one literal in each clause evaluates to True, and a no-instance otherwise. This problem is NP-complete for each $k \geq 3$, but polynomial-time solvable for $k = 2$.

Theorem 8.4.3. EGAL *is* NP*-complete even when restricted to γ-PP instances with $\lambda = 1$, for any fixed $\gamma \geq 3$.*[1]

Proof. We describe a reduction from γ-SAT to EGAL restricted to γ-PP instances with $\lambda = 1$.

[1]It is important to note that this does not mean that when $\lambda = 1$, EGAL under FP is always NP-complete. We cannot use the $\gamma = m$ argument here, even if $m \geq 3$.

Consider an instance of γ-SAT given by ν Boolean variables $X = \{x_1, \dots, x_\nu\}$ and ρ clauses $\mathcal{C} = \{C_1, \dots, C_\rho\}$. In our instance of EGAL, we have a set of ρ agents $N = \{1, \dots, \rho\}$, 2ν projects $P = \{p_1, \dots, p_{2\nu}\}$, and 2ν timesteps. Each agent encodes a clause: for each $i \in N$, $j \in [\nu]$ we set

$$s_{i,2j-1} = \begin{cases} p_j & \text{if } C_i \text{ contains the positive literal } x_j \\ \varnothing & \text{otherwise} \end{cases}$$

and

$$s_{i,2j} = \begin{cases} p_j & \text{if } C_i \text{ contains the negative literal } \neg x_j \\ \varnothing & \text{otherwise} \end{cases}$$

As we start with an instance of γ-SAT, we have $| \cup_{r \in [2\nu]} s_{i,r}| \leq \gamma$ for each $i \in N$, i.e., we obtain a valid γ-PP instance.

We will now prove that there exists an outcome $\mathbf{o}$ that gives each agent a positive utility if and only if our instance of γ-SAT admits a satisfying assignment.

For the 'if' direction, consider a satisfying assignment $(\xi_j)_{j \in [\nu]}$. For $j \in [\nu]$, if ξ_j is set to True, let $o_{2j-1} = p_j$, $o_{2j} = p_{\nu+j}$ and if ξ_j is set to False, let $o_{2j-1} = p_{\nu+j}$, $o_{2j} = p_j$. Consider an agent $i \in [\rho]$. Since the assignment $(\xi_j)_{j \in [\nu]}$ satisfies C_i, there is a literal $\ell \in C_i$ that is satisfied by this assignment. If $\ell = x_j$ is a positive literal then ξ_j is set to True, so $o_{2j-1} = p_j$, and we have $s_{i,2j-1} = p_j$. If $\ell = \neg x_j$ is a negative literal then ξ_j is set to False, so $o_{2j} = p_j$, and we have $s_{i,2j} = p_j$. In either case, the utility of agent i is at least 1.

For the 'only if' direction, consider an outcome $\mathbf{o}$ that gives each agent a positive utility. Arguing as in the proof of Theorem 8.4.1, we can assume that for each $j \in [\nu]$ it holds that p_j is scheduled at $2j-1$ or at $2j$: if this is not the case for some $j \in [\nu]$, we can move p_j to one of these timesteps without lowering the utility of any agent. We construct a truth assignment $(\xi_j)_{j \in [\nu]}$ by setting ξ_j to True if $o_{2j-1} = p_j$ and to False if $o_{2j} = p_j$. Now, consider a clause C_i. Since the utility of agent i is at least 1, it follows that our assignment satisfies at least one of the literals in C_i. As this holds for all clauses, the proof is complete. □

Theorem 8.4.4. EGAL *is polynomial-time solvable when restricted to* 2*-PP instances with* $\lambda = 1$.

Proof. Consider an instance of EGAL with $\lambda = 1$ given by n agents $N = \{1, \dots, n\}$, m projects $P = \{p_1, \dots, p_m\}$ and ℓ timesteps. For each project $p_j \in P$ and timestep $r \in [\ell]$, create a variable x_{jr}; intuitively, we want this variable to evaluate to True if project p_j is scheduled at timestep r and to False otherwise.

First, we encode the fact that each project can be scheduled at most once: for each project $p_j \in P$ and each pair of timesteps $r, r' \in [\ell]$ with $r \neq r'$ we add the clause $\neg x_{jr} \vee \neg x_{jr'}$. Let the conjunction of these clauses be C^*.

Next, we encode the fact that in each timestep we have at most one project: for each timestep $r \in [\ell]$ and each pair of projects $p_j, p_{j'} \in P$ with $j \neq j'$ we add the clause $\neg x_{jr} \vee \neg x_{j'r}$. Let the conjunction of these clauses be C'.

Finally, for each agent $i \in [n]$, we create a clause that requires this agent to have positive utility. Specifically, for each $i \in [n]$ we create a clause C_i as follows. If there exists a single timestep r such that $s_{i,r} \neq \varnothing$, we set $C_i = x_{jr}$ if $s_{i,r} = \{p_j\}$ and $C_i = x_{jr} \vee x_{j'r}$ if $s_{i,r} = \{p_j, p_{j'}\}$. If there exists two timesteps r, r' such that $s_{i,r}, s_{i,r'} \neq \varnothing$, then it has to be the case that $s_{i,r} = p_j$, $s_{i,r'} = p_{j'}$ for some $p_j, p_{j'} \in P$, so we set $C_i = x_{jr} \vee x_{j'r'}$.

It is now easy to see that there exists a truth assignment that satisfies C^*, C', and all clauses in $C_1, \ldots, C_n$ if and only if there exists an outcome that guarantees positive utility to each agent. Moreover, each of the clauses in our construction is a disjunction of at most two literals. It remains to observe that 2-SAT is solvable in $\mathcal{O}(n + m)$ time [Krom, 1967, Aspvall et al., 1979, Even et al., 1975]. $\square$

8.5 Equitability

In Section 8.2, we have seen that not all instances admit equitable outcomes. We will now show that deciding the existence of equitable outcomes is computationally intractable.

> EQUITABILITY (EQ):
> **Input**: An instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$, and a parameter $\lambda \in \mathbb{N}$.
> **Question**: Is there an outcome $\mathbf{o}$ such that $u_i(\mathbf{o}) = \lambda$ for each $i \in N$?

Theorem 8.5.1. EQ *is* NP*-complete. The hardness result holds even in the PP setting.*

Proof. It is clear that EQ is in NP. To show that this problem is NP-hard, we first formulate an intermediate problem to be used in our proof. Namely, we introduce the EXACT PARTIAL BINARY CLOSEST STRING PROBLEM (EXACT-P-BCSP). This problem is similar to the BCSP, but with two differences: (1) the Hamming distance between the output string and each of the input strings must be exactly κ, and (2) we allow an additional character, $*$, in the solution string. Formally, an instance of

EXACT-P-BCSP consists of ν binary strings of length ρ each, and an integer κ; it is a yes-instance if there exists a string y of length ρ over the alphabet $\{0, 1, *\}$ such that the Hamming distance between y and each of the ν input strings is exactly κ (equivalently, the number of matches is exactly $\rho - \kappa$), and a no-instance otherwise.

We begin with the following lemma.

Lemma 8.5.2. EXACT-P-BCSP *is* NP-*hard.*

Proof. We prove the claim by reduction from EXACT-2-3SAT (X2-3SAT). In an instance of X2-3SAT, we are given a 3-CNF formula with n Boolean variables $x_1, \ldots, x_n$ and m clauses $C_1, \ldots, C_m$. In each clause, exactly two literals are evaluated to True. Note that X2-3SAT is equivalent to Exact-1-3SAT (where exactly one literal is evaluated to True), and the latter is known to be NP-hard [Schaefer, 1978].

Consider an instance of X2-3SAT, given by n variables and m clauses, each with at most 3 literals. We will encode the Boolean value assignment to variables in the bits of the strings. Consider strings of length $2n + 1$; for all $i \in [n]$, if $x_i = 1$, let $s_{i,2j-1} = 1$ and $s_{i,2j} = 0$; if $x_i = 0$, let $s_{i,2j-1} = 0$ and $s_{i,2j} = 1$.

