NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

FAIRNESS AND EFFICIENCY IN RESOURCE ALLOCATION

LI ZIHAO

SCHOOL OF PHYSICAL AND MATHEMATICAL SCIENCES

2025

FAIRNESS AND EFFICIENCY IN RESOURCE ALLOCATION

LI ZIHAO

SCHOOL OF PHYSICAL AND MATHEMATICAL SCIENCES

A thesis submitted to the Nanyang Technological University in partial fulfilment of the requirement for the degree of Doctor of Philosophy

2025

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research done by me except where otherwise stated in this thesis. The thesis work has not been submitted for a degree or professional qualification to any other university or institution. I declare that this thesis is written by myself and is free of plagiarism and of sufficient grammatical clarity to be examined. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

09/01/2025 ....

Date

Li Zihao ....

LI ZIHAO

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it of sufficient grammatical clarity to be examined. To the best of my knowledge, the thesis is free of plagiarism and the research and writing are those of the candidate's except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

10/01/2025
. . . .
Date

. . . .
Prof. Zhenzhen YAN

Authorship Attribution Statement

(B) This thesis contains material from 5 paper(s) published in the following peer-reviewed journal(s) / from papers accepted at conferences in which I am listed as an author.

Chapter 3 is published as Zihao Li, Hao Wang, and Zhenzhen Yan. Fully online matching with stochastic arrivals and departures. In Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence (AAAI'23), Vol. 37. AAAI Press, Article 1348, 12014–12021. DOI: 10.1609/aaai.v37i10.26417.

The contributions of the co-authors are as follows:

- Since this is a publication in theoretical computer science, authors are listed in alphabetical order and are considered to have contributed equally by convention. All the results of this work were obtained through collaborative discussion.
- For the preparation of this paper, all of us were involved in the writing. Specifically, I wrote the theoretical part, Hao Wang wrote the experimental part, and Zhenzhen Yan revised the submission.

Chapter 4 is published as Bo Li, Zihao Li, Shengxin Liu, and Zekai Wu. Allocating Mixed Goods with Customized Fairness and Indivisibility Ratio. In the 33rd International Joint Conference on Artificial Intelligence (IJCAI-24). DOI: 10.48550/ARXIV.2404.18132.

The contributions of the co-authors are as follows:

- Since this is a publication in theoretical computer science, authors are listed in alphabetical order and are considered to have contributed equally by convention. All the results of this work were obtained through collaborative discussion.
- For the preparation of this paper, all of us were involved in the writing. Each of us wrote parts of the paper, and we all participated in the revision process.

Chapter 5 is published as Zihao Li, Xiaohui Bei, and Zhenzhen Yan. Proportional Allocation of Indivisible Resources under Ordinal and Uncertain Preferences. In Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence, PMLR 180:1148-1157, 2022.

The contributions of the co-authors are as follows:

- For the contributions of the ideas in this paper, all results were obtained through collaborative discussion.

- For the preparation of this paper, I drafted the initial version, and Xiaohui Bei and Zhenzhen Yan revised the submission.

Chapter 6 is published as Zihao Li, Shengxin Liu, Xinhang Lu, Biaoshuai Tao and Yichen Tao. A Complete Landscape for the Price of Envy-Freeness. In Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS), 2024. DOI: 10.5555/3635637.3662975.

The contributions of the co-authors are as follows:

- Since this is a publication in theoretical computer science, authors are listed in alphabetical order and considered to have contributed equally by convention. All the results of this work were obtained through collaborative discussion.
- For the preparation of this paper, all of us were involved in the writing. Each of us wrote different parts of the paper, and all of us participated in the revision process.

Chapter 7 is published as Xiaohui Bei, Zihao Li and Junjie Luo. Fair and Efficient Multi-Resource Allocation for Cloud Computing. In The 18th Conference on Web and Internet Economics (WINE 2022). Cham: Springer International Publishing, 2022: 169-186. DOI: 10.1007/978-3-031-22832-2_10.

The contributions of the co-authors are as follows:

- Since this is a publication in theoretical computer science, authors are listed in alphabetical order and are considered to have contributed equally by convention. All the results of this work were obtained through discussion.
- For the preparation of this paper, all of us were involved in the writing. Each of us wrote different sections of the paper, and all of us participated in the revision process.

09/01/2025

Date

LI ZIHAO

Acknowledgements

I would like to begin by expressing my deepest gratitude to my advisor, Professor Zhenzhen Yan. Although she did not become my advisor until the second year of my Ph.D., her guidance and support have been invaluable to my academic journey. Her insights and encouragement have significantly shaped my research and academic growth.

I also sincerely thank Professor Xiaohui Bei, whose mentorship marked the beginning of my formal academic career. His invaluable assistance and unwavering support have been instrumental in my development as a researcher.

I also extend my heartfelt thanks to my Thesis Advisory Committee (TAC) members, Professor Bo An and Professor Yi Li. Your guidance and support throughout my PhD journey have been immensely helpful and I am deeply appreciative of your contributions to my academic success.

My heartfelt thanks go to all my collaborators: Xiaohui Bei, Xiaolin Bu, Bo Li, Yanwen Li, Jinyan Liu, Limeng Liu, Shengxin Liu, Xinhang Lu, Junjie Luo, Jiaxin Song, Biaoshuai Tao, Yichen Tao, Hao Wang, Zekai Wu and Zhenzhen Yan. Working with each of you has been a privilege and a pleasure, and I cherish the knowledge and experiences we have shared.

I also want to extend my sincere thanks to everyone in our research group: Shouchang Chen, Guangyan Gan, You Hui Goh, Jingkai Huang, Yanwen Li, Limeng Liu, Xinhang Lu, Junjie Luo, Yimeng Sun, Hao Wang, Hongao Wang, Ping Wang, Xuchen Wang, Chunxue Yang, and Ling Zhang. Your camaraderie and collaboration have enriched my research experience, and I look forward to more fruitful partnerships in the future.

A special thanks goes out to everyone in the office MAS-05-34. The past four years spent together have been memorable, and I am grateful for the shared experiences and support.

I also would like to express my thanks to all the secretaries in SPMS. Your help with administrative challenges has been invaluable and I appreciate your help.

I then want to extend my gratitude to Nanyang Technological University. I have truly enjoyed my PhD journey here, and I am thankful for the enriching environment and opportunities provided.

I am deeply grateful for my friends, whose advice and companionship have guided me through times of confusion and uncertainty. Your support means the world to me.

The greatest thanks go to my family for their unwavering support and encouragement throughout my academic journey. Your love and belief in me have been my greatest source of strength.

I would also like to express my gratitude to my thesis examiners. Thanks for your insightful comments on my thesis.

Finally, I express my gratitude to Wang Chen for providing a very useful thesis template. Your help has been greatly appreciated and has made a significant difference in the completion of my work.

Table of Contents

Abstract

Fairness and efficiency are two fundamental topics in resource allocation. In resource allocation problems, we need to allocate a set of items to a set of agents. From an efficiency perspective, our goal is to maximize the total reward of the allocation. From the perspective of fairness, we want to ensure that every agent feels the allocation is fair. This thesis aims to identify required allocations from these different perspectives. We present our findings in three parts: (i) efficiency in online resource allocation, (ii) fairness in offline resource allocation, and (iii) fairness and efficiency in offline resource allocation.

In Part I, we study the efficiency guarantees in the fully online matching problem, which is a constrained resource allocation problem. Since efficiency is easily guaranteed in the offline setting, we focus on the fully online setting where both agents and items arrive dynamically. For this problem, we provide a randomized algorithm that achieves a competitive ratio of 0.1549.

In Part II, we investigate fairness guarantees in the offline resource allocation problem. In this part, we first consider a mixed goods setting where some items can be divided into arbitrarily small pieces, while others can only be fully allocated to an agent. In this problem, we propose two "up to a fraction" relaxations of some classical fairness concepts, and prove the existence of such fair allocations in the mixed goods setting. We then return to the indivisible goods setting, where every item must be fully allocated to a single agent. In this setting, we allow some uncertainties in agents' preferences towards items and provide some complexity analysis of finding allocations that can be fair with a high probability.

In Part III, we aim to find a fair and efficient allocation in the offline resource allocation problem. We first study the price of fairness, which measures the efficiency loss when considering the fairness constraints. For this, we consider both the indivisible goods and the mixed goods settings, and provide a tight analysis of the price of fairness in these two settings. We then come to a resource allocation problem for cloud computing, where agents have Leontief preferences over

items. In this setting, we propose several new mechanisms that can produce a fair allocation with better social welfare guarantee.

Chapter 1

Introduction

Resource allocation is a key concept in both economics and computer science, focusing on distributing a set of items among a group of agents to achieve optimal outcomes. The primary objective is to maximize social welfare, making efficiency the central goal. This concept has numerous practical applications. For example, in a ride-hailing platform, vehicle requests appear randomly over time. The platform's job is to match these requests with available drivers. The reward earned from each successful match can vary depending on the matched pairs of vehicles and requests, influenced by factors like location and personal preferences. The platform strives to find the optimal matches that maximize overall rewards.

Efficiency is a key objective in resource allocation, particularly because resources often have limited capacity. It is crucial to utilize these resources to their fullest potential to maximize the overall rewards. As highlighted by Hensley et al. [81], enhancements in the design of the ride-sharing market could influence around 20 percent of vehicle miles traveled in the United States in 2016. Consequently, optimal resource allocation is vital for improving efficiency and maximizing the benefits of these systems.

Besides efficiency, fairness is also a crucial concern for researchers. Fair resource allocation fosters trust and credibility among agents, leading to higher satisfaction levels. Furthermore, fair distribution encourages participation from a diverse range of agents, contributing to the platform's growth and sustainability. In certain contexts, fairness may even take precedence over efficiency. For instance, when the government distributes public resources among various groups, the main goal is to ensure that each group feels satisfied with their share, emphasizing the perception of fairness. Similarly, in a corporate setting, when employers assign tasks or benefits to employees, they must consider each employee's abilities and preferences to ensure a fair distribution. This approach can enhance employee satisfaction and boost their performance. As demonstrated by Kropp et al. [96], a perception of fairness among employees can lead to up to a 26% improvement in performance and a 27% increase in retention. These examples underscore the significance of fairness in resource allocation.

This thesis aims to explore the concepts of efficiency and fairness in resource allocation, focusing on both online and offline settings. In an offline setting, where complete information about agents and items is available, a simple matching algorithm using deterministic optimization is typically sufficient to achieve high efficiency. In contrast, an online setting involves requests arriving randomly and sequentially, requiring real-time resource allocation decisions upon their arrival. The unpredictability of future requests significantly impacts resource utilization and the rewards generated. Therefore, this thesis will focus on developing efficient algorithms for online resource allocation to address these challenges when considering the efficiency issue.

Fairness is a crucial yet challenging aspect of resource allocation in both online and offline settings. In the offline context, researchers have explored various notions of fairness and their applicability in fair division problems [5, 107]. Some other literature takes into account fairness when developing efficient online resource allocation algorithms. [66, 82, 109, 136]. In this thesis, we concentrate on the offline setting, examining fairness guarantees across multiple contexts.

1.1 Outline of the Thesis and Our Results

First, we introduce some resource allocation models and related concepts of fairness and efficiency in Chapter 2. Some specific variants of the models and concepts used in subsequent chapters will be presented in further detail within those chapters for a clearer demonstration.

In Part I, we focus on efficiency in the online resource allocation setting. In particular, in Chapter 3, we examine a fully online matching problem in a stochastic setting. Specifically, each online arrival (including agents and items) follows a known identical and independent distribution (i.i.d.) over a fixed set of vertex types and will depart the system after an unknown sojourn time. Our goal is to match arrivals to maximize the total utilities earned by these arrivals. For this problem, we propose a linear program (LP)-based algorithm that achieves a competitive ratio of at least 0.1549 under mild conditions. We also provide several hardness results to illustrate the difficulty of this problem. We finally test our proposed algorithm numerically.

In Part II, we come to the fairness issue. Given that fairness cannot be achieved trivially even in offline resource allocation settings, in this part we mainly focus on the offline scenarios.

In Chapter 4, we consider the mixed goods setting, where both divisible and indivisible goods need to be allocated to a set of agents. Here, divisible goods represent the items that can be divided into multiple pieces and given to different agents, while each indivisible good can be allocated to only one agent. For this, we propose "up to a fraction" relaxations of two classical fairness notions called envy-freeness (EF) and proportionality (PROP) and demonstrate the existence of such fair allocations.

In Chapter 5, we examine an indivisible goods setting where agents' preferences over items are assumed to be *ordinal* and have *uncertainties*. We study stochastic dominance proportionality and provide complexity analysis for a series of problems related to finding allocations that are fair with high probability. We also propose multiple heuristics and test their performance numerically.

Finally, in Part III, we focus on achieving both efficiency and fairness simultaneously in offline resource allocation settings.

In Chapter 6, we study the concept of the price of fairness, which measures the worst-case efficiency loss when imposing fairness constraints. This can, to some extent, reflect the cost required to achieve fairness. For this, we provide a comprehensive characterization of the price of envy-freeness in various settings, including the two-agent case and the n-agent case for both indivisible goods and mixed goods settings.

In Chapter 7, we address a specific resource allocation problem for the cloud computing system, which examines a divisible goods setting where agents' preferences over items are assumed to be *Leontief preferences*. In this setting, we propose a new approximation ratio measure called *fair-ratio*, defined as the worst-case ratio between the optimal social welfare (or utilization) among all fair allocations and that achieved by the mechanism. This can demonstrate the difficulty of achieving high efficiency when the fairness constraints are required. We then present multiple novel mechanisms that can output a fair allocation with better efficiency guarantees under our proposed benchmark. We also test our mechanisms numerically.

In Chapter 8, we conclude the issues covered in this thesis and propose some possible future directions.

Chapter 2

Preliminaries

In this chapter, we introduce multiple resource allocation models and the relevant concepts that will be used in the following chapters. These include the fully online matching model under stochastic settings, the indivisible goods setting, the mixed goods setting, and the cloud computing setting. Additional preliminaries specific to each model will be described in the corresponding chapters.

2.1 Fully Online Matching Model

Let $[k] := \{1, 2, \ldots, k\}$ for any positive integer k. In this section, we consider the following online matching model where the arrivals follow a known identical and independent (i.i.d.) probability distribution. Specifically, before the online process, we are given an edge-weighted graph $G = (V, E)$, where each vertex $v \in V$ represents a vertex type, and each $e = (x, y) \in E$ represents an edge between x and y with utility $u_e \in \mathbb{R}_{\geq 0}$. Here, each online vertex must be of a type among these vertex types.

Given a total time horizon T, an online vertex represented by (x, d) arrives at each time $t \in [T]$, where $x \in V$ is the vertex type of this arrival, and d is the sojourn time. Here, the sojourn time is defined as the number of subsequent arrivals for which this vertex is willing to wait. For each online vertex (x, d), x is sampled from a known i.i.d. distribution $\{p_v\}$, where $\sum_{v \in V} p_v = 1$ and $\Pr[x = v] = p_v$ for each

$v \in V$. Then, d is sampled from an unknown discrete distribution $\mathbb{D}_x$. For each $v \in V$, only the expectation D_v of $\mathbb{D}_v$ is known.

During the online process, we can match two available vertices at any time. An online vertex is called *available* if it has not left the system or been matched to another vertex. A match between vertices of type x and y will create a utility of u_{xy}. Here, a match between two vertices of the same type is also meaningful since there may exist multiple vertices of the same type simultaneously in the system. Our goal is to maximize the total utility induced from the matches. In Chapter 3, we may use the vertex and the agent interchangeably.