Then, first create a string of the form $(00)^n1$ and n strings of the form $(00)^i11(00)^{n-i-1}1$, for all $i \in [n-1]$. We mandate that the distance from the solution string be of hamming distance n from each of these strings. We derive the following claims.

Claim 1: Each consecutive pair of bits $s_{i,2j-1}$ and $s_{i,2j}$ only admit bit combinations of the form 10 or 01. Equation that arises from the first string is

$$(1 - x_{2n+1}) + \sum_{j \in [2n]} x_j = n \tag{8.1}$$

Equation that arises from the i-th string (for $i = 2, \ldots, n+1$) is,

$$(1 - x_{2n+1}) + (1 - x_{2i-3}) + (1 - x_{2i-2}) + \sum_{j \in [2i] \setminus \{2i-3, 2i-2\}} x_j = n \tag{8.2}$$

Now, for each $i = 2, \ldots, n+1$, subtract (8.1) from (8.2), we get

$$1 - x_{2i-3} - x_{2i-3} + 1 - x_{2i-2} - x_{2i-2} = 0. \tag{8.3}$$

Dividing both sides of the equation by 2, we get

$$x_{2i-3} + x_{2i-2} = 1, \tag{8.4}$$

indicating that the bits of each pair should be 01 or 10.

Claim 2: The last bit must be 1. Suppose, for a contradiction, that the last bit of the solution is 0. This means that the first string has distance $n-1$ for the first $2n$ bits, and by the pigeonhole principle, at least one pair will be 00. Let such a pair be (x_{2i-1}, x_{2i}) for some $i \in [n]$. However, if we consider the $(i+1)$-th string, where the $(2i-1)$-th and $(2i)$-th bit is 11, the distance of any solution to this (by Claim 1) has to be $n-1$ (for the $n-1$ remaining pairs of 00) $+$ 2 (for the mismatch mentioned above) $+$ 1 (for the last bit) $= n+2 > n$, contradicting the fact that the distance has to be exactly n.

Next, for each of the m clauses, we create a string whose bits depends on what truth assignment to the variable(s) in the clause would make the literal positive: i.e. if the literal appears and is positive, we let the corresponding pair be 10, if it is negative, we let it be 01, and if it doesn't appear in the clause, we let it be 00. Also, let the last bit be 0. We mandate that the distance of the solution string to each of these strings be n.

Note that an implication of Claim 2 is that the null character cannot appear in the solution string.

Claim 3: There must be exactly two pairs of matching consecutive bit pairs. From Claim 2 above, the last bit must be 1, so we have a hamming distance of 1 from there. For $n-3$ pairs corresponding to literals not in the clause, the distance to each must be 1 (by Claim 1 above), thereby giving us a hamming distance of $n-3$. Then, we have three pairs remaining with total hamming distance of 2 (since the total distance must be n). Again by Claim 1 above, it must be that exactly two pairs match and one pair does not.

We are left with proving that there exists a binary string y of length m such that the number of mismatches is exactly $m-n$ if and only if there exists an assignment of Boolean values to variables such that exactly two literals in each clause evaluates to True.

By the three claims above, it is easy to see that the functions mapping the solutions between the problem are the same (by looking at the first $2n$ timesteps in Exact-P-Bcsp). □

We will now reduce Exact-P-BCSP to Eq. Consider an instance of Exact-P-BCSP given by ν binary strings $X = \{x_1, \ldots, x_\nu\}$ of length ρ each and an integer κ. For each $i \in [\nu]$, $j \in [\rho]$, denote the j-th bit of the i-th string by x_{ij}. We introduce 2ρ projects $p_1, \ldots, p_{2\rho}$ and 2ρ timesteps. We encode the bit strings in X as the agents' preferences: for each $i \in [\nu]$, $j \in [\rho]$, if $x_{ij} = 1$, let $s_{i,2j-1} = p_j$, $s_{i,2j} = \varnothing$; and if $x_{ij} = 0$, let $s_{i,2j-1} = \varnothing$, $s_{i,2j} = p_j$.

We will now prove that there exists an outcome $\mathbf{o}$ that gives each agent a utility of exactly $\rho - \kappa$ if and only if there exists a binary string y of length ρ such that the number of mismatches is exactly κ.

For the 'if' direction, let y be a solution string with exactly $\rho - \kappa$ matches to each of the strings in X. We construct an outcome $\mathbf{o}$ as follows. For each $j \in [\rho]$, if $y_j = 1$ we let $o_{2j-1} = p_j$ and if $y_j = 0$ we let $o_{2j} = p_j$; we assign the remaining projects to the remaining timesteps arbitrarily, with the constraint that if $y_j = *$, then p_j is not assigned to either of the timesteps $2j - 1$ or $2j$ (it is not difficult to verify that this can always be done efficiently; we omit the details). Consider an agent $i \in [\nu]$ and a timestep $j \in [\rho]$. If $x_{ij} = y_j = 1$ we have $o_{2j-1} = s_{i,2j-1}$, $s_{i,2j} = \varnothing$, if $x_{ij} = y_j = 0$ we have $o_{2j} = s_{i,2j}$, $s_{i,2j-1} = \varnothing$, and if $x_{ij} \neq y_j$ then $o_{2j-1} \notin s_{i,2j-1}$, $o_{2j} \notin s_{i,2j}$. Hence, each pair of timesteps $(2j - 1, 2j)$ such that $x_{ij} = y_j$ contributes exactly 1 to the utility of agent i, so $u_i(\mathbf{o}) = \rho - \kappa$.

For the 'only if' direction, consider any outcome $\mathbf{o}$ that gives each agent a utility of exactly $\rho - \kappa$. To construct the string y, for each $j \in [\rho]$ we set

$$y_j = \begin{cases} 1 & \text{if } o_{2j-1} = p_j \\ 0 & \text{if } o_{2j} = p_j \\ * & \text{otherwise} \end{cases}$$

Consider an agent i. Observe that her utility from the pair of timesteps $(2j-1, 2j)$ is at most 1; moreover, it is 1 if and only if (1) $s_{i,2j-1} = o_{2j-1} = p_j$ or (2) $s_{i,2j} = o_{2j} = p_j$. Condition (1) holds if and only if $x_{ij} = y_j = 1$, and condition (2) holds if and only if $x_{ij} = y_j = 0$. That is, agent i's utility from $(2j - 1, 2j)$ is 1 if and only if $x_{ij} = y_j$. Since we have $u_i(\mathbf{o}) = \rho - \kappa$ for each $i \in [\nu]$, this means that y has κ mismatches with every input string. □

8.6 Proportionality

Finally, we consider the complexity of finding proportional outcomes.

> PROPORTIONALITY (PROP):
> **Input**: An instance $\mathcal{I} = (P, N, \ell, (\mathbf{s}_i)_{i \in N})$.
> **Question**: Is there an outcome $\mathbf{o}$ such that $u_i(\mathbf{o}) \geq \mu_i/n$ for each $i \in N$?

It is easy to see that PROP does not necessarily imply UTIL or EGAL. Indeed, consider the case where $n = m = \ell$, and all agents have the same preference: $s_{i,j} = p_j$ for all $i \in [n]$, $j \in [\ell]$. Then, the only outcome that maximizes utilitarian or egalitarian social welfare is $(p_1, \ldots, p_m)$. However, any outcome with just a single project scheduled at the "correct" timestep would be a PROP outcome.

For proportionality, we obtain the following result. The proof can be found in the appendix.

Theorem 8.6.1. PROP *is at least as hard as* PCBP-RM. *The hardness result holds even in the FP setting.*

Proof. We prove the claim by reduction from EGAL with $\lambda = 1$. An instance of EGAL with $\lambda = 1$ consists of n agents, m projects, m timesteps, and an integer λ; it is a yes-instance if there exists an outcome $\mathbf{o}'$ such that every agent gets a utility of at least 1; and a no-instance otherwise.

Consider such an instance of EGAL with $\lambda = 1$ given by n agents $N = \{1, \ldots, n\}$, m projects $P = \{p_1, \ldots, p_m\}$ and m timesteps. Now, make copies of all agents and their approval sets. We make only one modification: if $n < m$, then duplicate the number of agents until its equal to m. Let the number of agents in this new instance be n' Thus, we have that $n' \geq m$. Note that an outcome is a solution to the EGAL problem with the original n agents if and only if the same outcome is a solution to the EGAL problem with n' agents (i.e., the only change to the problem duplicating agents does is changing the number of agents).

Next, we will prove that there exists an outcome $\mathbf{o}$ that gives each agent at least a utility of $\frac{m}{n'}$ if and only if there exists an outcome $\mathbf{o}'$ that gives each agent at least a utility of 1.

For the 'if' direction, let $\mathbf{o}'$ be an outcome that guarantees each agent at least a utility of 1. Then, since $n' \geq m$, $\frac{m}{n'} \leq 1$ and hence, any outcome that guarantees each agent a utility of 1 satisfies proportionality, i.e., let $\mathbf{o} = \mathbf{o}'$.

For the 'only if' direction, let $\mathbf{o}$ be an outcome that guarantees each agent at least a utility of $\frac{m}{n'}$. Then, since the utility of an agent is an integer, the outcome $\mathbf{o}$ guarantees each agent at least a utility of 1, i.e. let $\mathbf{o}' = \mathbf{o}$. □

8.7 Conclusion

We considered a variety of welfare and fairness objectives in the scheduling problem. Specifically, we provide a polynomial-time algorithm for maximizing utilitarian welfare, and hardness results for the decision problems for maximizing egalitarian welfare, achieving equitability or obtaining proportionality; for the problem of maximizing egalitarian welfare, we also identify special cases where optimal outcomes can be computed efficiently.

Avenues for future research include the following: (1) relaxing the capacity constraints on timesteps, so that we can implement multiple projects at each timestep; (2) considering agents with different entitlements and the associated fairness notions [Chakraborty et al., 2021a, Farhadi et al., 2019, Montanari et al., 2024, Suksompong and Teh, 2022]; or (3) designing strategyproof scheduling mechanisms.

Appendix A

Omitted Proofs from Chapter 4

A.1 Proof of Lemma 4.3.13

```
from itertools import combinations
from copy import deepcopy

# If there exists a partial allocation for first 2n+2 rounds such that bundle
    ↪ valuations are equal
if_some_envy_exists = False

def is_ef1(allocation, agents, valuations, if_some_envy_exists,
    ↪ partial_alloc_envy_from, partial_alloc_envy_to):
    """
    Check if the current allocation is EF1.

    Parameters:
    - allocation: List of lists, where allocation[i] is the list of goods allocated
        ↪  to agent i.
    - agents: List of agent identifiers.
    - valuations: Dictionary where valuations[agent][good] gives the value of a
        ↪ good for an agent.
    - if_some_envy_exists: If the partial allocation for the first 2n+2 rounds is
        ↪ EF (i.e., equal bundle values)
    - partial_alloc_envy_from: If if_some_envy_exists is True, then which agent
        ↪ envies
    - partial_alloc_envy_to: If if_some_envy_exists is True, then which agent is
        ↪ being envied

    Returns:
    - True if allocation is EF1, False otherwise.
    """
    num_agents = len(agents)

    # Compute the value each agent has for their own bundle
    agent_own_values = []
    for agent_idx in range(num_agents):
        agent = agents[agent_idx]
```

```
        total = sum(valuations[agent][good] for good in allocation[agent_idx])
        agent_own_values.append(total)

    # Check EF1 condition for every pair of agents (i, j)
    for i in range(num_agents):
        for j in range(num_agents):
            if i == j:
                continue
            agent_i = agents[i]
            agent_j_bundle = allocation[j]
            # Agent i's value for agent j's bundle
            lst = [valuations[agent_i][good] for good in agent_j_bundle]
            if lst:
                max_value = max(lst)
            else:
                max_value = 0
            value_i_for_j_less_one = sum(lst) - max_value
            # Agent i's own value
            value_i_own = agent_own_values[i]

            if if_some_envy_exists:
              if i == partial_alloc_envy_from:
                if j == partial_alloc_envy_to:
                  value_i_for_j_less_one += 1

            if value_i_own < value_i_for_j_less_one:
                return False
    return True

def find_ef1_allocations(agents, goods, valuations, if_some_envy_exists,
    ↪ partial_alloc_envy_from=0, partial_alloc_envy_to=0):
    """
    Find all allocations that are EF1 at each step of allocating goods one by one.

    Parameters:
    - agents: List of agent identifiers.
    - goods: List of goods to be allocated.
    - valuations: Dictionary where valuations[agent][good] gives the value of a
        ↪ good for an agent.
    - if_some_envy_exists: If the partial allocation for the first 2n+2 rounds is
        ↪ EF (i.e., equal bundle values)
    - partial_alloc_envy_from: If if_some_envy_exists is True, then which agent
        ↪ envies
    - partial_alloc_envy_to: If if_some_envy_exists is True, then which agent is
        ↪ being envied

    Returns:
    - List of allocations. Each allocation is a list of lists, where allocation[i]
        ↪ is the list of goods for agent i.
    """
    num_agents = len(agents)
    all_allocations = []

    def backtrack(current_allocation, index):
```

```
    """
    Recursive helper function to perform backtracking.

    Parameters:
    - current_allocation: Current allocation state.
    - index: Index of the next good to allocate.
    """
    if index == len(goods):
        # All goods allocated, add to results
        all_allocations.append(deepcopy(current_allocation))
        return

    current_good = goods[index]

    for agent_idx in range(num_agents):
        # Assign current_good to agent_idx
        current_allocation[agent_idx].append(current_good)

        # Check EF1 condition at this step
        if is_ef1(current_allocation, agents, valuations, if_some_envy_exists,
            ↪ partial_alloc_envy_from, partial_alloc_envy_to):
            # Continue to allocate the next good
            backtrack(current_allocation, index + 1)

        # Backtrack: remove the good from the agent's allocation
        current_allocation[agent_idx].pop()

# Initialize allocation: list of empty lists for each agent
initial_allocation = [[] for _ in agents]
backtrack(initial_allocation, 0)

return all_allocations

# Example Usage
if __name__ == "__main__":
    # Define agents and goods
    agents = ['A', 'B','C']
    goods = ['g1', 'g2', 'g3','g4', 'g5','g6', 'g7','g8', 'g9','g10', 'g11','g12',
        ↪ 'g13','g14', 'g15','g16', 'g17','g18', 'g19','g20','g21']

    # Define valuations for each agent
    valuations = {
        'A': {'g1': 90, 'g2': 80, 'g3': 70, 'g4' : 100, 'g5' : 100,'g6': 100, 'g7'
            ↪ :15,'g8':10000, 'g9':11000,'g10':12000, 'g11':20000,'g12':20000, '
            ↪ g13':20000,'g14':20000, 'g15':20000,'g16':20000, 'g17':20000,'g18'
            ↪ :20000, 'g19':20000,'g20':19010, 'g21' :18005},
      'B': {'g1': 90, 'g2': 70, 'g3': 80, 'g4' : 100, 'g5' : 100,'g6': 100, 'g7'
          ↪ :95,'g8':10000, 'g9':11000,'g10':12000, 'g11':20000,'g12':20000, 'g13'
          ↪ :20000,'g14':20000, 'g15':20000,'g16':20000, 'g17':20000,'g18':12000,
          ↪ 'g19':12000,'g20':19085, 'g21' :14106},
      'C': {'g1': 80, 'g2': 90, 'g3': 70, 'g4' : 100, 'g5' : 100,'g6': 100, 'g7'
          ↪ :25,'g8':10000, 'g9':11000,'g10':12000, 'g11':20000,'g12':20000, 'g13'
          ↪ :18500,'g14':20000, 'g15':20000,'g16':20000, 'g17':20000,'g18':20000,
          ↪ 'g19':20000,'g20':19010, 'g21' :19496}
```

```
}

# Find all EF1 allocations
if if_some_envy_exists:
    for partial_alloc_envy_from in range(3):
        for partial_alloc_envy_to in range(3):
            if partial_alloc_envy_from != partial_alloc_envy_to:
                ef1_allocations = find_ef1_allocations(agents, goods, valuations
                    ↪ ,True, partial_alloc_envy_from, partial_alloc_envy_to)
                # Each iteration considers different combinations of envy that
                    ↪ exists in the partial allocation for the first 2n+2
                    ↪ rounds

                # Print the allocations
                print(f"Total␣EF1␣allocations:␣{len(ef1_allocations)}\n")
else:
    ef1_allocations = find_ef1_allocations(agents, goods, valuations, False)
    # Print the allocations
    print(f"Total␣EF1␣allocations:␣{len(ef1_allocations)}\n")
for idx, alloc in enumerate(ef1_allocations, 1):
    print(f"Allocation␣{idx}:")
    for agent_idx, agent in enumerate(agents):
        print(f"␣␣{agent}:␣{alloc[agent_idx]}")
    print(f"␣{sum(valuations['A'][good]␣for␣good␣in␣alloc[0])␣-␣sum(valuations
        ↪ ['A'][good]␣for␣good␣in␣alloc[1])},{sum(valuations['A'][good]␣for␣
        ↪ good␣in␣alloc[0])␣-␣sum(valuations['A'][good]␣for␣good␣in␣alloc[2])}
        ↪ ")
    print(f"␣{sum(valuations['B'][good]␣for␣good␣in␣alloc[1])␣-␣sum(valuations
        ↪ ['B'][good]␣for␣good␣in␣alloc[0])},{sum(valuations['B'][good]␣for␣
        ↪ good␣in␣alloc[1])␣-␣sum(valuations['B'][good]␣for␣good␣in␣alloc[2])}
        ↪ ")
    print(f"␣{sum(valuations['C'][good]␣for␣good␣in␣alloc[2])␣-␣sum(valuations
        ↪ ['C'][good]␣for␣good␣in␣alloc[0])},{sum(valuations['C'][good]␣for␣
        ↪ good␣in␣alloc[2])␣-␣sum(valuations['C'][good]␣for␣good␣in␣alloc[1])}
        ↪ ")
    print()
```