Applications. The fully online matching model has broad applicability across various domains, including ride-sharing. In ride-hailing platforms, drivers and passengers arrive and depart dynamically, which requires efficient matching strategies to maximize the total reward. Additional applications are discussed in Chapter 3.

Competitive ratio. We use the competitive ratio to measure the performance of online algorithms. Specifically, for an online algorithm ALG and a problem instance I, we use $\text{ALG}(I)$ to represent the expected total utility of the matches output by this algorithm, where the expectation is taken over the random sequences of online arrivals and the randomization used in the algorithm. We also define $\text{OPT}(I)$ similarly, where OPT is an optimal offline algorithm that can utilize the information of all online arrivals. We can then define the competitive ratio of ALG as the minimum ratio of $\text{ALG}(I)$ over $\text{OPT}(I)$ among all possible instances I. In the following, we may drop the term I when there is no ambiguity.

Remark. When the graph G is a bipartite graph, this can be equivalent to the online resource allocation problem where both agents and items arrive dynamically and each agent can be allocated only one item. Thus, the setting we present here is more general and can be applied to a broader range of applications. For example, in a ridesharing system, the platform may try to match pairs of passengers to achieve a higher profit, where the underlying structure can be non-bipartite.

2.2 Indivisible Goods Setting

In this section, we explore the offline resource allocation setting, beginning with the indivisible goods model. This model is highly relevant in practical scenarios. For instance, when a school allocates courses to students, we would like to ensure the allocation is fair and satisfies the students' preferences to maintain their satisfaction.

In this setting, we need to allocate a set of m indivisible goods $M = \{g_1, g_2, \ldots, g_m\}$ to a set of n agents $N = [n]$. We call M' a *bundle* if M' is a subset of M.

Allocation. An *allocation* $\mathcal{A}$ is denoted by $(A_1, A_2, \ldots, A_n)$, where for each $i \in [n]$, A_i is the bundle allocated to agent i. An allocation $\mathcal{A}$ is *feasible* if all $A_i, i \in [n]$ are disjoint. We call a feasible allocation $\mathcal{A}$ *complete* if $\cup_{i\in[n]} A_i = M$. Otherwise, such a feasible allocation $\mathcal{A}$ is *partial.*

Utility. We denote the *utility profile* $\mathbf{u}$ by $(u_1, u_2, \ldots, u_n)$, where for each $i \in [n]$, u_i represents the non-negative utility function of agent i. Here, we assume each utility function is *additive.* That is, for any bundle $M' \subseteq M$ and any agent $i \in [n]$, the utility of agent i over this bundle M' is $u_i(M') = \sum_{g\in M'} u_i(g)$, where $u_i(g)$ is the utility of agent i over the item g.

Compared to the fully online matching model discussed earlier, the indivisible goods setting here is restricted to an offline framework, where matches occur only between goods and agents, forming a bipartite graph. Unlike the online setting, the indivisible goods model allows multiple goods to be assigned to a single agent and prioritizes fairness objectives in addition to the efficiency concerns emphasized in the fully online matching model.

2.3 Mixed Goods Setting

We then address the mixed goods model, where we need to simultaneously allocate both divisible and indivisible goods. This model applies to various real-world scenarios, such as inheritance division. In such cases, divisible goods like land or money and indivisible goods like cars or houses must be allocated together to achieve fairness or efficiency. Unlike the indivisible goods setting discussed earlier, this model involves allocating both indivisible and divisible goods. Specifically, in this setting, we need to allocate a set of m indivisible goods $M = \{g_1, g_2, \ldots, g_m\}$ and a cake $C = [0, 1]$ to a set of n agents $N = [n]$. We call $A' = M' \cup C'$ a *bundle* if M' is a subset of M and $C' \subseteq C$.

Allocation. An *allocation* $\mathcal{A}$ is denoted by $(A_1 = M_1 \cup C_1, A_2 = M_2 \cup C_2, \ldots, A_n = M_n \cup C_n)$, where for each $i \in [n]$, $A_i = M_i \cup C_i$ is the bundle allocated to agent i. An allocation $\mathcal{A}$ is *feasible* if $M_i \cap M_j = \emptyset$ and $C_i \cap C_j = \emptyset$ holds for any two different agents $i, j \in [n]$. We call a feasible allocation $\mathcal{A}$ *complete* if $\cup_{i\in[n]} M_i = M$ and $\cup_{i\in[n]} C_i = C$. Otherwise, such a feasible allocation $\mathcal{A}$ is *partial*.

Utility. We denote the *utility profile* $\mathbf{u}$ by $(u_1, u_2, \ldots, u_n)$, where for each $i \in [n]$, u_i represents the non-negative utility function of agent i. Here, we assume each utility function is *additive* and integrable over C. That is, for any bundle $A' = M' \cup C'$ and any agent $i \in [n]$, the utility of agent i over this bundle A' is $u_i(A') = \sum_{g\in M'} u_i(g) + \int_{C'} u_i(x)dx$, where $u_i(g)$ is the utility of agent i over the item g.

Robertson-Webb (RW) query model [122]. When accessing the utility function over the cake, we adopt the Robertson-Webb (RW) query model to solve this. Specifically, there are two feasible queries: 1) $eval_i(x, y)$: returns $u_i([x, y])$; 2) $cut_i(x, a)$: returns the leftmost point y such that $u_i([x, y]) = a$.

Multiple Homogeneous Goods Assumption. In our thesis, we also focus on a restricted setting called *multiple homogeneous goods setting.* Instead of utilizing a heterogeneous and integrable cake C to represent the divisible goods, we use a set of $\bar{m}$ *homogeneous divisible goods* $D = \{d_1, d_2, \ldots, d_{\bar{m}}\}$ to denote the divisible goods.

We then correspondingly modify the definitions of bundle, allocation, and utility, while the remaining keeps exactly the same. We call $A' = (M', \mathbf{x}' = (x_1, x_2, \ldots, x_{\bar{m}}) \in [0, 1]^{\bar{m}})$ a *bundle* if M' is a subset of M. Here, each $x_i, i \in \bar{m}$ specifies the received fraction of the corresponding homogeneous divisible good d_i.

An *allocation* is denoted by $\mathcal{A} = (A_1 = (M_1, \mathbf{x}_1), A_2 = (M_2, \mathbf{x}_2), \ldots, A_n = (M_n, \mathbf{x}_n))$, where for each agent $i \in [n]$, A_i is the bundle allocated to agent i.

The utility of agent i over a bundle $A' = (M', \mathbf{x}')$ is $u_i(A') = \sum_{g \in M'} u_i(g) + \sum_{k \in \bar{m}} x'_i \cdot u_i(d_k)$, where $u_i(g)$ and $u_i(d_k)$ are the utilities of agent i over the indivisible good g and the homogeneous divisible good d_k, respectively.

2.4 Cloud Computing Setting

In this section, we explore the cloud computing setting, a divisible goods model characterized by Leontief preferences. Unlike the previous two sections, this setting emphasizes the allocation of divisible goods under a significantly different utility function.

In this setting, we need to allocate a set of m divisible goods $R = \{r_1, r_2, \ldots, r_m)$ to a set of n agents $N = [n]$. We call $A' = (A'_1, A'_2, \ldots, A'_m) \in [0,1]^m$ a *bundle*, where for each $i \in [m]$, A'_i specifies the received fraction of the divisible good r_i.

Allocation. An allocation $\mathcal{A}$ is denoted by $(A_1, A_2, \ldots, A_n)$, where for each $i \in [n]$, $A_i = (A_{i1}, A_{i2}, \ldots, A_{im})$ is the bundle allocated to agent i. An allocation $\mathcal{A}$ is *feasible* if $\sum_{i \in [n]} A_{ij} \leq 1$ holds for any $j \in [m]$.

Utility. We denote the *utility profile* $\mathbf{u}$ by $(u_1, u_2, \ldots, u_n)$, where for each $i \in [n]$, u_i represents the non-negative utility function of agent i. Here, we assume agents have *Leontief preferences*. That is, for each agent $i \in [n]$, we can define a normalized demand vector $\mathbf{d}_i = \{d_{i1}, d_{i2}, \ldots, d_{im}\}$, where $d_{ij} \in (0,1]$ holds for each $j \in [m]$ and there exists some $j \in [m]$ such that $d_{ij} = 1$. According to the demand vector $\mathbf{d}_i$ of agent $i \in [n]$, the utility of agent i over a bundle $A' \in [0,1]^m$ is $u_i(A') = \max\{y \in \mathbb{R}_+ : \forall j \in [m], A'_j \geq y \cdot d_{ij}\}$.

2.5 Efficiency and Fairness Notions

For efficiency, we utilize the social welfare to measure this.

Social Welfare. The *social welfare* of an allocation $\mathcal{A} = (A_1, A_2, \ldots, A_n)$ is defined as $\mathrm{SW}(\mathcal{A}) := \sum_{i \in [n]} u_i(A_i)$.

For fairness, we first introduce the two most fundamental fairness notions used in fair division, *envy-freeness (EF)* and *proportionality (PROP)*.

Definition 2.1 (EF & PROP [68, 126])**.** An allocation $\mathcal{A} = (A_1, A_2, \ldots, A_n)$ is said to satisfy

- *envy-freeness (EF)* if $u_i(A_i) \geq u_i(A_j)$ holds for any two agents $i, j \in [n]$;
- *proportionality (PROP)* if $u_i(A_i) \geq u_i(M)/n$ holds for any agent $i \in [n]$.

Informally, an allocation is EF if every agent does not envy any other agent's bundle, and is PROP if each agent receives enough resources which are at least $1/n$ of the utility when she receives all the resources, where n is the number of agents. In the following, we may use PROP and share incentive (SI) interchangeably, as they are defined exactly the same.

However, in instances where a valuable indivisible item needs to be allocated to two agents, EF or PROP allocations may not exist. Therefore, some classical relaxations of these two fairness notions have been proposed for the indivisible goods setting.

Definition 2.2 (EF1 & EFX & PROP1 [46, 50, 58, 106, 118])**.** An allocation $\mathcal{A} = (A_1, A_2, \ldots, A_n)$ is said to satisfy

- *envy-freeness up to one good (EF1)* if for any two agents $i, j \in [n]$, $u_i(A_i) \geq u_i(A_j \setminus \{g\})$ holds for *some* item $g \in A_j$;
- *envy-freeness up to any good (EFX)* if for any two agents $i, j \in [n]$, $u_i(A_i) \geq u_i(A_j \setminus \{g\})$ holds for *any* item $g \in A_j$;
- *proportionality up to one good (PROP1)* if for any agent $i \in [n]$, $u_i(A_i \cup \{g\}) \geq u_i(M)/n$ holds for some item $g \notin A_i$.

Here, EF1 ensures that the envy between two agents can be resolved by removing a single good, and PROP1 requires each agent should receive enough resources which are at least $1/n$ of the utility when she receives all the resources, after grabbing an additional good from some other agent's bundle. EFX is a stronger version of EF1, which ensures that the envy between two agents can be resolved by removing *any* single good.

For the mixed goods setting, two additional fairness notions have been proposed [27, 116].

Definition 2.3 (EFM [27])**.** An allocation $\mathcal{A} = (A_1 = M_1 \cup C_1, A_2 = M_2 \cup C_2, \dots, A_n = M_n \cup C_n)$ is said to satisfy *envy-freeness for mixed goods (EFM)* if for any two agents $i, j \in [n]$,

- if agent j's bundle contains only indivisible goods, that is, $C_j = \emptyset$ and $M_j \neq \emptyset$, $u_i(A_i) \geq u_i(A_j \setminus \{g\})$ holds for *some* good $g \in M_j$;
- otherwise, $u_i(A_i) \geq u_i(A_j)$.

In words, for any two agents i and j, agent i does not EFM-envy agent j if (1) agent i does not envy agent j, or (2) agent j's bundle only contains indivisible goods and agent i does not EF1-envy agent j. EFM serves as a stronger notion than EF1 as condition (2) forces j's bundle to only contain indivisible goods if agents i envies j.

Definition 2.4 (EFXM [116])**.** An allocation $\mathcal{A} = (A_1 = M_1 \cup C_1, A_2 = M_2 \cup C_2, \dots, A_n = M_n \cup C_n)$ is said to satisfy *envy-freeness up to any good for mixed goods (EFXM)* if for any two agents $i, j \in [n]$,

- if agent j's bundle contains only indivisible goods, that is, $C_j = \emptyset$ and $M_j \neq \emptyset$, $u_i(A_i) \geq u_i(A_j \setminus \{g\})$ holds for *any* good $g \in M_j$;
- otherwise, $u_i(A_i) \geq u_i(A_j)$.

EFXM is a stronger version of EFM since it replaces the EF1-envy by EFX-envy in the condition (2) mentioned above.

Part I

Efficiency in Online Resource Allocation

Chapter 3

Fully Online Matching in Stochastic Settings

In the first part of our thesis, we mainly focus on efficiency. Since simple matching algorithms or linear programming typically suffice for finding an efficient allocation in offline resource allocation, we will concentrate on the online setting and investigate the efficiency guarantee in the fully online matching model. This chapter has been published in Li et al. [104].

3.1 Introduction

Starting from the seminal work by Karp et al. [93], online matching has been a fundamental research topic in online resource allocation. Many online matching studies focus on online bipartite matching, where vertices on one side are assumed to be known upfront, and those on the other side arrive online. However, this setting fails to model some modern applications, such as ride-sharing, where all vertices arrive online and depart after a sojourn time. This chapter studies this general setting. In particular, all vertices arrive in the system in an online manner. When a vertex arrives, the edges with the previously arrived vertices are revealed. A vertex will be matched to another unmatched *neighboring vertex* (linked to the vertex by an incident edge) before its departure or be left unmatched and depart. The goal is to maximize the total reward of successful matches. We name this general problem a fully online matching problem.

Fully online matching generalizes online bipartite matching in several dimensions (e.g., from a bipartite graph to a general graph, all agents arrive online) and is hence much more complicated. There is limited literature on the related study. Huang et al. [85] and Huang et al. [86] are inspiring ones. They assume that agents arrive and depart in an adversary manner. Their goal is to maximize the number of matches. In contrast, in this chapter we assume arrivals follow an identical and independent (i.i.d.) probability distribution, which is a common assumption in online matching literature [67, 83, 87, 88]. Upon arrival, each agent will stay in the system for a sojourn time before leaving the system. We do not specify the exact distribution of the sojourn time but assume it follows a type-specific distribution with known expectations. In addition, we consider maximizing the edge-weighted reward of successful matches. We claim the model settings considered in this chapter are more applicable in ride-sharing. The arrival distribution can be easily estimated using customers' arrival data. However, the data on sojourn time is often less available. Hence we consider a distributionally free setting, without assuming a specific distribution but require information on the mean sojourn time. Finally, the rewards from different paired agents are often different. Our model captures this feature by maximizing the total weight of matched pairs.

Fully online matching has potential applications in various domains besides ride-sharing. For example, in a chess game platform, players join the platform in an online manner and will wait for an opponent to match for only a limited time. We can measure the quality of a match by the rating difference between the two matched players. The platform's goal is to maximize the total quality of successful matches. Another example is kidney exchange. Donors and recipients arrive in the market sequentially and stochastically. The lifetime for recipients and kidneys is limited. They must be matched within their lifetime, otherwise, they will be abandoned. The goal is to maximize the total matching quality.

3.1.1 Our Contributions

We summarize the main contributions in this chapter as follows. We study a fully online matching model with stochastic arrivals and departures. In particular, the arrivals follow a known i.i.d. distribution, and the sojourn time before agents depart can follow a large family of distributions with a known expectation. The goal is to maximize the total weight (defined on edges) of successful matches. The model settings are applicable in a wide family of applications, including ride-sharing.