A.2 TEF1 for Mixed Manna

We first define TEF1 for mixed manna.

Definition A.2.1 (Temporal EF1 for mixed manna)**.** In the case of with both goods and chores, an allocation $\mathcal{A}^t = (A_1^t, \ldots, A_n^t)$ is said to be *temporal envy-free up to one item (TEF1)* if for all $t' \leq t$ and $i, j \in N$, there exists an item $o \in A_i^{t'} \cup A_j^{t'}$ such that $v_i(A_i^{t'} \setminus \{o\}) \geq v_i(A_j^{t'} \setminus \{o\})$.

Then, we can extend the result of Theorem 4.3.2 to the more general mixed manna setting, with the following result.

Theorem A.2.2. *When $n = 2$, a* TEF1 *allocation exists in the mixed manna setting, and can be computed in polynomial time.*

Proof. For an agent $i \in \{1,2\}$ and round $t \in [T]$, we define $S_i^t \subseteq O^t$ as the set of items that have arrived up to round t which only agent i has a positive value for. Then, for any $t \in [t]$ and $i, j \in \{1,2\}$ where $i \neq j$, $v_i(S_i^t) \geq 0$ and $v_i(S_j^t) \leq 0$. Clearly, if some allocation $\mathcal{A}^t$ is TEF1 over $O^t \setminus (S_1^t \cup S_2^t)$, then $\mathcal{B}^t = (S_1^t \cup A_1^t, S_2^t \cup A_2^t)$ is a TEF1 allocation over O^t. Furthermore, for any $t \in [T]$ and $i, j \in \{1,2\}$ where $i \neq j$, if there exists an item $o \in A_i^t \cup A_j^t$ such that $v_i(A_i^t \setminus \{o\}) \geq v_i(A_j^t \setminus \{o\})$, then

$$\begin{aligned} v_i(B_i^t \setminus \{o\}) &= v_i(A_i^t \setminus \{o\}) + v_i(S_i^t) \\ &\geq v_i(A_i^t \setminus \{o\}) \\ &\geq v_i(A_j^t \setminus \{o\}) \\ &\geq v_i(A_j^t \setminus \{o\}) + v_i(S_j^t) \\ &= v_i(B_j^t \setminus \{o\}), \end{aligned}$$

where the first and third inequalities are due to the fact that $v_i(S_i^t) \geq 0$ and $v_i(S_j^t) \leq 0$. It therefore suffices to assume that for each item $o \in O$, either $v_1(o) \leq 0$ and $v_2(o) \leq 0$, or $v_1(o) \geq 0$ and $v_2(o) \geq 0$, and we make this assumption for the remainder of the proof.

Let $v_i'(o) = |v_i(o)|$ for all $i \in \{1,2\}$ and $o \in O$. Note that $v_i'(o) \geq 0$ for all $i \in \{1,2\}$ and $o \in O$ and thus, with respect to the augmented valuations, each $o \in O$ is a good. We use Algorithm 2 in He et al. [2019], which returns a TEF1 allocation for goods in polynomial time, to compute an allocation $\mathcal{B}$ which is TEF1 with respect to the augmented valuations $\mathbf{v}'$.

For a round $t \in [T]$, let $G^t, C^t \subseteq O^t$ be, respectively, the subsets of goods and chores (with respect to the original valuation profile $\mathbf{v} = (v_1, v_2)$) that have arrived up to round t. Then, for each $t \in [T]$ and $i \in \{1,2\}$, let $G_i^t = G^t \cap B_i^t$ and $C_i^t = C^t \cap B_i^t$. We construct allocation $\mathcal{A} = (G_1^T \cup C_2^T, G_2^T \cup C_1^T)$ from $\mathcal{B}$ by swapping the agents' bundles of chores. We now show that $\mathcal{A}$ is TEF1.

Recall that all items are goods with respect to $\mathbf{v}'$. Since $\mathcal{B}$ is TEF1, we know that for any $t \in [T]$ and $i, j \in \{1, 2\}$ where $i \neq j$, there exists an item $o \in B_j^t$ such that

$$
\begin{aligned}
& v_i'(B_i^t) \geq v_i'(B_j^t \setminus \{o\}) \\
& \implies v_i(G_i^t) - |v_i(C_i^t)| \geq v_i(G_j^t) - |v_i(C_j^t)| - |v_i(o)| \\
& \implies v_i(G_i^t) + v_i(C_j^t) \geq v_i(G_j^t) + v_i(C_i^t) - |v_i(o)| \\
& \implies \begin{cases} v_i(G_i^t) + v_i(C_j^t) \geq v_i(G_j^t \setminus \{o\}) + v_i(C_i^t) & \text{if } o \in G_j^t \\ v_i(G_i^t) + v_i(C_j^t \setminus \{o\}) \geq v_i(G_j^t) + v_i(C_i^t) & \text{if } o \in C_j^t \end{cases} \\
& \implies v_i(G_i^t \cup C_j^t \setminus \{o\}) \geq v_i(G_i^t \cup C_j^t \setminus \{o\}) \\
& \implies v_i(A_i^t \setminus \{o\}) \geq v_i(A_j^t \setminus \{o\}).
\end{aligned}
$$

Thus, $\mathcal{A}$ is TEF1. $\square$

Bibliography

Ron Aharoni, Eli Berger, Dani Kotlar, and Ran Ziv. On a conjecture of Stein. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 87: 203–211, 2017.

Hannaneh Akrami and Jugal Garg. Breaking the 3/4 barrier for approximate maximin share. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 74–91, 2024.

Hannaneh Akrami, Rojin Rezvan, and Masoud Seddighin. An EF2X allocation protocol for restricted additive valuations. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 17–23, 2022.

Susanne Albers and Matthias Hellwig. Semi-online scheduling revisited. *Theoretical Computer Science*, 443:1–9, 2012.

Martin Aleksandrov, Haris Aziz, Serge Gaspers, and Toby Walsh. Online fair division: Analysing a food bank problem. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2540–2546, 2015.