We design a *distributionally free* LP-based algorithm, and investigate its performance measured by a competitive ratio, see Theorem 3.7. Under mild assumptions, we prove that the competitive ratio of our algorithm is at least 0.1549, see Corollary 3.8. Moreover, for some specific distributions, we can achieve better competitive ratios when we restrict the choices of the parameters of these distributions, see Corollaries 3.9, 3.10 and 3.11.

We further establish two hardness results to foreground the technical challenges of the problem we study. We first show no online algorithm can achieve a competitive ratio of more than $\frac{2}{3}$, see Theorem 3.12. But if we restrict the algorithms to LP-based algorithms with respect to the LP we derive, we show that there exists no such online algorithm with a competitive ratio larger than $\frac{1}{3}$, see Theorem 3.13.

We conduct extensive numerical studies to evaluate the performance of our algorithms. Our algorithms can significantly outperform the baseline algorithms from related works in most parameter settings.

3.1.2 Related Work

There is extensive literature on online bipartite matching, where there exists a set of offline vertices, and each online vertex will be matched to an offline vertex *immediately* upon its arrival or be rejected. A seminar work by [93] considered maximizing the number of matches when agents arrive in an adversary setting. Many works further consider generalizing the objective to maximizing vertex-weighted (or edge-weighted) matching, and the arrival process to a stochastic process [1, 67, 83, 87]. To the best of our knowledge, the best bound under a stochastic arrival model is achieved by [87]. They provided a 0.716-competitive algorithm in a vertex-weighted setting and a 0.706-competitive algorithm in an edge-weighted with free disposal setting, i.e., each offline vertex can update its matching vertex upon new arrivals.

Recently, fully online matching has attracted increasing attention, where each vertex has its arrival and departure time and can be matched *anytime* before it departs. In other words, a delay in matching is allowed. Our work in this chapter lies in this steam of research. Starting from a non-weighted setting, Eckl et al.

[63], Huang et al. [84, 85, 86] studied fully online matching with adversarial arrivals and departures, and provided a 0.569-competitive algorithm and hardness results. Considering edge-weighted reward and assuming fixed and identical sojourn time, Ashlagi et al. [8] proposed a 0.25-competitive algorithm when agents arrive in an adversary manner and a 0.279-competitive algorithm when the arrival sequence follows a random order model. Several papers focus on the setting where both arrival and departure follow a type-specific Poisson process. Collina et al. [57] proposed a 0.125-competitive algorithm for an edge-weighted setting, where the goal is to maximize total weights defined on edges. Aouad and Saritac [6] studied a dynamic stochastic matching with the same arrival and departure process. They model the problem as an infinite-horizon continuous-time Markov decision process and provide an approximation policy that can achieve $\frac{e-1}{4e} \approx 0.158$ of the optimality, in sharp contrast with the competitive ratio for online matching problems. Our work in this chapter differs from those papers in the following perspectives. First, all online vertices arrive according to a known i.i.d. distribution. Second, we do not assume a specific type of distribution for agents' sojourn time. In other words, our algorithm works for a large family of distributions with a known expectation and is robust when the distribution varies.

Another stream of literature models the delay in matching by incorporating the delay cost in the total cost function and makes the matching decision to minimize cost [7, 11, 12, 65, 132]. It is in contrast to the modeling perspective in fully online matching literature, where a hard constraint for the match to be restricted in a time interval is imposed.

3.2 Linear Programming Benchmark

To bound the competitive ratio, we first provide a linear program to bound the OPT. We define a variable n_{xy} for each ordered pair (x, y) where $x, y \in V$. We then define a benchmark LP (3.1) as follows.

$$\max \sum_{x,y \in V} w_{xy} n_{xy} \tag{3.1}$$

$$\text{s.t.} \sum_{y \in V} n_{xy} + \sum_{y \in V} n_{yx} \le p_x T, \qquad \forall x \in V, \tag{3.1a}$$

$$n_{xy} \le p_x T p_y D_x, \qquad \forall x, y \in V, \tag{3.1b}$$

$$n_{xy} \ge 0, \qquad \forall x, y \in V, \tag{3.1c}$$

In LP (3.1), each variable n_{xy} denotes the expected number of times that an online agent of type y is matched to an unmatched online agent of type x. Constraints (3.1a) upper bound each type x's total expected number of matches by its expected number of occurrences. Constraints (3.1b) restrict n_{xy} by the total number of occurrences of the event that an online agent of type y is in the sojourn time of x.

We then show in Lemma 3.1 that LP (3.1) is a relaxation of the offline optimal. The intuition behind the proof is as follows. We use n^*_{xy} to denote the optimal solution to OPT. We then show such $\{n^*_{xy}\}$ is a feasible solution to LP (3.1).

Lemma 3.1. *For any instance I, the optimal value of LP (3.1) is an upper bound of OPT(I).*

Proof. Let r denote a realization of instance I as one possible input sequence of agent types with the corresponding sojourn time after randomization. We further define $n_{r,x,y}$ as the number of matches between one agent of type y and one previous agent of type x under realization r given by OPT and P_r as the probability of the realization r.

We can define $n^*_{xy} = \sum_r P_r n_{r,x,y}$. Since $\text{OPT}(I) = \sum_r P_r \sum_{x,y \in V} n_{r,x,y} w_{xy} = \sum_{x,y \in V} w_{xy} \left(\sum_r P_r n_{r,x,y}\right)$, which is equal to $\sum_{x,y \in V} w_{xy} n^*_{xy}$. It suffices to show $\{n^*_{xy}\}$ is one feasible solution to LP (1).

The first is to check the feasibility of Constraints (1a). $\forall x \in V$, $\sum_{y \in V} n^*_{xy} + \sum_{y \in V} n^*_{yx} = \sum_r P_r (\sum_{y \in V} n_{r,x,y} + n_{r,y,x}) \le \sum_r P_r \cdot [\text{number of type } x \text{ in realization } r] = p_x T$. The inequality is because the number of matched agents of type x cannot exceed the number of appearing agents of type x in any realization. The last equality is from the linearity of expectation.

Since Constraints (1c) are obviously satisfied, the remaining is to show Constraints (1b) are satisfied. $\forall x, y \in V$, we have $n^*_{xy} = \sum_r P_r n_{r,x,y}$, which is weakly smaller than $\sum_r P_r \cdot$ [number of the occurrence of event E in r], where event E is that one agent of type x can see one following agent of type y. The last inequality has the same reason as that of the above inequality. By linearity of expectation, we can transform it into the sum of expected times of one agent of type x at time t seeing one following agent of type y over all t from 1 to T, which is upper bounded by the multiple of Tp_x and the expected times of one following agent of type y existing in the sojourn time of one agent of type x. We assume the support set of $\mathbb{D}_x$ is S_x where the probability of support $s \in S_x$ is q_s. So n^*_{xy} is upper bounded by $Tp_x \sum_{s \in S_x} q_s \cdot p_y s$ from linearity of expectation, which is equal to $p_x T p_y D_x$. □

For analysis convenience, we let α_{xy} be $\frac{n_{xy}}{p_y T}$ for all $x, y \in V$. $\alpha_{xy} \leq 1$ according to Constraints (3.1a). Then we can reformulate LP (3.1) as LP (3.2), and we will use LP (3.2) in the following analysis.

$$\mathbf{max} \sum_{x,y \in V} w_{xy} \alpha_{xy} p_y T \tag{3.2}$$

$$\mathbf{s.t.} \sum_{y \in V} \alpha_{xy} p_y + \sum_{y \in V} \alpha_{yx} p_x \leq p_x, \quad \forall x \in V, \tag{3.2a}$$

$$\alpha_{xy} \leq p_x D_x, \quad \forall x, y \in V, \tag{3.2b}$$

$$\alpha_{xy} \in [0, 1], \quad \forall x, y \in V, \tag{3.2c}$$

3.3 Approximation Algorithm

Inspired by the algorithm used in Collina et al. [57], we propose our LP-based Algorithm 1. In the algorithm, we set the matching probability according to the optimal solution $\{\alpha_{xy}\}$ to LP (3.2). Specifically, the matching probability between an arriving agent of type y and an existing agent of type x is set to $\gamma \cdot \alpha_{x,y}/(p_x D_x)$, where γ is a scaling parameter and the term $1/(p_x D_x)$ is designed to increase the matching probability appropriately. The matching probability is not greater than 1 according to Constraints (3.2b) and $\gamma \leq 1$. We use J to denote the *multiset* of types of all existing unmatched agents when an agent i of type $y \in V$ arrives. We enumerate all elements x in J in a uniformly random order and match agent i with the specific agent j of type x with the above probability (Lines 5-6 in Algorithm

Algorithm 1 SAM(γ)

Input: Online arrivals of agents
Parameter: Scaling parameter $\gamma \in (0, 1]$

1: $\{\alpha_{xy}\}$:= Solution to LP (3.2);
2: **for** each arriving agent i whose type is $y \in V$ **do**
3:     J := The multiset of types of unmatched agents;
4:     **for** each type $x \in J$ in a uniformly random order **do**
5:         j := The corresponding unmatched agent of type x;
6:         Match i and j w.p. $\gamma \cdot \alpha_{xy}/(p_x D_x)$;
7:     **end for**
8: **end for**

1). When an agent j is matched with i successfully, no further enumeration is needed. Algorithm 1 is *solvable in polynomial time* since LP (3.2) can be solved in polynomial time and the number of computations per arrival is $O(|J|)$ where the size $|J|$ of the set J defined in Line 3 of Algorithm 1 can be bounded by the maximum support among all $\mathbb{D}_v$s of $v \in V$.

We next analyze the competitive ratio of Algorithm 1. In the following analysis in this section, we assume the maximal value in the support of the distribution $\mathbb{D}_v$ is much lower than T for each $v \in V$. The assumption is mild in ride-sharing applications since the time horizon is much larger than the possible sojourn time of every agent.

3.3.1 Analysis

Note that the total weight generated by OPT cannot be greater than the optimal value of LP (3.2) from Lemma 3.1, we can compare the performance of Algorithm 1 with the value of LP (3.2) to get a lower bound of the competitive ratio. Thus, the strategy of calculating the competitive ratio is to lower bound the ratio between the expected number of successful matches and the term $\alpha_{xy} p_y T$ in the objective function of LP (3.2), for each ordered pair (x, y), where $x, y \in V$. Here, we only consider the pair (x, y) such that $D_x > 0$ since agents of type x will not wait otherwise.

We assume an agent i of type $y \in V$ arrives at time t. We will calculate the probability of matching this agent to an existing agent j of type $x \in V$ who arrives at time $t' < t$ by considering four events separately.

The first event E_1 is defined as an agent j of type x arrives at time t'. From the known i.i.d. arrival setting we can easily derive Lemma 3.2.

Lemma 3.2. *The probability of E_1 is p_x.*

The second event is defined to calculate the probability of the agent j who arrives earlier (at $t' < t$) and is of type x being unmatched. We denote this event as E_2. In our proof, we use a vector $\vec{b}$ to store the information of unmatched agents at time t', where each element b_z equals the number of type z in set J defined in Line 3 of Algorithm 1. By conditioning on the probability distribution over $\vec{b}$, we upper bound the probability that there is one agent of type $z \in V$ matching to the agent j. By union bound, we get Lemma 3.3.

Lemma 3.3. *The probability of E_2 is at least $1 - \gamma$.*

Proof. We use a vector $\vec{b}$ to store the information of unmatched agents at time t', i.e. the element b_z is equal to the number of z in set J in Line 3. Thus,

$$\begin{aligned}
\Pr[E_2] &= \sum_{\vec{b}} \Pr[\vec{b}] \Pi_{z \in V} \left(1 - \frac{\gamma \alpha_{zx}}{p_z D_z}\right)^{b_z} \\
&\geq \sum_{\vec{b}} \Pr[\vec{b}] \left(1 - \gamma \sum_{z \in V} \frac{b_z \alpha_{zx}}{p_z D_z}\right) \\
&= 1 - \gamma \sum_{z \in V} \frac{\alpha_{zx}}{p_z D_z} \sum_{\vec{b}} \Pr[\vec{b}] b_z \\
&\geq 1 - \gamma \sum_{z \in V} \frac{\alpha_{zx}}{p_z D_z} \cdot p_z D_z \\
&= 1 - \gamma \sum_{z \in V} \alpha_{zx} \geq 1 - \gamma
\end{aligned}$$

The first equality is from the description of Algorithm 1. The first inequality is from the union bound, and the second inequality is because at any fixed time, the total number of remaining unmatched agents of one fixed type is not greater than the total number of all existing agents of this type. The last inequality is from Constraints (3.2a) after ignoring the first term in the left hand side. □

Next, we use E_3 to represent the event that no arriving agent between time $t' + 1$ and $t - 1$ matches agent j given the occurrence of events E_1, E_2.

Lemma 3.4. *If $D_x \geq 1$, the probability of E_3 is at least* $\left(1 - \frac{\gamma}{D_x}\right)^{t-t'-1}$.

The sketch of this proof is to first provide an upper bound of the event that an arriving agent at time $t'' \in [t'+1, t-1]$ matches agent j. Using the independence between different t'', we prove Lemma 3.4.

Proof.

$$\begin{aligned}\Pr[E_3] &\geq \left(1 - \sum_{z \in V} p_z \frac{\gamma \alpha_{xz}}{p_x D_x}\right)^{t-t'-1} \\ &= \left(1 - \frac{\gamma}{p_x D_x} \sum_{z \in V} p_z \alpha_{xz}\right)^{t-t'-1} \\ &\geq \left(1 - \frac{\gamma}{D_x}\right)^{t-t'-1}\end{aligned}$$

The first inequality is because at each time between $t'+1$ and $t-1$, the matching probability of the arriving agent of one possible type z at this time with agent j is upper bounded by $\frac{\gamma \alpha_{xz}}{p_x D_x}$ by the description of our algorithm. The second inequality is from Constraint (3.2a) after ignoring the second term in the left hand side. □

The remaining is to measure the probability of the agent i of type y matching the agent j of type x given the fact that agent j is unmatched before time t. We denote this event as E_4.

Lemma 3.5. *The probability of E_4 is at least* $\frac{\gamma \alpha_{xy}}{p_x D_x}\left(1 - \frac{\gamma}{2}\right)$.

The intuition behind the proof is that there are two parts needed to match i to j successfully. The first is that j can be matched to i in Line 6 of Algorithm 1, and the second is that all unmatched type $z \in J$ that joins J earlier than the agent j's type x cannot match i successfully. The first can be easily calculated, while the second needs to utilize the uniformly random order of elements in J and apply similar arguments as in the proof of Lemma 3.3.

Proof. We define $\vec{b}$ in a similar way as above, but here there are two differences from the above $\vec{b}$ in the proof of Lemma 3.3. The first is that we focus on the time

t but not t' here, while the second is that the corresponding x of agent j is not counted in $\vec{b}$.

$$\begin{aligned}
\Pr[E_4] &\geq \frac{\gamma\alpha_{xy}}{p_x D_x}\left(1 - \sum_{\vec{b}} \Pr[\vec{b}] \sum_{z \in V} \frac{\gamma\alpha_{zy} b_z}{2 p_z D_z}\right) \\
&= \frac{\gamma\alpha_{xy}}{p_x D_x}\left(1 - \frac{\gamma}{2} \sum_{z \in V} \frac{\alpha_{zy}}{p_z D_z} \sum_{\vec{b}} \Pr[\vec{b}] b_z\right) \\
&\geq \frac{\gamma\alpha_{xy}}{p_x D_x}\left(1 - \frac{\gamma}{2} \sum_{z \in V} \frac{\alpha_{zy}}{p_z D_z} p_z D_z\right) \\
&= \frac{\gamma\alpha_{xy}}{p_x D_x}\left(1 - \frac{\gamma}{2} \sum_{z \in V} \alpha_{zy}\right) \geq \frac{\gamma\alpha_{xy}}{p_x D_x}\left(1 - \frac{\gamma}{2}\right)
\end{aligned}$$

The reason for the first inequality is that if we want to match i and j successfully, besides that i can match j in Line 6 corresponding to the first term, all $z \in J$ before the corresponding x of j cannot match i successfully. Because the probability of one $z \in J$ before the corresponding x of j is $\frac{1}{2}$ from the uniformly random order and the matching probability is $\frac{\gamma\alpha_{zy}}{p_z D_z}$, we can calculate one upper bound of the probability of the event that there exists one $z \in J$ before the corresponding x of j matching i successfully by union bound.

The second inequality is because the total number of unmatched agents of each type z is not greater than the total number of existing agents of type z at time t, which is further upper bounded by $p_z D_z$ because of the difference of the existence of agent j at time t'. The last inequality is again from Constraints (3.2a) after ignoring the first term in the left hand side. □

Besides the analysis of the four events, we need to utilize another lemma to calculate the total ratio between the expected matching number of (x, y) and the term $\alpha_{xy} p_y T$, which can be proved by induction.

Lemma 3.6. $(1-x)^d \leq 1 - dx + \frac{d(d-1)}{2} x^2$, *for all* $x \in [0, 1]$ *and all non-negative integer* d.

Proof. We can prove this by a simple induction. When $d = 0$, this is satisfied. It suffices to show it's satisfied for $d + 1$ when it's true for d.

$$\begin{aligned}(1-x)^{d+1} &= (1-x)^d(1-x) \\ &\leq \left(1 - dx + \frac{d(d-1)}{2}x^2\right)(1-x) \\ &= 1-(d+1)x + \frac{d(d-1)}{2}x^2 \\ &+ dx^2 - \frac{d(d-1)}{2}x^3 \\ &\leq 1-(d+1)x + \frac{d(d+1)}{2}x^2.\end{aligned}$$

□

Now we are ready to formally present our main result.

Theorem 3.7. *Under the assumption that $D_v \geq 1$ for all $v \in V$, the competitive ratio of Algorithm 1 with parameter γ is at least $\gamma(1-\gamma)\left(1-\frac{\gamma}{2}\right)\left(1-\frac{\gamma}{2}+\frac{\gamma}{2}C\right)$, where we define Var_v as the variance of the distribution $\mathbb{D}_v$ and $C = \min_{v\in V}\frac{D_v - Var_v}{D_v^2}$.*

Proof. We assume the support set S of the distribution D_x with probability q_s of support $s \in S$. Fix agent i of type y at time t, we want to calculate the expected matching number of agent j of type x, which should be equal to

$$\sum_{s\in S} q_s \sum_{t'=t-s}^{t-1} \Pr[E_1]\Pr[E_2]\Pr[E_3]\Pr[E_4].$$

Here, because of the assumption that $T \gg s$ and we will also enumerate all possible ts, the case that $t - s < 1$ can be ignored, so we can directly start t' from $t - s$ but not $\max\{1, t - s\}$.

Then, by observation, the lower bound of $\Pr[E_1]$, $\Pr[E_2]$ and $\Pr[E_4]$ don't contain the term t', we can directly move these terms outside and only focus on the value

$\sum_{s\in S} q_s \sum_{t'=t-s}^{t-1} \Pr[E_3]$, which is equal to

$$\begin{aligned}
&\sum_{s\in S} q_s \sum_{t'=t-s}^{t-1} (1-\gamma/D_x)^{t-t'-1} \\
&= \sum_{s\in S} q_s \frac{1-(1-\gamma/D_x)^s}{\gamma/D_x} \\
&\geq \sum_{s\in S} q_s \frac{s\cdot\gamma/D_x - \frac{1}{2}s(s-1)\cdot(\gamma/D_x)^2}{\gamma/D_x} \\
&\geq D_x - \frac{\gamma}{2D_x}\sum_{s\in S} q_s(s^2-s) \\
&= D_x + \frac{\gamma}{2} - \frac{\gamma}{2D_x}\left(Var_x + D_x^2\right) \\
&= D_x\left(1-\frac{\gamma}{2}+\frac{\gamma}{2}\frac{D_x - Var_x}{D_x^2}\right)
\end{aligned}$$

Here, the inequality is from Lemma 3.6. Since t can choose from 1 to T and agent i of type y will arrive w.p. p_y, the total expected number of ordered pairs (x, y) should be at least

$$\begin{aligned}
&Tp_y p_x(1-\gamma)\frac{\gamma\alpha_{xy}}{p_x D_x}\left(1-\frac{\gamma}{2}\right) D_x\left(1-\frac{\gamma}{2}+\frac{\gamma}{2}C\right) \\
&= Tp_y\alpha_{xy}\gamma(1-\gamma)\left(1-\frac{\gamma}{2}\right)\left(1-\frac{\gamma}{2}+\frac{\gamma}{2}C\right)
\end{aligned}$$

where $C = \min_{v\in V} \frac{D_v - Var_v}{D_v^2}$. □

Theorem 3.7 provides a lower bound of competitive ratio with respect to γ and C. Note that C depends on the mean and variance of the online arrivals' sojourn time. In the remaining part, we will discuss several special types of distributions to get their bounds. For each type, we can tune γ to achieve the best bound.

Corollary 3.8. *Under the assumptions that $D_v \geq Var_v$ and $D_v \geq 1$ for all $v \in V$, the competitive ratio of algorithm 1 is at least $\gamma(1-\gamma)\left(1-\frac{\gamma}{2}\right)^2$. By setting $\gamma^* = \frac{7-\sqrt{17}}{8} \approx 0.360$, the competitive ratio is at least 0.1549.*

We claim the assumptions in Corollary 3.8 are mild since they hold for many classic discrete distributions, such as binomial distribution, Poisson distribution, hypergeometric distribution, and geometric distribution with parameter $p^G \geq 0.5$.

3.3.2 Specific Distributions

In this part, we will discuss several special types of the distributions $\mathbb{D}_v$ on the achieved competitive ratios. First, we consider the case where the distribution $\mathbb{D}_v$ is a single-point distribution, i.e., each type $v \in V$ has fixed sojourn time D_v.

Corollary 3.9. *Under the assumption that each type $v \in V$ has fixed sojourn time D_v, the competitive ratio of algorithm 1 is at least 0.1588 by setting $\gamma^* \approx 0.373$.*

Proof. Most are similar to the proof of Theorem 3.7. We replace the inequality from Lemma 3.6 by the inequality $(1 - {}^{x}\!/_{d})^d \leq e^{-x}$ which is satisfied when $x \in [0, 1]$ and d is a positive number, and we have $\sum_{s\in S} q_s \sum_{t'=t-s}^{t-1} \Pr[E_3]$ is at least $\frac{1-e^{-\gamma}}{{}^{\gamma}\!/_{D_x}}$. Thus, the total expected number of ordered pair (x, y) is at least

$$\begin{aligned} &T p_y p_x (1-\gamma) \frac{\gamma \alpha_{xy}}{p_x D_x} \left(1 - \frac{\gamma}{2}\right) \frac{1 - e^{-\gamma}}{{}^{\gamma}\!/_{D_x}} \\ &= T p_y \alpha_{xy} (1-\gamma) \left(1 - \frac{\gamma}{2}\right) (1 - e^{-\gamma}). \end{aligned}$$

When $\gamma \approx 0.373$, the competitive ratio is the largest, which is ≈ 0.1588. □

Second, we consider three common discrete distributions: Poisson distribution $\mathrm{Poi}(\lambda^P)$, geometric distribution $\mathrm{Geo}(p^G)$ and binomial distribution $\mathrm{B}(n^B, p^B)$. The performance over these three distributions will also be further evaluated in the following experiments section. Since the expectation and the variance of $\mathrm{Poi}(\lambda^P)$ are both λ^P, we get the same competitive ratio as in Corollary 3.8 if $\lambda^P \geq 1$. For the remaining two distributions, from the simple expressions of the expectation and variance, we can transform the formula in Theorem 3.7 to $\gamma(1-\gamma)\left(1 - \frac{\gamma}{2}\right)(1-(1-p^G)\gamma)$ for $\mathrm{Geo}(p^G)$ and $\gamma(1-\gamma)\left(1 - \frac{\gamma}{2}\right)\left(1 - \frac{\gamma}{2}(1 - \frac{1}{n^B})\right)$ for $\mathrm{B}(n^B, p^B)$ to get the respective competitive ratios (see Corollaries 3.10 and 3.11, respectively).

Corollary 3.10. *Under the assumption that each distribution $\mathbb{D}_v$ is a geometric distribution Geo(p_v^G), by setting $\gamma^* = 0.35$, the competitive ratio of Algorithm 1 is at least $0.122 + 0.066p^G$, where p^G is the smallest p_v^G for all $v \in V$.*

Corollary 3.11. *Under the assumption that each distribution $\mathbb{D}_v$ is a binomial distribution $B(n_v^B, p_v^B)$ satisfying $D_v = n_v^B p_v^B \geq 1$, by setting $\gamma^* = 0.40$, the competitive ratio of Algorithm 1 is at least $0.154 + \frac{0.038}{n^B}$, where n^B is the largest n_v^B for all $v \in V$.*

Compared to the results in Corollary 3.8, we can get a better guarantee when $p^G > 0.5$ and $n^B < 38$, respectively. Moreover, the above choice of γ^* is from the consideration of being able to achieve relatively good performance over all possible values of p^G and n^B. If more refined ranges of p^G and n^B are given, we can adjust the value of γ^* to reach a better performance.

Remark 3.1. Algorithm 1 can achieve a similar performance guarantee if the arrival of agents in each type $v \in V$ is driven by an independent Poisson arrival process (see Huang and Shu [83] for details) and the corresponding $\mathbb{D}_v$ is an arbitrary (continuous or discrete) distribution with non-negative support. Specifically, after modifying the statements of some used lemmas such as Lemma 3.4, we can show the competitive ratio of Algorithm 1 is at least $\gamma(1-\gamma)\left(1-\frac{\gamma}{2}\right)\left(1-\frac{\gamma}{2}+\frac{\gamma}{2}C'\right)$, where $C' = \min_{v \in V} \frac{-Var_v}{D_v^2}$.

3.4 Hardness Results

In this section, we will present hardness results to demonstrate the challenge of the problem considered in this chapter. We will first show that no online algorithm can reach a competitive ratio better than $\frac{2}{3}$. Next, by restricting the algorithms to LP-based online algorithms with respect to our LP (3.2), we further show that no LP-based online algorithm with respect to LP (3.2) can obtain a competitive ratio better than $\frac{1}{3}$.

Theorem 3.12. *No online algorithm can reach a competitive ratio better than* $\frac{2}{3}$.

Proof. We consider such an instance:

- $T \to \infty$ and $V = \{1, 2\}$;
- $p_1 = \varepsilon$ and $p_2 = 1 - \varepsilon$ where ε is significantly small;
- $\mathbb{D}_1$ and $\mathbb{D}_2$ are both single-point distributions where $D_1 = 0$ and $D_2 = 1$;
- $w_{(1,2)} = \frac{1}{\varepsilon(1-\varepsilon)}$, $w_{(1,1)} = 0$ and $w_{(2,2)} = 1$;

We define $f(t)$ as the expected value of t rounds output by the online optimal algorithm given the first two agents are of type 2 and define $g(t)$ as the expected value of T rounds output by the online optimal algorithm given the first agent is of type 1. Our decision is needed only for each $f(t)$ with $t \geq 2$.

For $f(2)$, the optimal decision is to match the existing two agents of type 2, which means $f(2) = 1$. For $f(3)$, the value is the maximum of $q_3 \cdot w_{(2,2)} + (1 - q_3) \cdot (p_1 \cdot w_{(1,2)} + p_2 \cdot f(2))$, where $q_3 \in [0, 1]$ is the decision parameter such that we match the existing two agents of type 2 with probability q_3. Since $p_1 \cdot w_{(1,2)} = \frac{1}{1-\varepsilon} > 1 = w_{(2,2)}$, $q_3 = 0$ is the optimal strategy.

We next consider $f(t)$ with $t \geq 4$. We again denote $q_t \in [0, 1]$ as the decision parameter such that we match the first two agents of type 2 with probability q_t. If we match the first two agents, we get the expected value $1 + p_1 \cdot (0 + g(t-2)) + p_2 p_1 \cdot (w_{(1,2)} + g(t-3)) + p_2 p_2 \cdot f(t-2)$, and we denote it by A_t, where the three terms except the first one are corresponding to the following arrival sequence of type (1), $(2, 1)$ and $(2, 2)$, respectively. If we don't match the first two agents, we get the expected value $p_1 \cdot (w_{(1,2)} + g(t-2)) + p_2 \cdot f(t-1)$, and we denote it by B_t, where these two terms corresponding to the following arrival type 1 and 2, respectively. The value with respect to q_t is equal to $q_t A_t + (1 - q_t) B_t$. $f(t)$ is the optimum among them.

We then compare A_t and B_t. Since the representation of B_t also holds for the case when $t = 3$, we replace $f(t-1)$ in the representation of B_t by $f(t-1) \geq B_{t-1}$. So we have $B_t \geq 1 + \frac{1}{1-\varepsilon} + \varepsilon g(t-2) + \varepsilon(1-\varepsilon) g(t-3) + (1-\varepsilon)^2 f(t-2) > A_t$. Thus, $q_t = 0$ is the optimal strategy again.

To sum up, since $f(2) = 1$, the expected value output by the online optimal algorithm is not greater than the sum of the expected value output by the strategy which only matches agents between type 2 and type 1 and one.

We now compare the expected values output by the offline and the online optimal algorithm.

The offline optimal algorithm will match every pair of the type sequence $(2, 1)$, which is equal to $p_2 p_1 T w_{(1,2)} + o(T)$. Considering the expected matching number of type sequence $(2, 2)$, for every consecutive sequence of agents of type 2, if the total number len is even, the matching number is at least $(len - 2)/2$, while if the total number len is odd, the matching number is $(len - 1)/2$. With the fact that the total number of consecutive sequence is at most the number of agents of type 1 plus 1, the expected matching number of type sequence $(2,2)$ is lower bounded by $\frac{p_2-2p_1}{2}T + o(T)$. Thus, the expected value output by the offline optimal algorithm is $p_2p_1Tw_{(1,2)} + \frac{p_2-2p_1}{2}T + o(T)$.

Then, in the strategy which only matches agents between type 2 and type 1, the expected value is exactly $p_2p_1Tw_{(1,2)} + o(T)$. So the expected value output by the online optimal algorithm is $p_2p_1Tw_{(1,2)} + o(T)$.

Replacing all the variables by ε and T, the competitive ratio is $\frac{2}{3(1-\varepsilon)}$, which is $\frac{2}{3}$ when ε is significantly small. □

To capture the hardness of our problem based on the LP (3.2), we conclude the following theorem.

Theorem 3.13. *No LP-based online algorithm with respect to LP (3.2) can reach a competitive ratio better than* $\frac{1}{3}$.