Amine Allouah, Christian Kroer, Xuan Zhang, Vashist Avadhanula, Nona Bohanon, Anil Dania, Caner Gocmen, Sergey Pupyrev, Parikshit Shah, Nicolas Stier-Moses, and Ken Rodríguez Taarup. Fair allocation over time, with applications to content moderation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, pages 25–35, 2023.

Noga Alon and Joel H Spencer. *The probabilistic method.* John Wiley & Sons, 2016.

Shiri Alouf-Heffetz, Laurent Bulteau, Edith Elkind, Nimrod Talmon, and Nicholas Teh. Better collective decisions via uncertainty reduction. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence (IJCAI)*, pages 24–30, 2022.

Georgios Amanatidis, Georgios Birmpas, and Vangelis Markakis. Comparing approximate relaxations of envy-freeness. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 42–48, 2018.

Georgios Amanatidis, Evangelos Markakis, and Apostolos Ntokos. Multiple birds with one stone: Beating 1/2 for EFX and GMMS via envy cycle elimination. *Theoretical Computer Science*, 841:94–109, 2020.

Georgios Amanatidis, Georgios Birmpas, Aris Filos-Ratsikas, Alexandros Hollender, and Alexandros A. Voudouris. Maximum Nash welfare and other stories about EFX. *Theoretical Computer Science*, 863:69–85, 2021.

Georgios Amanatidis, Haris Aziz, Georgios Birmpas, Aris Filos-Ratsikas, Bo Li, Hervé Moulin, Alexandros A. Voudouris, and Xiaowei Wu. Fair division of indivisible goods: Recent progress and open questions. *Artificial Intelligence*, 322(C), 2023.

Enrico Angelelli, Maria Grazia Speranza, and Zsolt Tuza. Semi on-line scheduling on three processors with known sum of the tasks. *Journal of Scheduling*, 10:263–269, 2007.

Spyros Angelopoulos, Shahin Kamali, and Kimia Shadkami. Online bin packing with predictions. *Journal of Artificial Intelligence Research*, 78:1111–1141, 2023.

R. Pablo Arribillaga and Agustín G. Bonifacio. Obvious manipulations of tops-only voting rules. *Games and Economic Behavior*, 143:12–24, 2024.

Kenneth J. Arrow. A difficulty in the concept of social welfare. *Journal of Political Economy*, 58(4):328–346, 1950.

Kenneth J. Arrow. *Social Choice and Individual Values.* Yale University Press, 2012.

Bengt Aspvall, Michael Plass, and Robert Tarjan. A linear-time algorithm for testing the truth of certain quantified boolean formulas. *Information Processing Letters*, 8:121–123, 1979.

Yossi Azar, Niv Buchbinder, and Kamal Jain. How to allocate goods in an online market? In *Proceedings of the 18th Annual European Symposium on Algorithms (ESA)*, pages 51–62, 2010.

Haris Aziz and Alexander Lam. Obvious manipulability of voting rules. In *Proceedings of the 7th International Conference on Algorithmic Decision Theory (ADT)*, pages 179–193, 2021.

Haris Aziz and Barton Lee. Proportionally representative participatory budgeting with ordinal preferences. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, pages 5110–5118, 2021.

Haris Aziz and Barton E Lee. The expanding approvals rule: Improving proportional representation and monotonicity. *Social Choice and Welfare*, 54(1):1–45, 2020.

Haris Aziz, Markus Brill, Vincent Conitzer, Edith Elkind, Rupert Freeman, and Toby Walsh. Justified representation in approval-based committee voting. *Social Choice and Welfare*, 48:461–485, 2017.

Haris Aziz, Edith Elkind, Shenwei Huang, Martin Lackner, Luis Sánchez-Fernández, and Piotr Skowron. On the complexity of extended and proportional justified representation. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*, pages 902–909, 2018.

Haris Aziz, Ioannis Caragiannis, Ayumi Igarashi, and Toby Walsh. Fair allocation of indivisible goods and chores. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 53–59, 2019.

Haris Aziz, Hervé Moulin, and Fedor Sandomirskiy. A polynomial-time algorithm for computing a pareto optimal and almost proportional allocation. *Operations Research Letters*, 48(5):573–578, 2020.

Haris Aziz, Xin Huang, Nicholas Mattei, and Erel Segal-Halevi. Computing welfare-maximizing fair allocations of indivisible goods. *European Journal of Operational Research*, 307(2):773–784, 2023a.

Haris Aziz, Jeremy Lindsay, Angus Ritossa, and Mashbat Suzuki. Fair allocation of two types of chores. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, pages 143–151, 2023b.

Moshe Babaioff and Uriel Feige. Fair shares: Feasibility, domination, and incentives. *Mathematics of Operations Research*, 49(4):2180–2211, 2024.

Moshe Babaioff, Tomer Ezra, and Uriel Feige. Fair and truthful mechanisms for dichotomous valuations. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, pages 5119–5126, 2021.

Santiago Balseiro, Christian Kroer, and Rachitesh Kumar. Single-leg revenue management with advice. In *Proceedings of the 24th ACM Conference on Economics and Computation (EC)*, page 207, 2023.

Siddhartha Banerjee, Vasilis Gkatzelis, Artur Gorokh, and Billy Jin. Online nash social welfare maximization with predictions. In *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1–19, 2022.

Siddhartha Banerjee, Vasilis Gkatzelis, Safwan Hossain, Billy Jin, Evi Micha, and Nisarg Shah. Proportionally fair online allocation of public goods with predictions. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 20–28, 2023.

Siddharth Barman and Ranjani G. Sundaram. Uniform welfare guarantees under identical subadditive valuations. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 46–52, 2020.

Siddharth Barman and Paritosh Verma. Truthful and fair mechanisms for matroid-rank valuations. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI)*, pages 4801–4808, 2022.

Siddharth Barman, Sanath Kumar Krishnamurthy, and Rohit Vaish. Finding fair and efficient allocations. In *Proceedings of the 19th ACM Conference on Economics and Computation (EC)*, pages 557–574, 2018a.

Siddharth Barman, Sanath Kumar Krishnamurthy, and Rohit Vaish. Greedy algorithms for maximizing Nash social welfare. In *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 7–13, 2018b.

Siddharth Barman, Umang Bhaskar, Anand Krishna, and Ranjani G. Sundaram. Tight approximation algorithms for p-mean welfare under subadditive valuations. In *Proceedings of the 28th Annual European Symposium on Algorithms (ESA)*, pages 11:1–11:17, 2020.

Siddharth Barman, Arindam Khan, and Arnab Maiti. Universal and tight online algorithms for generalized-mean welfare. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI)*, pages 4793–4800, 2022.

Sanjoy K Baruah. Fairness in periodic real-time scheduling. In *Proceedings 16th IEEE Real-Time Systems Symposium*, pages 200–209. IEEE, 1995.

Xiaohui Bei, Xinhang Lu, Pasin Manurangsi, and Warut Suksompong. The price of fairness for indivisible goods. *Theory of Computing Systems*, 65:1069–1093, 2021.

Xiaohui Bei, Zihao Li, and Junjie Luo. Fair and efficient multi-resource allocation for cloud computing. In *Proceedings of the 18th Conference on Web and Internet Economics (WINE)*, pages 169–186, 2022.

Nawal Benabbou, Mithun Chakraborty, Ayumi Igarashi, and Yair Zick. Finding fair and efficient allocations for matroid rank valuations. *ACM Transactions on Economics and Computation*, 9(4):21:1–21:41, 2021.

Gerdus Benadè, Aleksandr M. Kazachkov, Ariel D. Procaccia, and Christos-Alexandros Psomas. How to make envy vanish over time. In *Proceedings of the 19th ACM Conference on Economics and Computation (EC)*, pages 593–610, 2018.

Gerdus Benadè, Aleksandr M Kazachkov, Ariel D Procaccia, Alexandros Psomas, and David Zeng. Fair and efficient online allocations. *Operations Research*, 72(4): 1438–1452, 2024.

Dimitris Bertsimas, Vivek F. Farias, and Nikolaos Trichakis. The price of fairness. *Operations Research*, 59:17–31, 2011.