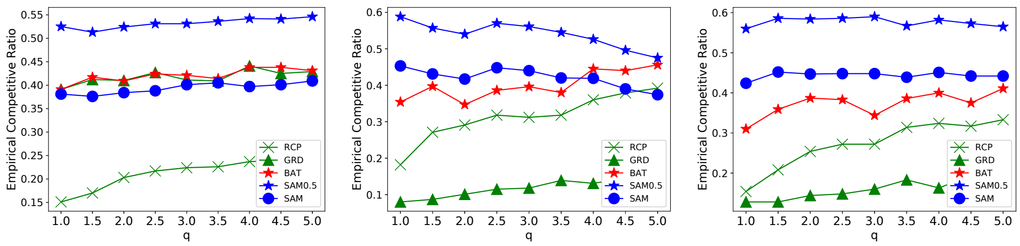
Proof. We consider such an instance:

- $T = 3$ and $V = \{1, 2\}$;
- $p_1 = p_2 = 0.5$;
- $\mathbb{D}_1$ and $\mathbb{D}_2$ are both single-point distributions where $D_1 = 2, D_2 = 0$;
- $w_{(1,2)} = 1$ and $w_{(1,1)} = w_{(2,2)} = 0$;

In this instance, the optimal value of LP (2) is $\frac{3}{2}$, while the offline optimal value is $\frac{1}{2}$. □

3.5 Experiments

In this section, we compare our algorithms to several baseline algorithms over synthetic datasets to demonstrate the effectiveness and efficiency of our algorithms.

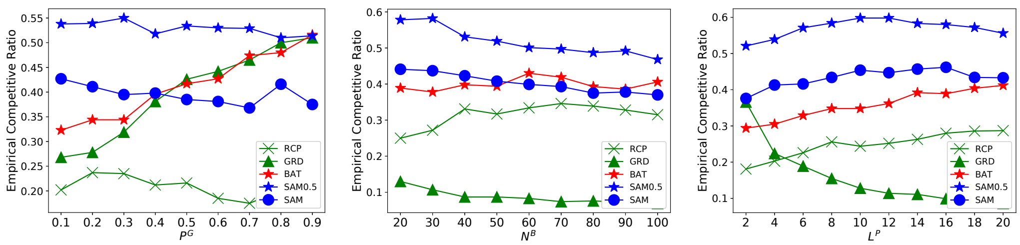


(A) Geo. Dist. $P^G = 0.5$

(B) Bin. Dist. $N^B = 30$

(C) Poi. Dist. $L^P = 10$

FIGURE 3.1: Performance of different algorithms w.r.t. different distributions and densities, $q = 1.0, 1.5, \dots, 5.0$



(A) Geo. Dist., $P^G = 0.1, 0.2, \dots, 0.9$

(B) Bin. Dist. $N^B = 20, 30, \dots, 100$

(C) Poi. Dist., $L^P = 2, 4, \dots, 20$

FIGURE 3.2: Performance of different algorithms w.r.t. different distributions, $q = 2.5$

3.5.1 Synthetic Datasets

We denote $U(a, b)$ as the uniform distribution on $[a, b]$, and $U_{int}[a, b]$ as the uniform distribution on $[a, b] \cup \mathbb{Z}$, where $\mathbb{Z}$ is the set of integers.

We generate a graph $G = (V, E)$ with $|V| = m = 100$ and a parameter density q. Without loss of generality, we set $V = \{1, 2, \dots, m\}$. For each pair $(x, y) \in V^2$ and $x \leq y$, we generate a value w'_{xy} from $U(0, 1)$. If the value $w'_{xy} \geq 1 - \frac{2q}{m+1}$, we add two non-trivial (positive-weighted) edges $e = (x, y)$ and $e = (y, x)$ with a weight $w_e = w'_{xy}$ to the edge set E. For the rest of the cases, we add trivial edges with $w_{xy} = 0$. It is straightforward to see that q is approximately the ratio between the number of non-trivial edges and the number of vertices (m). If a graph is sparse, q should be small compared to 1. The probability p_v of each type v is randomly generated from $U(0, 1)$, and then we normalize it to satisfy $\sum_{v \in V} p_v = 1$.

Three types of sojourn time distributions are tested:

- Geometric distribution Geo(p^G): $p^G \sim U(P^G, 1)$ where $P^G \in (0, 1)$ is a hyperparameter.
- Binomial distribution B(n^B, p^B): $n^B \sim U_{int}[10, N^B]$ and $p^B \sim U(0, 1)$ where $N^B \geq 10$ is a hyperparameter.
- Poisson distribution Poi(λ^P): $\lambda^P \sim U(0, L^P)$ where $L^P \geq 1$ is a hyperparameter.

We assume the sojourn time of all vertices in a graph follows the same type of distribution (geometric, binomial, or Poisson), and their distributions' parameters are randomly generated from a probability distribution. For example, if we assume vertices' sojourn time follow a geometric distribution with $P^G = 0.5$, then we will generate $p_v^G \sim U(0.5, 1)$ for each vertex $v \in V$.

In summary, a problem instance I is defined by a graph parametric by q and the type of distribution (geometric, binomial, or Poisson distribution) with its corresponding hyperparameter (P^G, N^B or L^P). For all experiments, we set $T = 3000$ which is much larger than the sojourn time of any vertex under any tested distribution.

3.5.2 Baseline Algorithms

- Rcp: This is the randomized compatibility policy from Appendix B.3 of [6]. We adjust it to make it suitable for our model.
- Grd: Each arrival is matched to an available neighboring vertex with an incident edge whose weight is the largest.
- Bat: This is the batching algorithm described in Section 4.1 of [8]. We set the batch size as $\lfloor \tilde{d} \rfloor + 1$ where $\tilde{d}$ is the expected sojourn time over all types.
- Sam0.5: Algorithm 1 with $\gamma = 0.5$.
- Sam: Algorithm 1 with $\gamma = 0.36$.

Here we test two different γs for Algorithm 1. Sam uses $\gamma = 0.36$ which is suggested by Corollary 3.8 for theoretical analysis. However, we note that Sam may be too

OPT	4.420	Bat	0.652
Rcp	1.131	Sam0.5	0.696
Grd	0.016	Sam	0.737

Table 3.1: Average runtimes of different algorithms (second)

conservative in practice. Hence, we would like to test a larger value. $\gamma = 0.5$ is selected since its associated lower bound for the competitive ratio is 0.141 according to Corollary 3.8, which is not bad in the theoretical bound but turns out to generate much better performance in expectation (the results will be discussed later). Note that we have tried many values for γ and obtained similar insights. These two values are chosen without loss of generality.

3.5.3 Performance criterion

Let r denote a realization of our generated instance and R as the set of r that we test. We use *empirical competitive ratio* (ECR) as our performance criterion for an algorithm ALG: ECR $= \frac{\sum_{r \in R} \mathrm{ALG}(r)}{\sum_{r \in R} \mathrm{OPT}(r)}$ where ALG(r) is the reward if we run ALG for r and OPT(r) is the hindsight optimal for r. For each parameter setting, we test $|R| = 50$ realized sequences.

3.5.4 Runtime

We list the average runtimes of different algorithms in Table 3.1. The parameters are $q = 2.5$ and geometric distribution with $P^G = 0.5$. We use Gurobi [78] as our solver. We use a computer with 2.2 GHz Intel Core i7 processor, 16 GB 1600 MHz DDR3 memory and Intel Iris Pro 1536 MB Graphics to run all the experiments. In this parameter setting, the most time-consuming benchmark is OPT and the runtimes of our algorithm are comparable with other baselines except the simple Grd algorithm which shows that our algorithms are efficient. Other parameter settings obtain similar results.

3.5.5 Results

Results are shown in Figures 3.1 and 3.2. In general, SAM0.5 outperforms other baselines by at least 10% in most parameter settings and SAM can dominate other baselines (except SAM0.5) in around $\frac{1}{3}$ test settings. As discussed earlier, SAM is too conservative to achieve a good performance in expectation, although it generates a good lower bound of the competitive ratio. SAM0.5 is good in both theoretical analysis and practice.

Sparsity. Figure 3.1 compares the performance under different distributions and densities with fixed hyperparameters. We can see that our algorithms' performance is stable when the density changes and SAM0.5 consistently outperforms all the other tested algorithms in all cases. In contrast, BAT, as the best baseline algorithm, does not perform as robust as ours. Its performance drops significantly when q decreases. However, in practice, the graph is often sparse. For instance, in ride-sharing, a non-trivial edge only exists between two vertices with close locations and arrival times. In other words, the advantage of our algorithms becomes more significant in applications with a sparse graph.

Diversity. Figure 3.2 compares the performance under different distributions and hyperparameters when fixing $q = 2.5$. Recall that the parameter of each vertex's distribution for sojourn time is uniformly generated from an interval defined by a hyperparameter (P^G, N^B, or L^P). The change of the hyperparameter will lead to different levels of diversity among agents (in terms of their sojourn time). For instance, for geometric distribution, when P^G decreases, the range to sample p^G for sojourn time's distribution gets larger, which leads to a higher level of diversity. In this case, BAT and GRD's performance drops significantly whereas our algorithms continue their good performance. This pattern is less significant for the other two distributions. But SAM0.5 consistently performs the best among all cases and outperforms the second-best algorithm by at least 10%.

In summary, our algorithms perform consistently well in all test cases and the advantage over the baseline algorithms is especially significant in a *sparse graph with heterogeneous agents*, which makes our algorithms practically relevant.

3.6 Conclusions

In this chapter, we study a general fully online matching model with stochastic arrivals and departures. We provide an LP benchmark for this problem and based on this LP, we design an algorithm with at least a 0.1549 competitive ratio. Our algorithm applies to a large family of departure distributions with a performance guarantee. To demonstrate the challenge of the problem, we further provide several hardness results. Specifically, we show that no algorithm can achieve a competitive ratio better than $\frac{2}{3}$ and no algorithm based on our LP can achieve a ratio better than $\frac{1}{3}$. Finally, we demonstrate the effectiveness and efficiency of our algorithm by conducting extensive numerical studies.

In the future, we may improve our LP benchmark by capturing the structure of the optimal offline solution. An extension to a fully online k-way matching (a match needs k agents) is also interesting.

Part II

Fairness in Offline Resource Allocation

Chapter 4

Allocating Mixed Goods with Customized Fairness

In this part, we address the fairness issue in the resource allocation problem. Given that fairness cannot be achieved trivially, we return to the offline setting and investigate the fairness guarantee across multiple offline resource allocation scenarios.

In this chapter, we first investigate the fairness under the mixed goods setting. This chapter has been published in Li et al. [99].

4.1 Introduction

Fair division of a mixture of divisible and indivisible goods has been well motivated since Bei et al. [27]. This scenario is exemplified in the context of dividing inheritances, where the assets include both money and land (divisible goods) as well as houses and cars (indivisible goods). In contrast to the division of purely divisible goods, one of the key challenges lies in defining and characterizing the notions of fairness that are both ideal and practical. This aspect continues to be a subject of ongoing discussion and exploration in the literature [95, 107], and our work contributes to this ongoing debate.

When the goods are all divisible, *envy-freeness* (EF) [68, 131] and *proportionality* (PROP) [126] are the prominent fairness notions, where the definitions of these two notions can be referred to Chapter 2. When the goods are all indivisible, due to the fact that EF and PROP allocations barely exist, the "up to one" relaxation is one of the most widely accepted notions, such as *envy-freeness up to one good* (EF1) [46, 106], and *proportionality up to one good* (PROP1) [58]. EF1 and PROP1 have some nice properties, such as guaranteed existence, simple computation, and being compatible with Pareto optimality (PO) [50].

When the goods are mixed, EF1 and PROP1 can be directly applied and guaranteed to be satisfiable, by treating the divisible goods as hypothetical infinitesimally indivisible units. However, these "up to one" relaxations are rather weak fairness criteria, as the presence of divisible goods can help alleviate the burden of unfairness. In light of this, Bei et al. [27] introduced *envy-freeness for mixed goods* (EFM), whose existence is also guaranteed.

Apart from EFM, a more straightforward approach to enhance EF1 and quantify the help of divisible goods in achieving fairness is to directly strengthen the "up to one" relaxation to the "up to a fraction", and the specific fraction depends on the portion of indivisible goods in relation to all goods. Intuitively, an agent may desire fairer allocations when her portion of divisible goods is more valuable. One possible way to quantify the portion of (in)divisible goods for each agent i is through her *indivisibility ratio* α_i, where α_i represents the portion of utility derived from indivisible goods. Then, an allocation is *envy-free up to α-fraction of one good* (EFα) to agent i if any envy she has towards another agent j can be resolved by obtaining an α_i fraction of some indivisible item from agent j's bundle. Similarly, an allocation is *proportional up to α-fraction of one good* (PROPα) to agent i if her utility remains at least $1/n$ after acquiring an α_i fraction of some indivisible item from another agent's bundle. It is important to note that the "up to α" relaxation allows for varying indivisibility ratios among the agents, thereby tailoring the evaluation of fairness based on each agent's specific perspective. In this work, we focus on EFα and PROPα.

Example. To illustrate the difference between EFM and EFα, we consider the following example, where two indivisible goods $M = \{g_1, g_2\}$ and one cake C are allocated to three identical agents. The utility function $u(\cdot)$ is shown in Table 4.1. Allocation $\mathcal{A} = (A_1, A_2, A_3)$ with $A_1 = A_2 = \frac{1}{2}C$ and $A_3 = \{g_1, g_2\}$ is EFM, but it is not EFα since removing 0.5 fraction from item g_1, the remaining utility of A_3 is $0.25 \times 0.5 + 0.25 = 0.375$ which is still greater than 0.25.

From this example, we observe that when the indivisibility ratio is small, EFα can ensure a fairer or more balanced allocation which can be closer to EF. In contrast, EFM may return an allocation that appears somewhat unfair due to its adherence to the EF1 criteria for bundles comprising solely indivisible goods. Furthermore, unlike EFM, when agents are non-identical, EFα guarantees customized fairness based on various personalized indivisibility ratios.

	g_1	g_2	C	α
$u(\cdot)$	0.25	0.25	0.5	0.5

TABLE 4.1: An Example on EFM v.s. EFα

4.1.1 Main Results

In this chapter, we propose to study the "up to a fraction" relaxation of EF and PROP, when a mixture of divisible and indivisible goods are allocated. We show that an EFα allocation may not exist and a PROPα allocation always exists. Thus we would like to understand to what extent EFα needs to be relaxed and PROPα can be strengthened, namely EF$f(\alpha)$ and PROP$f(\alpha)$, so that a fair "up to a fraction" allocation exists.

In Section 4.3, we study the "up to a fraction" relaxation of EF, i.e., EFα and EF$f(\alpha)$. We first prove that $f(\alpha) = \Theta(n)\alpha$ is necessary and sufficient to satisfy EF by removing $f(\alpha)$ fraction of a good. We find that any EFM allocation is EF$n\alpha$, and thus an EF$n\alpha$ allocation always exists (by [27]). The guarantee of EFM cannot be improved even when agents have identical valuations. On the other hand, we prove that at least $\frac{n^2}{4(n-1)}\alpha$ fraction of the good has to be removed in order to satisfy EF, and thus our results are tight up to a constant. Besides, when agents have identical valuations, we show that a simple greedy algorithm ensures an EF$\frac{n^2}{4(n-1)}\alpha$ allocation, which exactly characterizes the extent to which EF$f(\alpha)$ can be guaranteed in this restricted case.

We then focus on the "up to a fraction" relaxation of PROP, namely, PROPα and PROP$f(\alpha)$, in Section 4.4. In contrast to EFα, EFM implies PROPα whose existence is thus guaranteed. Additionally, we design a simple polynomial-time algorithm to compute such an allocation. On the negative side, we find that a

PROP($\frac{n-1}{n} - \varepsilon$)$\alpha$ allocation does not always exist for any $\varepsilon > 0$ so that our bound is asymptotically the best possible.

The relations between the "up to a fraction" fairness notions and other well-known notions in the mixed goods setting are discussed in Section 4.5.