Bhavook Bhardwaj, Rajnish Kumar, and Josue Ortega. Fairness and efficiency in cake-cutting with single-peaked preferences. *Economics Letters*, 190:109064, 2020.

Duncan Black. On the rationale of group decision-making. *Journal of Political Economy*, 56(1):23–34, 1948.

Niclas Boehmer, Markus Brill, Alfonso Cevallos, Jonas Gehrlein, Luis Sánchez-Fernández, and Ulrike Schmidt-Kraepelin. Approval-based committee voting in practice: A case study of (over-)representation in the polkadot blockchain. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pages 9519–9527, 2024.

Niclas Boehmer, Sara Fish, and Ariel D. Procaccia. Generative social choice. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

Anna Bogomolnaia and Hervé Moulin. Random matching under dichotomous preferences. *Econometrica*, 72(1):257–279, 2004.

Sylvain Bouveret and Michel Lemaître. Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *Autonomous Agents and Multi-Agent Systems*, 30(2):259–290, 2016.

Joan Boyar, Lene M. Favrholdt, Christian Kudahl, Kim S. Larsen, and Jesper W. Mikkelsen. Online algorithms with advice: A survey. *ACM Computing Surveys (CSUR)*, 50(2), 2017.

Steven Brams and Peter Fishburn. *Approval voting*. Springer, 1983.

Steven J. Brams and Peter C. Fishburn. *Approval Voting*. Springer, 2007.

Steven J. Brams and Alan D. Taylor. *Fair Division: From Cake-Cutting to Dispute Resolution*. Cambridge University Press, 1996.

Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel Procaccia. *Handbook of Computational Social Choice*. Cambridge University Press, 2016.

Robert Bredereck, Piotr Faliszewski, Ayumi Igarashi, Martin Lackner, and Piotr Skowron. Multiwinner elections with diversity constraints. In *Proceedings of the 32th AAAI Conference on Artificial Intelligence (AAAI)*, pages 933–940, 2018.

Robert Bredereck, Piotr Faliszewski, Andrzej Kaczmarczyk, and Rolf Niedermeier. An experimental view on committees providing justified representation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 109–115, 2019.

Markus Brill and Jannik Peters. Robust and verifiable proportionality axioms for multiwinner voting. In *Proceedings of the 24th ACM Conference on Economics and Computation (EC)*, page 301, 2023.

Markus Brill and Jannik Peters. Completing priceable committees: Utilitarian and representation guarantees for proportional multiwinner voting. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pages 9528–9536, 2024.

Laurent Bulteau, Noam Hazon, Rutvik Page, Ariel Rosenfeld, and Nimrod Talmon. Justified representation for perpetual voting. *IEEE Access*, 9:96598–96612, 2021.

Jeff Burdges, Alfonso Cevallos, Peter Czaban, Rob Habermeier, Syed Hosseini, Fabio Lama, Handan Kilinç Alper, Ximin Luo, Fatemeh Shirazi, Alistair Stewart, and Gavin Wood. Overview of polkadot and its design considerations. *arXiv preprint arXiv:2005.13456*, 2020.

Franklin Camacho, Rigoberto Fonseca-Delgado, Ramón Pino Pérez, and Guido Tapia. Generalized binary utility functions and fair allocations. *Mathematical Social Sciences*, 121:50–60, 2023.

Ioannis Caragiannis and Shivika Narang. Repeatedly matching items to agents fairly and efficiently. *Theoretical Computer Science*, 981:114246, 2024.

Ioannis Caragiannis, David Kurokawa, Hervé Moulin, Ariel D. Procaccia, Nisarg Shah, and Junxing Wang. The unreasonable fairness of maximum Nash welfare. *ACM Transactions on Economics and Computation*, 7(3):12:1–12:32, 2019.

Ioannis Caragiannis, Jugal Garg, Nidhi Rathi, Eklavya Sharma, and Giovanna Varricchio. New fairness concepts for allocating indivisible items. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2554–2562, 2025.

L. Elisa Celis, Lingxiao Huang, and Nisheeth K. Vishnoi. Multiwinner voting with fairness constraints. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 144–151, 2018.

Alfonso Cevallos and Alistair Stewart. A verifiably secure and proportional committee election rule. In *Proceedings of the 3rd ACM Conference on Advances in Financial Technologies (AFT)*, pages 29–42, 2021.

Mithun Chakraborty, Ayumi Igarashi, Warut Suksompong, and Yair Zick. Weighted envy-freeness in indivisible item allocation. *ACM Transactions on Economics and Computation*, 9(3):18:1–18:39, 2021a.

Mithun Chakraborty, Ulrike Schmidt-Kraepelin, and Warut Suksompong. Picking sequences and monotonicity in weighted fair division. *Artificial Intelligence*, 301: 103578, 2021b.

Nikhil Chandak, Shashwat Goel, and Dominik Peters. Proportional aggregation of preferences for sequential decision making. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pages 9573–9581, 2024.

Bhaskar Ray Chaudhury, Jugal Garg, and Ruta Mehta. Fair and efficient allocations under subadditive valuations. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, pages 5269–5276, 2021.

Bhaskar Ray Chaudhury, Jugal Garg, and Kurt Mehlhorn. EFX exists for three agents. *Journal of the ACM*, 71(1), 2024.

Jiehua Chen, Christian Hatschka, and Sofia Simola. Multi-winner reconfiguration. In *Proceedings of the 37th International Conference on Neural Information Processing Systems (NeurIPS)*, pages 85899–85931, 2024.

T.C. Edwin Cheng, Hans Kellerer, and Vladimir Kotov. Semi-on-line multiprocessor scheduling with given total processing time. *Theoretical Computer Science*, 337(1): 134–146, 2005.

Davin Choo, Yan Hao Ling, Warut Suksompong, Nicholas Teh, and Jian Zhang. Envy-free house allocation with minimum subsidy. *Operations Research Letters*, page 107103, 2024.

Vincent Conitzer, Rupert Freeman, and Nisarg Shah. Fair public decision making. In *Proceedings of the 18th ACM Conference on Economics and Computation (EC)*, pages 629–646, 2017.

Marek Cygan, Fedor V Fomin, Łukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized algorithms*, volume 5. Springer, 2015.

Andreas Darmann and Joachim Schauer. Maximizing Nash product social welfare in allocating indivisible goods. *European Journal of Operational Research*, 247(2): 548–559, 2015.

Eva Michelle Deltl, Till Fluschnik, and Robert Bredereck. Algorithmics of egalitarian versus equitable sequences of committees. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2651–2658, 2023.

Debasis Dwibedy and Rakesh Mohanty. Semi-online scheduling: A survey. *Computers & Operations Research*, 139:105646, 2022.

Soroush Ebadian, Dominik Peters, and Nisarg Shah. How to fairly allocate easy and difficult chores. In *21st International Conference on Autonomous Agents and Multiagent Systems*, 2022.

Tomáš Ebenlendr, John Noga, Jiří Sgall, and Gerhard Woeginger. A note on semi-online machine covering. In *Approximation and Online Algorithms*, pages 110–118. Springer Berlin Heidelberg, 2006.

Owen Eckart, Alexandros Psomas, and Paritosh Verma. On the fairness of normalized p-means for allocating goods and chores. *arXiv preprint arXiv:2402.14996*, 2024.

Edith Elkind, Piotr Faliszewski, Piotr Skowron, and Arkadii Slinko. Properties of multiwinner voting rules. *Social Choice and Welfare*, 48:599–632, 2017.

Edith Elkind, Sonja Kraiczy, and Nicholas Teh. Fairness in temporal slot assignment. In *Proceedings of the 15th International Symposium on Algorithmic Game Theory (SAGT)*, pages 490–507, 2022.

Edith Elkind, Warut Suksompong, and Nicholas Teh. Settling the score: Portioning with cardinal preferences. In *Proceedings of the 26th European Conference on Artificial Intelligence (ECAI)*, pages 621–628, 2023.

Edith Elkind, Piotr Faliszewski, Ayumi Igarashi, Pasin Manurangsi, Ulrike Schmidt-Kraepelin, and Warut Suksompong. The price of justified representation. *ACM Transactions on Economics and Computation*, 12(3):11:1–11:27, 2024a.