4.1.2 Related Work

The study of fair allocation is extensive (see, e.g., Amanatidis et al. [5], Brams and Taylor [42], Moulin [112], Robertson and Webb [122], Suksompong [127] for a survey). To capture fairness, various notions have been proposed for divisible and indivisible goods, including EF [68, 131] and PROP [126] for divisible goods; EF1 [46, 106] and PROP1 [58] for indivisible goods. There are also some notable fairness notions, e.g., *envy-freeness up to any good* (EFX) [76, 106] and *maximin share* (MMS) [46], etc.

Recently, a stream of literature has focused on the fair allocation problem with a mixture of divisible and indivisible goods (mixed goods) [27, 28, 31, 36, 116]. In particular, Bei et al. [27] initiated the fair division problem with mixed goods and proposed the fairness notion *envy-freeness for mixed goods* (EFM). Further, Bei et al. [28] and Kawase et al. [95] considered the fairness notions of MMS and *envy-freeness up to one good for mixed goods* (EF1M) in the mixed goods setting, respectively. Note that, the ratio of approximate MMS allocation obtained in [28] is a monotonically increasing function determined by how agents value the divisible goods relative to their MMS values. On the other hand, our proposed indivisible ratio is determined by how an agent values the divisible goods relative to her value of all goods. Furthermore, Li et al. [103] and Li et al. [105] examined EFM in conjunction with the issues of truthfulness and price of fairness, respectively. See a recent survey on the mixed fair division for more details [107].

In addition, several studies considered the interplay between fairness and efficiency for fair allocation [19, 21, 22, 69, 113, 135]. Specifically, EF, EF1, and EF1M are compatible with Pareto optimality (PO) (i.e., a criterion of efficiency) via the maximum Nash welfare allocation in the divisible, indivisible, and mixed goods settings, respectively [50, 124]. It is worth noting that EFM is incompatible with

PO while whether a weak version of EFM can be combined with PO is an open question in the mixed goods setting [27].

4.2 Preliminaries

Beyond the notations presented for the mixed goods setting in Chapter 2, we introduce additional notations required for this chapter in this section. In this chapter, we assume without loss of generality that agents' utilities are normalized to 1, i.e., $u_i(M \cup C) = 1$ for each $i \in [n]$.

As illustrated in the introduction, EF1 and PROP1 are rather weak in the mixed goods setting. In this chapter, we introduce new "up to a fraction" fairness notions with the help of *indivisibility ratio*.

Definition 4.1 (Indivisibility Ratio)**.** For each agent i, the *indivisibility ratio* α_i is defined as $\alpha_i := \frac{u_i(M)}{u_i(M)+u_i(C)}$.

For each agent i, α_i is the ratio between the utility for all *indivisible* goods and the utility for all goods. We point out that each agent has a *personalized* indivisibility ratio, allowing us to define the fairness with respect to each agent's perspective. Specifically, we introduce the following new fairness notions.

Definition 4.2 (EFα & PROPα)**.** An allocation $\mathcal{A}$ is called

- *envy-freeness up to α-fraction of one good (EFα)* if for any agents $i, j \in N$, there exists an indivisible good $g \in M_j$ such that $u_i(A_i) \geq u_i(A_j) - \alpha_i \cdot u_i(g)$.
- *proportionality up to α-fraction of one good (PROPα)* if for any agent $i \in N$, there exists an indivisible good $g \in M \backslash M_i$ such that $u_i(A_i) + \alpha_i \cdot u_i(g) \geq 1/n$.

It is easy to observe that when an agent has a higher utility for the cake, her indivisible ratio becomes smaller. This, in turn, implies that she is more likely to receive an allocation closer to EF/PROP under the EFα/PROPα criteria. One can easily check that when good is only the cake, EFα (resp., PROPα) reduces to EF (resp., PROP); when goods are all indivisible, EFα (resp., PROPα) reduces to EF1 (resp., PROP1). We can also observe that EFα implies PROPα.

As we will show later, an EFα allocation may not exist and a PROPα allocation always exists. For a better understanding of what EFα needs to be relaxed and PROPα can be strengthened, we next introduce the generalizations of EFα and PROPα.

Definition 4.3 (EF$f(\alpha)$ & PROP$f(\alpha)$)**.** An allocation $\mathcal{A}$ is

- *envy-freeness up to one $f(\alpha)$-fraction of good (EF$f(\alpha)$)* if for any agents $i, j \in N$, there exists an indivisible good $g \in M_j$ such that $u_i(A_i) \geq u_i(A_j) - f(\alpha_i) \cdot u_i(g)$.
- *proportionality up to one $f(\alpha)$-fraction of good (PROP$f(\alpha)$)* if for any agent $i \in N$, there exists an indivisible good $g \in M \backslash M_i$ such that $u_i(A_i) + f(\alpha_i) \cdot u_i(g) \geq 1/n$.

When $f(\alpha) = \alpha$, the above notions degenerate to EFα and PROPα. In this chapter, we focus on the linear function form $f(\alpha) = g(n) \cdot \alpha$, where $g(n)$ is a function of the number of agents. One can obtain stronger (resp., weaker) fairness requirements by making $g(n)$ smaller (resp., larger).

4.3 Envy-freeness up to a Fractional Good

In this section, we focus on envy-freeness up to a fractional good, i.e., EFα and EF$f(\alpha)$. We first present that for two agents, an EFα allocation can be found in polynomial time. Then, we proceed to consider the case with $n \geq 3$ agents and show that there does not exist EF$(\frac{n^2}{4(n-1)} - \varepsilon)\alpha$ allocations for any $\varepsilon > 0$. We then explore the best fairness guarantee under EF$f(\alpha)$. In particular, we find that an EF$n\alpha$ allocation always exists which is tight up to a constant factor. When agents have identical utility, we further show that $f(\alpha) = \frac{n^2}{4(n-1)}\alpha$ is the exact fraction we can guarantee for EF$f(\alpha)$.

4.3.1 Two Agents

In this subsection, we make use of the polynomial-time algorithm for finding an EFM allocation with two agents in [27] to provide the existence of EFα allocations for two agents.

Theorem 4.1. *When $n = 2$, an EFα allocation exists and can be found in polynomial time. Specifically, such allocation can be found in $O(m)$ time.*

Proof. The polynomial-time algorithm for finding an EFM allocation with two agents is also capable of finding an EFα allocation with two agents ([27]): "We begin with an EF1 allocation (M_1, M_2) of all indivisible goods. Assume without loss of generality that $u_1(M_1) \geq u_1(M_2)$. Next agent 1 adds the cake into M_1 and M_2 so that the two bundles are as close to each other as possible. Note that if $u_1(M_1) > u_1(M_2 \cup C)$, agent 1 would add all cake to M_2. If $u_1(M_1) \leq u_1(M_2 \cup C)$, agent 1 has a way to make the two bundles equal. We then give agent 2 her preferred bundle and leave to agent 1 the remaining bundle." It is easy to see that we only need to analyze the case when $u_1(M_1) > u_1(M_2 \cup C)$ holds and agent 1 gets $M_2 \cup C$. Since there exists some good g in M_1 such that $u_1(M_2) \geq u_1(M_1 \setminus \{g\})$, we have

$$\begin{aligned} u_1(M_2 \cup C) &= u_1(M_2) + (1 - \alpha_1) \\ &\geq u_1(M_1) - u_1(\{g\}) + (1 - \alpha_1)u_1(\{g\}) \\ &= u_1(M_1) - \alpha_1 \cdot u_1(\{g\}), \end{aligned}$$

which completes the proof.

Since the initial EF1 allocation can be found in $O(m)$ time by applying the envy-cycle elimination [106], and all the remaining procedure can also be done in $O(m)$ time. Thus, such allocation can be found in $O(m)$ time. □

4.3.2 General Number of Agents

We move on to consider the case of a general number of agents $n \geq 3$ in this part. When the resources to be allocated contain only divisible or indivisible goods, EFα allocations always exist. However, when the goods are mixed, we show that EFα allocations fail to exist even when there is only one homogeneous cake [1] and one indivisible good.

[1]We call a cake *homogeneous* if the utility over a subset of cake C depends only on the length of this subset, i.e., for each $i \in [n]$ and any $C' \subseteq C$, $u_i(C') = \frac{|C'|}{|C|} \cdot u_i(C)$, where $|C'|$ and $|C|$ represents the length of C' and C, respectively.

Theorem 4.2. *For $n \geq 3$ agents, an EFα allocation does not always exist. Specifically, for any $\varepsilon > 0$, an EF$(\frac{n^2}{4(n-1)} - \varepsilon)\alpha$ allocation does not always exist.*

Proof. The proof is derived from the following counterexample where we have n identical agents, one indivisible good g, and one homogeneous cake C.

	g	C	α
$u_i(\cdot), \forall i \in [n]$	$\frac{2}{n}$	$\frac{n-2}{n}$	$\frac{2}{n}$

Suppose the indivisible good g is allocated to agent n. Then there must exist one agent $i \in [n-1]$ such that $u_i(A_i) \leq \frac{n-2}{n(n-1)}$. For this agent i,

$$u_i(A_n) - \alpha \cdot u_i(o) = \frac{2}{n} - (\frac{2}{n})^2 = \frac{2(n-2)}{n^2}. \tag{4.1}$$

When $n \geq 3$, we have $\frac{2}{n} > \frac{1}{n-1}$, and thus $\frac{2(n-2)}{n^2} > \frac{n-2}{n(n-1)}$ which implies that agent i fails to achieve EFα.

The above analysis can be tighter, and we have

$$\begin{aligned} & u_i(A_n) - (\frac{n^2}{4(n-1)} - \varepsilon)\alpha \cdot u_i(o) \\ = \quad & \frac{n-2}{n(n-1)} + \frac{4\varepsilon}{n^2} > u_i(A_i), \end{aligned}$$

which implies agent i fails to achieve EF$(\frac{n^2}{4(n-1)} - \varepsilon)\alpha$. □

On the positive side, we will show in Section 4.5 that EFM implies EF$n\alpha$ (Theorem 4.7) which means that an EF$n\alpha$ allocation can be derived by using the polynomial algorithm for EFM in [27]. This also implies that when $n \geq 3$, the best fairness guarantee under EF$f(\alpha)$ notion would be EF$\Theta(n)\alpha$. Further, when all agents are identical, we show that the exact fairness guarantee under EF$f(\alpha)$ notion is EF$\frac{n^2}{4(n-1)}\alpha$.

Theorem 4.3. *When agents have identical utility functions, an EF$\frac{n^2}{4(n-1)}\alpha$ allocation always exist. Such allocation can be found in $O(mn^3)$ time.*

Proof. As described after the statement of Theorem 4.3, we adopt an algorithm which finds an EF1 allocation for indivisible goods first and then utilizes the water-filling procedure to allocate the cake, where the cake will be always allocated to the agents with the smallest utility.

Based on the form $f(\alpha) = g(n) \cdot \alpha$, we then prove that for an arbitrary integer $n \geq 3$, via this algorithm, each instance that admits a function $g(n)$ can be modified to an instance with only one indivisible good and a cake that also admits a function lower bounded by $g(n)$.

Under the instance I_1 that admits a function $g(n)$, we assume the envy of $g(n) \cdot \alpha$ occurs from agent i to agent j. Since when j is fixed, the smaller $u(i)$ can lead to a larger $g(n)$, we can assume the bundle i has the smallest utility among all bundles, which always receives divisible goods during the whole water-filling process and has exactly the utility at the water level (the utility of all agents receiving some cake).

Since the allocation is EF1 before allocating the cake, the difference between the utilities of bundle j and bundle i before allocating the divisible goods is upper bounded by some good $g \in M_j$. We then assume g' is the good with the largest utility in M_j, which is the corresponding o used in the condition of EF$f(\alpha)$. We can then compare to the instance I_2 where only one indivisible good with utility g' is given to agent j. In both instances, the g's corresponding to the o used in the condition of EF$f(\alpha)$ are the same but the initial difference between the utilities of agent j and i is larger under I_2, which is from a value weakly less than $u(g) \leq u(g')$ to exactly $u(g')$. Further, the divisible goods are required to be allocated to exactly $n-1$ agents from the beginning in I_2, which can makes a lower water level and a larger difference after terminating the water-filling procedure. With the decrease of α since there exists only one indivisible good g' in I_2, instance I_2 admits a larger $g(n)$ than that in I_1.

We can then focus on the case where there is only one indivisible good and a cake. We assume the value over the indivisible good is x and the value over the cake is $1-x$, where α here is exactly x. Without loss of generality, we assume $x \geq \frac{1-x}{n-1}$ otherwise we can directly reach a PROP allocation. We then need to find an x which minimizes the $g(n)$ in the inequality $x - g(n) \cdot x^2 \leq \frac{1-x}{n-1}$, which is the condition for EF$f(\alpha)$. By some calculus, the optimal x is exactly $\frac{2}{n}$, corresponding to the counterexample in Theorem 4.2.

For the time complexity, we can adopt the envy-cycle elimination [106] to find the initial EF1 allocation for indivisible goods, which can be done in $O(mn^3)$ time. For the water-filling procedure, since the size of the set of agents with the smallest utility can be increased at most n times and we can determine the allocated amount of the cake in each round in $O(n^2)$ time. Thus, the water-filling procedure can be done in $O(n^3)$ time. □

However, such an idea is not applied to the non-identical agents setting, where the presence of multiple indivisible goods may complicate the envy graph and prevent us from reducing it to a case with only one indivisible good.

4.4 Existence and Computation of PROP$f(\alpha)$

In this section, we focus on the proportionality up to a fractional good (i.e., PROPα and PROP$f(\alpha)$). We first prove by presenting a polynomial algorithm that a PROPα allocation always exists in the mixed good setting. Subsequently, we consider the PROP$f(\alpha)$ notion and show a lower bound of $f(\alpha)$, giving an asymptotically tight characterization for the existence of PROP$f(\alpha)$.

4.4.1 The Algorithm

The complete algorithm for finding a PROPα allocation is shown in Algorithm 2. Conceptually, Algorithm 2 performs the "moving-knife" procedure on indivisible goods and the cake separately through $n-1$ rounds (Lines 2-14). In each round, one agent is allocated with a bundle that yields a utility that achieves PROPα. The bundle is firstly filled by indivisible goods (Line 4) until there exists an indivisible good o such that after adding o to the bundle, some agent j will be satisfied with the bundle. Then, depending on whether we can make some agent satisfied by only adding some cake to the bundle, we execute either Case 1 (Lines 5-8) or Case 2 (Lines 9-14).

- Case 1: when the cake is not large enough to make any agent satisfied, we simply add the indivisible good o to the bundle and allocate it to agent j.
- Case 2: we first find out the minimum required piece of cake for each agent that will make her satisfied (Line 11). This step can be implemented through the RW model. Note that the condition of entering the second case ensures that at least one agent can be satisfied by adding some piece of the cake. we then find the optimal agent i^* that requires the minimum piece of cake among all agents and allocate the piece of cake together with the bundle of indivisible goods to her.

After allocating the bundles to $n-1$ agents, we give all the remaining goods to the last agent (Line 17). Assume, w.l.o.g., that agents $1, 2, \ldots, n$ receive their bundles in order.

We remark that when all goods are indivisible, our algorithm omits Case 2 and thus degenerates to the well-known bag-filling procedure. When the whole good is a divisible cake, the algorithm only executes Case 2 and then becomes the classical moving-knife algorithm. Though Algorithm 2 is a natural extension of the algorithms in the divisible goods setting and the indivisible goods settings, we claim that the analysis is non-trivial as shown in the next subsection.

4.4.2 Analysis

Our main result for the existence and computation of PROPα allocations is as follows.