Edith Elkind, Ayumi Igarashi, and Nicholas Teh. Fair division of chores with budget constraints. In *Proceedings of the 17th International Symposium on Algorithmic Game Theory (SAGT)*, pages 55–71, 2024b.

Edith Elkind, Tzeh Yuan Neoh, and Nicholas Teh. Temporal elections: Welfare, strategyproofness, and proportionality. In *Proceedings of the 27th European Conference on Artificial Intelligence (ECAI)*, pages 3292–3299, 2024c.

Edith Elkind, Svetlana Obraztsova, and Nicholas Teh. Temporal fairness in multiwinner voting. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pages 22633–22640, 2024d.

Edith Elkind, Alexander Lam, Mohamad Latifian, Tzeh Yuan Neoh, and Nicholas Teh. Temporal fair division of indivisible items. In *Proceedings of the 24th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2025a.

Edith Elkind, Tzeh Yuan Neoh, and Nicholas Teh. Not in my backyard! temporal voting over public chores. In *Proceedings of the 34th International Joint Conference on Artificial Intelligence (IJCAI)*, 2025b.

Edith Elkind, Svetlana Obraztsova, Jannik Peters, and Nicholas Teh. Verifying proportionality in temporal voting. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence (AAAI)*, pages 13805–13813, 2025c.

Shimon Even, Alon Itai, and Adi Shamir. On the complexity of time table and multi-commodity flow problems. In *Proceedings of the 16th Symposium on Foundations of Computer Science (FOCS)*, pages 184–193, 1975.

Brandon Fain, Kamesh Munagala, and Nisarg Shah. Fair allocation of indivisible public goods. In *Proceedings of the 19th ACM Conference on Economics and Computation (EC)*, pages 575–592, 2018.

Piotr Faliszewski, Piotr Skowron, Arkadii Slinko, and Nimrod Talmon. Multiwinner voting: A new challenge for social choice theory. In *Trends in Computational Social Choice*, chapter 2, pages 27–47. 2017.

Alireza Farhadi, Mohammad Ghodsi, MohammadTaghi Hajiaghayi, Sebastien Lahaie, David Pennock, Masoud Seddighin, Saeed Seddighin, and Hadi Yami. Fair allocation of indivisible goods to asymmetric agents. *Journal of Artificial Intelligence Research*, 64:1–20, 2019.

Sara Fish, Paul Gölz, David C. Parkes, Ariel D. Procaccia, Gili Rusak, Itai Shapira, and Manuel Wüthrich. Generative social choice. In *Proceedings of the 25th ACM Conference on Economics and Computation (EC)*, page 985, 2024.

Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2006.

Moti Frances and Ami Litman. On covering problems of codes. *Theory of Computing Systems*, 30:113–119, 2007.

Rupert Freeman, Seyed Majid Zahedi, and Vincent Conitzer. Fair and efficient social choice in dynamic settings. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 4580–4587, 2017.

Rupert Freeman, Sujoy Sikdar, Rohit Vaish, and Lirong Xia. Equitable allocations of indivisible goods. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 280–286, 2019.

Michael Randolph Garey, David Stifler Johnson, and Larry Joseph Stockmeyer. Some simplified np-complete graph problems. *Theoretical Computer Science*, 1(3):237–267, 1976.

Jugal Garg, Aniket Murhekar, and John Qin. Fair and efficient allocations of chores under bivalued preferences. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI)*, pages 5043–5050, 2022.

Jugal Garg, Aniket Murhekar, and John Qin. Weighted ef1 and po allocations with few types of agents or chores. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2799–2806, 2024.

Grzegorz Gawron and Piotr Faliszewski. Using multiwinner voting to search for movies. *Theory and Decision*, 2024.

Abheek Ghosh, Tzeh Yuan Neoh, Nicholas Teh, and Giannis Tyrovolas. Fraud-proof revenue division on subscription platforms. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, 2025.

Vasilis Gkatzelis, Alexandros Psomas, and Xizhi Tan. Fair and efficient online allocations with normalized valuations. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, pages 5440–5447, 2021.

Hiromichi Goko, Ayumi Igarashi, Yasushi Kawase, Kazuhisa Makino, Hanna Sumita, Akihisa Tamura, Yu Yokoi, and Makoto Yokoo. A fair and truthful mechanism with limited subsidy. *Games and Economic Behavior*, 144:49–70, 2024.

Laurent Gourvès, Julien Lesca, and Anaëlle Wilczynski. On fairness via picking sequences in allocation of indivisible goods. In *Proceedings of the 7th International Conference on Algorithmic Decision Theory (ADT)*, pages 258–272, 2021.

Ronald Graham. Bounds on multiprocessing timing anomalies. *SIAM Journal on Applied Mathematics*, 17(2):416–429, 1969.

Daniel Halpern, Ariel D. Procaccia, Alexandros Psomas, and Nisarg Shah. Fair division with binary valuations: One rule to rule them all. In *Proceedings of the*

16th Conference on Web and Internet Economics (WINE), pages 370–383, 2020. Extended version available at arXiv:2007.06073.

Jiafan He, Ariel Procaccia, Alexandros Psomas, and David Zeng. Achieving a fairer future by changing the past. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 343–349, 2019.

Yong He and Guochuan Zhang. Semi on-line scheduling on two identical machines. *Computing*, 62(3):179–187, 1999.

Hadi Hosseini, Sujoy Sikdar, Rohit Vaish, Hejun Wang, and Lirong Xia. Fair division through information withholding. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, pages 2014–2021, 2020.

Ayumi Igarashi, Martin Lackner, Oliviero Nardi, and Arianna Novaro. Repeated fair allocation of indivisible items. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pages 9781–9789, 2024.

Sungjin Im and Benjamin Moseley. Fair scheduling via iterative quasi-uniform sampling. *SIAM Journal on Computing*, 49(3):658–680, 2020.

Sungjin Im, Ravi Kumar, Mahshid Montazer Qaem, and Manish Purohit. Online knapsack with frequency predictions. In *Proceedings of the 35th International Conference on Neural Information Processing Systems (NeurIPS)*, pages 2733–2743, 2021.

Ido Kahana and Noam Hazon. The leximin approach for a sequence of collective decisions. *arXiv preprint arXiv:2305.18024*, 2023.

Hans Kellerer, Vladimir Kotov, Maria Grazia Speranza, and Zsolt Tuza. Semi on-line algorithms for the partition problem. *Operations Research Letters*, 21(5):235–242, 1997.

Hans Kellerer, Vladimir Kotov, and Michaël Gabay. An efficient algorithm for semi-online multiprocessor scheduling with given total processing time. *Journal of Scheduling*, 18:623–630, 2015.

D. Marc Kilgour. Approval balloting for multi-winner elections. In Jean-François Laslier and M. Remzi Sanver, editors, *Handbook on Approval Voting*, chapter 6, pages 105–124. Springer, 2010.

Bettina Klaus and Eiichi Miyagawa. Strategy-proofness, solidarity, and consistency for multiple assignment problems. *International Journal of Game Theory*, 30(3): 421–435, 2002.

Melven Krom. The decision problem for a class of first-order formulas in which all disjunctions are binary. *Mathematical Logic Quarterly*, 13(1–2):15–20, 1967.

Maria Kyropoulou, Warut Suksompong, and Alexandros A. Voudouris. Almost envy-freeness in group resource allocation. *Theoretical Computer Science*, 841:110–123, 2020.

Martin Lackner. Perpetual voting: Fairness in long-term decision making. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, pages 2103–2110, 2020.

Martin Lackner and Piotr Skowron. Utilitarian welfare and representation guarantees of approval-based multiwinner rules. *Artificial Intelligence*, 288:103366, 2020.

Martin Lackner and Piotr Skowron. *Multi-Winner Voting with Approval Preferences*. Springer, 2023.

Martin Lackner, Jan Maly, and Simon Rey. Fairness in long-term participatory budgeting. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 299–305, 2021.