Theorem 4.4. *For any number of agents, Algorithm 2 returns a PROPα allocation in polynomial time. Specifically, the PROPα allocation can be found in $O(n^2m^2)$ time.*

To prove Theorem 4.4, we will utilize some useful concepts and facts. Let k be the first agent that receives her bundle with only indivisible goods in Lines 5 to 8. In other words, agents $1, \ldots, k-1$ are assigned their bundles with some pieces of cake in Lines 9 to 14. Moreover, for distinction, we set $k \leftarrow n$ if all the first $n-1$ agents are assigned in Lines 9 to 14. The agent k plays an important role in bounding the ratio α_i for the agent after k. The relation is shown below.

Algorithm 2 Finding a PROPα allocation

Data: Agents N, indivisible goods M and cake C
Result: A PROPα allocation $(A_1, A_2, \ldots, A_n)$

1: $\hat{M} \leftarrow M, \hat{C} \leftarrow C$;
2: **while** $|N| \geq 2$ **do**
3: $B \leftarrow \emptyset$;
4: Add one indivisible good in $\hat{M}$ at a time to B until adding the next indivisible good o will cause $u_j(B \cup \{o\}) \geq 1/n - \alpha_j \cdot u_j(g)$ for some agent j and some good $g \in M \setminus (B \cup \{o\})$, or $M \setminus (B \cup \{o\}) = \emptyset$;
5: **if** $\forall i \in N$ and $g \in M \setminus B$, $u_i(B \cup \hat{C}) < 1/n - \alpha_i \cdot u_i(g)$ **then**
6: // Case 1: allocate with only indivisible goods;
7: $A_j \leftarrow B \cup \{o\}$;
8: $N \leftarrow N \setminus \{j\}, \hat{M} \leftarrow \hat{M} \setminus (B \cup \{o\})$;
9: **else**
10: // Case 2: allocate with cake;
11: Suppose now $\hat{C} = [a, b]$. For all $i \in N$, if $u_i(B \cup [a, b]) \geq 1/n - \alpha_i \cdot u_i(g)$, let x_i be the leftmost point such that $u_i(B \cup [a, x_i)) = 1/n - \alpha_i \cdot u_i(g)$ for some good $g \in M \setminus B$; otherwise, let $x_i = b$;
12: $i^* \leftarrow \arg\min_{i \in N} x_i$;
13: $A_{i^*} \leftarrow B \cup [a, x_{i^*})$;
14: $N \leftarrow N \setminus \{i^*\}, \hat{M} \leftarrow \hat{M} \setminus B, \hat{C} \leftarrow \hat{C} \setminus [a, x_{i^*})$;
15: **end if**
16: **end while**
17: Give all the remaining goods to the last agent;
18: **return** $(A_1, A_2, \ldots, A_n)$;

Proposition 4.1. *$\alpha_i \geq \frac{n-k}{n}$ for each agent $i > k$.*

Proof. Fix an agent $i > k$. According to the definition of k, each agent j in $\{1, \ldots, k-1\}$ will be assigned a bundle with some pieces of cake C_j. It is clear to see that agent i's valuation on each C_j is at most $1/n$ since otherwise agent i will be an agent that receives a bundle before agent k.

Next, we focus on the remaining cake. Note that if the condition in Line 5 is true, i.e., $u_i(B \cup \hat{C}) < 1/n - \alpha_i \cdot u_i(g)$ for all goods $g \in M \setminus B$, this also means that the remaining cake $\hat{C}$ worth at most $1/n$ to agent i.

Combined the above arguments, we have $u_i(C) \leq k/n$ which completes the proof. □

Let o_i be the indivisible good o as defined in Line 4 for the i-th iteration of the while-loop. For each agent j, we define $g_{ij} = \arg\max_{g \in (M \setminus M_i) \cup o_i} u_j(g)$. Intuitively, g_{ij} is the good that, when agent i conducts the "moving-knife" procedure (and finally obtains A_i), for agent j, the most valuable good besides the bundle A_i without o_i. The next proposition makes a connection between the above two definitions.

Proposition 4.2. *$u_j(g_{ij}) \geq u_j(o_p)$ for any agents i, j, p with $i \neq n$ and $o_p \neq o_{i-1}$.*

Proof. For the first case when $p > i$, because o_p exists in round p, we have $o_p \notin A_i$ and then verify the case from the definitions of g_{ij}.

When $p < i$, since $o_p \neq o_{i-1}$, o_p is not observed at round i and we can prove the statement by the definitions of g_{ij}.

When $p = i$, the statement obviously holds. □

We are ready to prove the following lemma.

Lemma 4.5. *Before the j-th iteration, the remaining goods are enough for j to achieve PROPα. Specifically, we have*

$$u_j(\hat{M} \cup \hat{C}) \geq (n - j + 1)(\frac{1}{n} - \alpha_j \cdot u_j(g_{jj})),$$

where $\hat{M} \cup \hat{C}$ represents the remaining goods just before the j-th iteration.

Proof. We first look at the case with $j \leq k$. By definition of k, each agent in $\{1, \ldots, k-1\}$ is assigned her bundle in Lines 9 to 14. In other words, from the viewpoint of j, the value of each bundle A_i with $i \in \{1, \ldots, j-1\}$ is less than $1/n - \alpha_j \cdot u_j(g) \leq 1/n$ for some $g \notin A_i$ since, otherwise, agent j will be assigned before agent i. This means that, agent j's value on the remaining goods is at least

$$1 - \frac{j-1}{n} = \frac{n-j+1}{n} \geq (n - j + 1)(\frac{1}{n} - \alpha_j \cdot u_j(g_{jj})).$$

We note that $k = n$ belongs to this case as the analysis only focus on the fact that each agent in $\{1, \ldots, k-1\}$ is assigned her bundle in Lines 9 to 14.

We next consider the case with $j > k$. Similar to the above case, we can show that agent j's value on each bundle A_i with $i \in \{1, \ldots, k-1\}$ is weakly less than $1/n - \alpha_j \cdot u_j(g_{ij})$. We then focus on the k-th assignment to the $(j-1)$-st assignment which can either be implemented in Lines 5 to 8 or in Lines 9 to 14. If the i-th assignment, where $k \leq i \leq j-1$, is executed by Lines 9 to 14, we can similarly obtain that $u_j(A_i) \leq 1/n - \alpha_j \cdot u_j(g_{ij})$. For the situation with Lines 5 to 8, we can also have $u_j(A_i) \leq 1/n - \alpha_j \cdot u_j(g_{ij}) + u_j(o_i)$. Now we are ready to compute agent j's value over the remaining goods. Our discussion is divided into two parts: (a) $\arg\max_{k \leq p \leq j-1} u_j(o_p) = \{o_{j-1}\}$ and $o_{j-1} \in A_j$; (b) other situations, i.e., either (i) $\arg\max_{k \leq p \leq j-1} u_j(o_p) \neq \{o_{j-1}\}$ or (ii) $\arg\max_{k \leq p \leq j-1} u_j(o_p) = \{o_{j-1}\}$ but $o_{j-1} \notin A_j$.

We first consider Part (b), we can get:

$$
\begin{aligned}
&u_j(\hat{M} \cup \hat{C}) \\
&= 1 - \sum_{1 \leq i \leq j-1} u_j(A_i) \\
&\geq 1 - \sum_{1 \leq i \leq k-1} \left(\frac{1}{n} - \alpha_j \cdot u_j(g_{ij})\right) - \sum_{k \leq i \leq j-1} \left(\frac{1}{n} - \alpha_j \cdot u_j(g_{ij}) + u_j(o_i)\right) \\
&= \frac{n-j+1}{n} + \sum_{1 \leq i \leq k-1} (\alpha_j \cdot u_j(g_{ij})) + \sum_{k \leq i \leq j-1} (\alpha_j \cdot u_j(g_{ij}) - u_j(o_i)) \\
&\geq \frac{n-j+1}{n} + \alpha_j \sum_{1 \leq i \leq k-1} \max_{k \leq p \leq j-1} u_j(o_p) + (\alpha_j - 1) \sum_{k \leq i \leq j-1} \max_{k \leq p \leq j-1} u_j(o_p)
\end{aligned}
$$

Here, for the second inequality, we can first observe that $u_j(g_{ij}) \geq \max_{k \leq p \leq j-1} u_j(o_p)$ for each $1 \leq i \leq k-1$ from Proposition 4.2. For the third term in this inequality, we achieve this by presenting the fact that $\alpha_j \cdot u_j(g_{ij}) - u_j(o_i) \geq (\alpha_j - 1) \max_{k \leq p \leq j-1} u_j(o_p)$ is satisfied for each $k \leq i \leq j-1$. We can utilize $u_j(o_i) \leq \max_{k \leq p \leq j-1} u_j(o_p)$ under the case that $u_j(g_{ij}) \geq \max_{k \leq p \leq j-1} u_j(o_p)$ and apply $u_j(o_i) \leq u_j(g_{ij})$ under the case that $u_j(g_{ij}) < \max_{k \leq p \leq j-1} u_j(o_p)$ to show this.

By rearranging the terms, this is equivalent to:

$$
\begin{aligned}
&\frac{n-j+1}{n} + (\alpha_j(k-1) + (\alpha_j - 1)(j-k)) \max_{k \le p \le j-1} u_j(o_p) \\
\ge &\frac{n-j+1}{n} + \left(\frac{(n-k)(k-1) - k(j-k)}{n} \right) \max_{k \le p \le j-1} u_j(o_p) \\
= &\frac{n-j+1}{n} + \left(\frac{-(n-k) + k(n-j)}{n} \right) \max_{k \le p \le j-1} u_j(o_p) \\
\ge &\frac{n-j+1}{n} - \left(\frac{(n-j+1)(n-k)}{n} \right) \max_{k \le p \le j-1} u_j(o_p) \\
\ge &(n-j+1) \left(\frac{1}{n} - \alpha_j \cdot \max_{k \le p \le j-1} u_j(o_p) \right) \\
\ge &(n-j+1) \left(\frac{1}{n} - \alpha_j \cdot u_j(g_{jj}) \right).
\end{aligned}
$$

The first and third inequalities are from Proposition 4.1, while the fourth inequality is from the assumption of Part (b). From the assumptions of Part (b), we can conclude that $\arg\max_{k \le p \le j-1} u_j(o_p)$ is not in the bundle A_j so $u_j(g_{jj}) \ge \max_{k \le p \le j-1} u_j(o_p)$.

We next consider Part (a). We first define a term r which is the largest value between k and $j-1$ such that $o_r \ne o_{j-1}$. Since $o_k \in A_k$ which cannot be o_{j-1}, such r must exist. We observe that each agent i from $r+1$ to $j-1$ gets her bundle with mixed goods in Lines 9 to 14 since $o_i = o_{j-1} \in A_j$. The analysis of Part (a) is similar to the one of Part (b) with some mild modifications.

$$
\begin{aligned}
&u_j(M \cup C) \\
&= 1 - \sum_{1 \le i \le j-1} u_j(A_i) \\
&\ge 1 - \sum_{1 \le i \le k-1} \left(\frac{1}{n} - \alpha_j u_j(g_{ij}) \right) - \sum_{k \le i \le r} \left(\frac{1}{n} - \alpha_j u_j(g_{ij}) + u_j(o_i) \right) - \sum_{r+1 \le i \le j-1} \left(\frac{1}{n} - \alpha_j u_j(g_{ij}) \right) \\
&\ge \frac{n-j+1}{n} + \sum_{\substack{1 \le i \le k-1 \\ or\ r+1 \le i \le k-1}} (\alpha_j \cdot u_j(g_{ij})) + \sum_{k \le i \le r} (\alpha_j \cdot u_j(g_{ij}) - u_j(o_i)) \\
&\ge \frac{n-j+1}{n} + \alpha_j \sum_{\substack{1 \le i \le k-1 \\ or\ r+1 \le i \le k-1}} \max_{k \le p \le r} u_j(o_p) + (\alpha_j - 1) \sum_{k \le i \le r} \max_{k \le p \le r} u_j(o_p) \\
&\ge (n-j+1) \left(\frac{1}{n} - \alpha_j \cdot u_j(g_{jj}) \right),
\end{aligned}
$$

where the inequalities can be derived based on the same reason as in Part (b). The proof of Lemma 4.5 is thus complete. □

We now turn our attention to the proof of Theorem 4.4.

Proof of Theorem 4.4. It is clear that all the goods are allocated after Line 17 of Algorithm 2. We next consider the correctness of the algorithm. By Lemma 4.5, we know that each iteration of the while-loop of Algorithm 2 is well-defined, which means that each agent in $\{1, \dots, n-1\}$ achieves her PROPα, either in Lines 5 to 8 or in Lines 9 to 14. For the last agent, Lemma 4.5 also implies that the PROPα of agent n can be satisfied.

Since each step in this algorithm can be executed in polynomial time and the total number of the while loop at Line 2 is exactly $n-1$, we can conclude the polynomial running time of the algorithm, which completes the proof. Specifically, during each execution of the while loop at Line 2, since we need to check the PROPα condition for each agent after adding each remaining indivisible good, the running time in this round is $O(m^2 n)$. All the remaining steps in each execution of the while loop at Line 2 can also be done in $O(mn)$ time. Thus, the total running time is $O(n^2 m^2)$. □

4.4.3 Impossibility Result

From Theorem 4.4, we know that a PROPα allocation always exists. In the following, we give a lower bound on $f(\alpha)$ such that PROP$f(\alpha)$ allocation is not guaranteed to exist.

Theorem 4.6. *For any $\varepsilon > 0$, a PROP$(\frac{n-1}{n} - \varepsilon)\alpha$ allocation does not always exist.*

Proof. Without loss of generality, we assume $\varepsilon \leq \frac{2}{5}$, otherwise we can choose the corresponding instance for $\varepsilon = \frac{2}{5}$ to prove this. Let $x = \frac{\varepsilon}{n-1}$. We consider the following instance with n identical agents, $n-1$ indivisible goods, and one cake.

	$\lvert M\rvert = n-1$	C	α
$u_i(\cdot), \forall i \in [n]$	$\frac{1-x}{n-1}, \forall g \in M$	x	$1-x$

There exists an agent i who obtains no indivisible goods. Thus $u_i(A_i) \leq u_i(C) = x$. For this agent and any good $g \in M$, we have

$$\begin{aligned}
& u_i(A_i) + (\frac{n-1}{n} - \varepsilon)\alpha_i \cdot u_i(g) \\
\leq\ & x + (\frac{1}{n} - x) \cdot (1-x)^2 \\
=\ & \frac{1}{n} - x(\frac{2}{n} - \frac{2n+1}{n}x + x^2) \\
=\ & \frac{1}{n} - x(\frac{2}{n} - \frac{2}{n}\frac{n+1/2}{n-1}\varepsilon + x^2) \\
<\ & \frac{1}{n}.
\end{aligned}$$

The last inequality is because $\frac{(n+\frac{1}{2})\varepsilon}{n-1} \leq 1$ from $n \geq 2$ and $\varepsilon \leq \frac{2}{5}$. Therefore, agent i fails to achieve PROP$(\frac{n-1}{n} - \varepsilon)\alpha$. □

This impossibility result with Theorem 4.4 also implies that we obtain an asymptotically tight characterization for the existence of PROP$f(\alpha)$.

Remark. We can observe that there exists a constant gap between Theorem 4.4 and Theorem 4.6. However, we cannot easily prove that our algorithm (Algorithm 2) can further ensure PROP$\frac{n-1}{n}\alpha$, since the proof of our algorithm heavily depends on the use of Proposition 4.1, which is not true if we enhance it to the statement that $\alpha_i \geq \frac{n-k}{n} \cdot \frac{n}{n-1} = \frac{n-k}{n-1}$ for each agent $i > k$.

4.5 Relation with Other Fairness Notions

We explore the relation among EF$f(\alpha)$, PROP$f(\alpha)$ and other notions (EFM and MMS) in the mixed goods setting. A summary of the results in this section is provided in Figure 4.1.