Kevin Lanctot, Ming Li, Bin Ma, Shaojiu Wang, and Louxin Zhang. Distinguishing string selection problems. *Information and Computation*, 185(1):41–55, 2003.

Euiwoong Lee. APX-hardness of maximizing Nash social welfare with indivisible items. *Information Processing Letters*, 122:17–20, 2017.

Kangbok Lee and Kyungkuk Lim. Semi-online scheduling problems on a small number of machines. *Journal of Scheduling*, 16:461–477, 2013.

Hendrik W Lenstra Jr. Integer programming with a fixed number of variables. *Mathematics of Operations Research*, 8(4):538–548, 1983.

Bo Li, Minming Li, and Ruilong Zhang. Fair scheduling for time-dependent resources. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:21744–21756, 2021.

Richard J. Lipton, Evangelos Markakis, Elchanan Mossel, and Amin Saberi. On approximately fair allocations of indivisible goods. In *Proceedings of the 5th ACM Conference on Electronic Commerce (EC)*, pages 125–131, 2004.

Komal Malik and Debasis Mishra. Pareto efficient combinatorial auctions: Dichotomous preferences without quasilinearity. *Journal of Economic Theory*, 191:105128, 2021.

Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. Adwords and generalized online matching. *Journal of the ACM*, 54(5):579–603, 2007.

Luisa Montanari, Ulrike Schmidt-Kraepelin, Warut Suksompong, and Nicholas Teh. Weighted envy-freeness for submodular valuations. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pages 9865–9873, 2024.

Hervé Moulin. *Fair Division and Collective Welfare*. MIT Press, 2003.

Hervé Moulin. Fair division in the internet age. *Annual Review of Economics*, 11: 407–441, 2019.

Riley Murray, Christian Kroer, Alex Peysakhovich, and Parikshit Shah. Robust market equilibria with uncertain preferences. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI)*, pages 2192–2199, 2020.

Dolev Mutzari, Yonatan Aumann, and Sarit Kraus. Resilient fair allocation of indivisible goods. In *Proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 2688–2690, 2023.

Tzeh Yuan Neoh and Nicholas Teh. Welfare maximization in perpetual voting (student abstract). In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pages 23597–23599, 2024.

Tzeh Yuan Neoh and Nicholas Teh. Understanding efx allocations: Counting and variants. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence (AAAI)*, pages 14036–14044, 2025.

Rolf Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006.

Christos H. Papadimitriou. *Computational complexity*. Academic Internet Publ., 2007. ISBN 978-1-4288-1409-7.

René Peeters. The maximum edge biclique problem is NP-complete. *Discrete Applied Mathematics*, 131(3):651–654, 2003.

Guillem Perarnau and Oriol Serra. Rainbow matchings: Existence and counting. *arXiv preprint arXiv:1104.2702*, 2011.

Dominik Peters. Proportional representation for artificial intelligence. In *Proceedings of the 27th European Conference on Artificial Intelligence (ECAI)*, pages 27–31, 2024.

Dominik Peters and Piotr Skowron. Proportionality and the limits of welfarism. In *Proceedings of the 21st ACM Conference on Economics and Computation (EC)*, pages 793–794, 2020.

Dominik Peters, Grzegorz Pierczyński, and Piotr Skowron. Proportional participatory budgeting with additive utilities. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pages 12726–12737, 2021.

Edvard Phragmén. Proportionella val. en valteknisk studie. In *Svenska spörsmål 25*. Lars Hökersbergs förlag, Stockholm, 1895.

Benjamin Plaut and Tim Roughgarden. Almost envy-freeness with general valuations. *SIAM Journal on Discrete Mathematics*, 34(2):1039–1068, 2020.

Ariel Procaccia. Technical perspective: An answer to fair division's most enigmatic question. *Communications of the ACM*, 63(4):118, 2020.

Manon Revel, Smitha Milli, Tyler Lu, Jamelle Watson-Daniels, and Max Nickel. Representative ranking for deliberation in the public sphere. In *Proceedings of the 42th International Conference on Machine Learning (ICML)*, 2025.

Jack Robertson and William Webb. *Cake-Cutting Algorithms: Be Fair if You Can.* Peters/CRC Press, 1998.

Luis Sánchez-Fernández, Edith Elkind, Martin Lackner, Norberto Fernández, Jesús Fisteus, Pablo Basanta Val, and Piotr Skowron. Proportional justified representation. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI)*, pages 670–676, 2017.

Thomas J. Schaefer. The complexity of satisfiability problems. In *Proceedings of the 10th Annual ACM Symposium on Theory of Computing (STOC)*, pages 216–226, 1978.

Alexander Schrijver. *Combinatorial Optimization*. Springer, 2003.

Uwe Schwiegelshohn and Ramin Yahyapour. Fairness in parallel job scheduling. *Journal of Scheduling*, 3(5):297–320, 2000.

Erel Segal-Halevi and Warut Suksompong. Democratic fair allocation of indivisible goods. *Artificial Intelligence*, 277:103167, 2019.

Steve Seiden, Jiri Sgall, and Gerhard Woeginger. Semi-online scheduling with decreasing job sizes. *Operations Research Letters*, 27(5):215–221, 2000.

Hugo Steinhaus. The problem of fair division. *Econometrica*, 16(1):101–104, 1948.

Errikos Streviniotis and Georgios Chalkiadakis. Preference aggregation mechanisms for a tourism-oriented Bayesian recommender. In *Proceedings of the 23rd International Conference on Principles and Practice of Multi-Agent Systems (PRIMA)*, pages 331–346, 2022.

Warut Suksompong. A characterization of maximum Nash welfare for indivisible goods. *Economics Letters*, 222:110956, 2023.

Warut Suksompong and Nicholas Teh. On maximum weighted Nash welfare for binary valuations. *Mathematical Social Sciences*, 117:101–108, 2022.

Warut Suksompong and Nicholas Teh. Weighted fair division with matroid-rank valuations: Monotonicity and strategyproofness. *Mathematical Social Sciences*, 126:48–59, 2023.

Zhiyi Tan and Yong Wu. Optimal semi-online algorithms for machine covering. *Theoretical Computer Science*, 372(1):69–90, 2007.

Thorvald Thiele. Om flerfoldsvalg. *Oversigt Over Det Kongelige Danske Videnskabernes Selskabs Forhandlinger*, pages 415–441, 1895.

William Thomson. Resource-monotonic solutions to the problem of fair division when preferences are single-peaked. *Social Choice and Welfare*, 11(3):205–223, 1994.

Peter Troyan and Thayer Morrill. Obvious manipulations. *Journal of Economic Theory*, 185:104970, 2020.

Vignesh Viswanathan and Yair Zick. Yankee Swap: a fast and simple fair allocation mechanism for matroid rank valuations. In *Proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 179–187, 2023.

Jizhou Wu, Jianye Hao, Tianpei Yang, Xiaotian Hao, Yan Zheng, Weixun Wang, and Matthew E. Taylor. Portal: Automatic curricula generation for multiagent reinforcement learning. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pages 15934–15942, 2024.

Yong Wu, Zhiyi Tan, and Qifan Yang. Optimal semi-online scheduling algorithms on a small number of machines. In *Combinatorics, Algorithms, Probabilistic and Experimental Methodologies*, pages 504–515. Springer Berlin Heidelberg, 2007.

Yaodong Yang, Jun Luo, Ying Wen, Oliver Slumbers, Daniel Graves, Haitham Bou Ammar, Jun Wang, and Matthew E. Taylor. Diverse auto-curriculum is critical for successful real-world multiagent learning systems. In *Proceedings of the 20th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 51–56, 2021.

Sheung Man Yuen and Warut Suksompong. Extending the characterization of maximum Nash welfare. *Economics Letters*, 224:111030, 2023.

Valentin Zech, Niclas Boehmer, Edith Elkind, and Nicholas Teh. Multiwinner temporal voting with aversion to change. In *Proceedings of the 27th European Conference on Artificial Intelligence (ECAI)*, pages 3236–3243, 2024.

Shengwei Zhou, Rufan Bai, and Xiaowei Wu. Multi-agent online scheduling: MMS allocations for indivisible items. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pages 42506–42516, 2023.