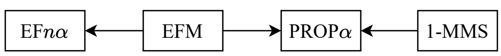


FIGURE 4.1: Relations among different fairness notions, where $X \to Y$ means that an allocation satisfying X must also satisfy Y.

4.5.1 Connections to EFM

We first discuss the relations of our "up to a fraction" fairness notions with EFM, proposed in [27], which is a natural combination of EF and EF1 in the mixed goods setting.

It is easy to verify that EFM does not imply EFα; recall the example in Section 4.1. However, EFM can imply a generalized version of EFα, as shown in the theorem below.

Theorem 4.7. *Any EFM allocation is EF$n\alpha$.*

Proof. Fix an agent 1 arbitrarily. If $\alpha_1 \geq \frac{1}{n}$, EF$n\alpha$ holds since EF1 is guaranteed by EFM. Otherwise, we have $u_1(C) > \frac{n-1}{n}$ and $u_1(M) < \frac{1}{n}$.

If $u_1(A_1) \geq \frac{1}{n}$, suppose $u_1(A_1) < u_1(A_i)$, EFM tells $C_i = \emptyset$ and thus $u_1(M_i) = u_1(A_i) > u_1(A_1) \geq \frac{1}{n}$, which contradicts with $u_1(M) < \frac{1}{n}$. Therefore, EF holds in this case.

We then consider the case when $u_1(A_1) < \frac{1}{n}$. Since $u_1(C) > \frac{n-1}{n}$ and any agent i containing some pieces of cake must have $u_1(C_i) \leq u_1(A_i) < u_1(A_1) < \frac{1}{n}$ by the definition of EFM, $u_1(C_i) > 0$ must hold for all $i \in N$ otherwise there must exist some divisible goods which cannot be allocated. Thus, agent 1 must not envy any other agent. □

On the other side, we show that EF$n\alpha$ is the best guarantee under EF$f(\alpha)$ that an EFM allocation ensures.

Theorem 4.8. *For any $\varepsilon > 0$, an EFM allocation may not be EF$(n-\varepsilon)\alpha$.*

Proof. Let $x = \frac{\varepsilon}{n}$. Consider the following instance with n agents, $\frac{1}{x}$ indivisible goods and one cake. Here, without loss of generality, we assume $\frac{1}{x}$ is an integer, otherwise we can choose another $\varepsilon' = \frac{1}{\lceil \frac{1}{\varepsilon} \rceil}$ and use the corresponding instance under this ε'. we do not normalize the total utility for better illustration here either.

	$\lvert M\rvert = \frac{1}{x}$	C	α
$u_i(\cdot), \forall i \in [n]$	$x, \forall g \in M$	$(n-1)(1-x)$	$\frac{1}{(1-x)n+x}$

By allocating M to agent 1 and dividing C equally among the rest of the agents, we have $u_i(A_1) = 1$ and $u_i(A_j) = 1 - x$ for any $i \in [n]$ and $j \in [n]\backslash\{1\}$. It's easy to check that this is an EFM allocation. However, for any good $g \in A_1$,

$$\begin{aligned} & u_2(A_1) - (n-\varepsilon)\alpha_2 \cdot u_2(g) \\ = \; & 1 - (n-\varepsilon) \cdot \frac{1}{(1-x)n + x} \cdot x \\ > \; & 1 - \frac{n-\varepsilon}{(1-x)n} \cdot x = u_2(A_2). \end{aligned}$$

This implies that the condition of EF$(n-\varepsilon)\alpha$ from agent 2 to agent 1 is not satisfied. ☐

We then consider the relation between PROPα and EFM.

Theorem 4.9. *An EFM allocation is PROPα.*

Proof. Fix an agent 1 arbitrarily. If $u_1(A_1) \geq \frac{1}{n}$, agent 1 achieves PROP (and thus PROPα). Otherwise, we assume $u_1(A_1) < \frac{1}{n}$ and pick the good $w \notin M_1$ such that $u_1(w) = \max_{g \notin M_1} u_1(g)$.

We assume an integer k such that $u_1(A_1) \geq u_1(A_j)$ for all $j \in [k]$ and $u_1(A_1) < u_1(A_j)$ for all $j > k$. By EFM, we have: $u_1(A_1) + u_1(w) \geq u_1(A_j)$ for all $j > k$. Adding all these inequalities for all $j \in [n]$, we have: $u_1(A_1) + \frac{n-k}{n} u_1(w) \geq \frac{1}{n}$.

If $u_1(M) \geq \frac{n-k}{n}$, the above inequality ensures PROPα. If not, we have $u_1(C) > \frac{k}{n}$. Since $\frac{1}{n} > u_1(A_1) \geq u_1(A_j)$ for all $j \in [k]$, there exists some cake allocated to agent $j > k$, a contradiction with the definition of EFM. ☐

This relation is also tight due to the following result.

Theorem 4.10. *For any $\varepsilon > 0$, an EFM allocation may not be PROP$(1-\varepsilon)\alpha$.*

Proof. Let $x = (n-1)\varepsilon$. Consider the following instance with n agents, $\frac{n-1}{x}$ indivisible goods, and one cake. Here, without loss of generality, we assume $\frac{1}{x}$ is an integer and $x \leq 1$, otherwise we can choose another $\varepsilon' = \frac{1}{\lceil \frac{1}{\min\{1/(n-1), \varepsilon\}} \rceil}$ and use the corresponding instance under this ε'. we do not normalize the total utility for better illustration here either.

	$\lvert M \rvert = \frac{n-1}{x}$	C	α
$u_i(\cdot), \forall i \in [n]$	$x, \forall g \in M$	$1-x$	$\frac{n-1}{n-x}$

By allocating C to agent 1 and dividing M equally among the rest of the agents, we have $u_i(A_1) = 1 - x$ and $u_i(A_j) = 1$ for any $i \in [n]$ and $j \in [n] \backslash \{1\}$. It is easy to check that this is an EFM allocation. However, for any good $g \in A \backslash A_1$,

$$\begin{aligned} & u_1(A_1) + (1-\varepsilon)\alpha_1 \cdot u_1(g) \\ = \ & 1 - x + (1-\varepsilon) \cdot \frac{n-1}{n-x} \cdot x \\ = \ & 1 - \left(\frac{1-x}{n-x} + \frac{n-1}{n-x}\varepsilon\right) \cdot x \\ = \ & 1 - \frac{x}{n-x} \\ < \ & 1 - \frac{x}{n} = \frac{n-x}{n}. \end{aligned}$$

This implies that the condition of PROP$(1-\varepsilon)\alpha$ for agent 1 is not satisfied. □

It is worth noting that Bei et al. [27] designed an algorithm for computing an EFM allocation. However, their algorithm utilizes the *perfect allocation oracle*, which is not in polynomial time when we have a heterogeneous cake, and consists of the intricate envy-graph maintenance and envy-cycle elimination subroutine. On the contrary, Chapter 2 runs in polynomial time and is simple to implement.

4.5.2 Connections to MMS

We also consider the relation of our "up to a fraction" fairness notions with MMS as defined in the following. Such fairness concept is first proposed by Budish [46], which is a natural extension of the proportionality from the divisible goods setting to the indivisible goods setting.

Definition 4.4 (β-MMS)**.** Let $\Pi_n(A)$ be the set of all n-partitions of A. The maximin share (MMS) of any agent $i \in N$ is defined as

$$\text{MMS}_i = \max_{\mathcal{P}=(P_1,P_2,\cdots,P_k)\in\Pi_n(A)} \min_{j\in N} u_i(P_j).$$

An allocation that reaches MMS_i is called an MMS-allocation of agent i. Given any $\beta \in [0, 1]$, allocation $\mathcal{A}$ is β-approximate MMS fair (β-MMS) if $u_i(A_i) \geq \beta \cdot \text{MMS}_i$ for every agent $i \in N$. When $\beta = 1$, we simply write MMS.

It is easy to see that when the goods are all divisible, MMS coincides with PROP. When the goods are all indivisible, MMS is strictly weaker than PROP but implies PROP1 [51]. Our next result is a generalization that encompasses these two extreme cases.

Theorem 4.11. *Any MMS allocation is PROPα.*

Proof. For an arbitrary MMS allocation $X = \{X_1, \ldots, X_n\}$, we assume this allocation X does not satisfy PROPα, i.e., there exists some $i \in [n]$ s.t. $u_i(X_i) < \frac{1}{n} - \alpha_i u_i(g)$ for every indivisible good $g \notin X_i$. Let $Y = \{Y_1, \ldots, Y_n\}$ as the MMS-allocation of agent i which minimizes the number of the bundles with the utility MMS_i from i's perspective. Without loss of generality, we assume Y_1 is the bundle with the largest fraction of the cake C among all bundles with the utility MMS_i and assume Y_n is the bundle with the largest utility under u_i.

Since X is an MMS allocation, $\frac{1}{n} > u_i(X_i) \geq \text{MMS}_i = u_i(Y_1)$, we then have $u_i(Y_n) > \frac{1}{n}$. If there exists a bundle Y_i with $u_i(Y_i) > \text{MMS}_i$ which contains some pieces of cake, we can move some of them to the bundle Y_1 which can make the allocation Y is no longer the required MMS-allocation as said above. Thus, the whole cake is shared by the bundles with utility MMS_i, which means that $u_i(Y_1) \geq u_i(C_1) \geq \frac{u_i(C)}{n}$ and Y_n contains only indivisible goods. Here, C_1 represents the cake in Y_1.

Because $u_i(Y_n) > \frac{1}{n} > u_i(X_i)$ and Y_n consists of no cake, there exists a good $g \in Y_n$ such that $g \notin X_i$. If $u_i(g) \geq \frac{1}{n}$, we have $u_i(X_i) + \alpha_i u_i(g) \geq u_i(Y_1) + \frac{u_i(M)}{n} \geq \frac{u_i(C)}{n} + \frac{u_i(M)}{n} = \frac{1}{n}$, which leads to a contradiction. If $u_i(g) \leq u_i(C_1)$ (so it is less than $u_i(Y_n)$), we can exchange the good g and an equivalent fraction of C_1 and then Y_n becomes a bundle with $u_i(Y_n) > \text{MMS}_i$ which contains some pieces of cake, which also leads to a contradiction with the definition of the allocation Y.

For the last case where $\frac{1}{n} > u_i(g) > u_i(C_1)$, we have $u_i(Y_1) + u_i(g) - u_i(C_1) \leq u_i(X_i) + u_i(g) - \frac{u_i(C)}{n} < u_i(X_i) + u_i(g) - u_i(g)u_i(C) = u_i(X_i) + u_i(g)u_i(M) < \frac{1}{n}$. The last inequality is from the fact that X_i violates the PROPα condition. With the fact that $u_i(Y_n) > \frac{1}{n}$, we can exchange the good g and C_1 to reach an MMS allocation with a smaller number of the bundles with utility MMS_i, which violates the definition of Y. This completes our proof. □

Recall that MMS allocations may not exist, however, an approximately MMS allocation may not be PROPα.

Theorem 4.12. *For any $\beta \in (0,1)$, a β-MMS allocation may not be PROPα.*

Proof. One example is to allocate the cake to n agents. Now an MMS allocation is also PROP (and PROPα since $\alpha_i = 0$ for all $i \in [n]$). Any approximately MMS allocation fails to satisfy MMS and thus PROPα.

Another example is allocating many small goods to n agents. For instance, $\frac{n}{x}$ goods with utility x for all agents. Here for any agent i, $\mathrm{MMS}_i = 1$ and $\alpha_i = 1$. By giving the first agent $\frac{1}{x} - 2$ goods and allocating the rest of the goods evenly to the rest of the agents (so that each agent obtains at least $\frac{1}{x}$ goods), this allocation is $(1 - 2x)$-MMS. However, it's obvious that this is not a PROP1 allocation. □

4.6 Conclusion

We study the fair allocation of a mixture of divisible and indivisible goods. We introduce the indivisibility ratio and fairness notions of envy-free and proportional up to a fractional good, which serves as a smooth connection between EF/PROP and EF1/PROP1. Our results exhibit the limit of the amount of the fractional item that we need to relax so that a fair allocation is guaranteed, which affirm the intuition that the more divisible items we have, the fairer allocations we can achieve.

There are some problems left open. For example, there is a constant gap between the upper and lower bounds of the fractional relaxation of EF. Our work also unveils intriguing possibilities for future research. One such avenue is proposing alternative relaxations of the ideal fairness principles to better capture the characteristics of mixed scenarios, such as the customized indivisibility ratio in our model.

Chapter 5

Fairness under Ordinal Uncertain Preferences

In this chapter, we turn our attention to the indivisible goods setting and investigate the fairness guarantee when the agents' preferences over items are ordinal and uncertain rather than cardinal. This chapter has been published in Li et al. [102].

5.1 Introduction

The problem of fairly allocating a set of scarce resources among multiple agents has been a central research topic in multi-agent systems and AI. One of the most prominent fairness notions in fair division is *proportionality*, which is also one of the first studied fairness concepts in the literature. The existence of a proportional allocation can always be guaranteed when the resources are divisible. However, with indivisible items, proportional allocations may fail to exist in some cases; a simple example is when there are only two agents and one item which both agents value positively.

A classic notion of proportionality is defined under the setting that each agent has a *cardinal* and *deterministic* valuation function over the items. In contrast, this chapter studies *ordinal* and *uncertain* preferences over items. Considering only ordinal preferences is a natural and appealing assumption due to several reasons. To name a few, first, it is often much easier to elicit ordinal preferences from the agents since it requires significantly less information. Second, some weaker fairness notions defined for ordinal preferences may permit more positive computational results for many problems. Finally, in certain applications, it's not available to get the cardinal preferences due to agents' limited knowledge on their valuations. In fact, the ordinal preferences have been widely studied in the resource allocation problems [18, 37, 47, 59, 60, 72, 97].

The assumption of uncertain preferences is also practical. In some settings the ordinal preferences may not be completely known due to a lack of information or a high cost to elicit a full preference list. Specifically, it may require a large number of pairwise comparisons among possible options to get the full preference order. It can also be difficult to even rank two close options without additional costly information [62]. In addition, according to the study of uncertain preferences in [17], an agent's preference may be a composition of several preference lists from other agents she represents, or a combination of preferences from different criteria, where their weights are not determined.

In summary, the problem studied in this chapter is to find a fair allocation for agents that have ordinal and uncertain preferences over items. In particular, the notion to define fairness under ordinal preferences is based on *stochastic dominance (SD)*, a standard way of comparing fractional allocations. We model the uncertain preferences following a similar way as in Aziz et al. [17] and Aziz et al. [14]. Specifically, we use tied sets to represent the items that the agents are not sure of their preferences towards and assume equal probabilities over each linear order extension of the preference order in the tied sets. This model of uncertainty is common for ordinal preferences. It has also been widely used in the matching literature [62, 119, 120]. To the best of our knowledge, our work presented in this chapter is the first one to consider fair allocations under such preference settings and provide algorithmic and complexity analysis for it.

5.1.1 Our results

In this chapter, we study how to find proportionally fair allocations when the agents have ordinal and uncertain preferences. Inspired by Aziz et al. [16] which studies Pareto optimal allocations under similar preference models, we study the computational complexity of the following four problems:

TABLE 5.1: Theorectical results

	Weak SD Proportionality
FAIRPROB	in P (Thm 5.1)
EXISTSPOSSIBLYFAIR	in P (Thm 5.2)
EXISTSCERTAINLYFAIR	NP-Hard (Thm 5.3) in P for constant n (Thm 5.5) in P for identical agents (Thm 5.4)
HIGHESTPROB	NP-hard (Thm 5.3) in P for constant n (Thm 5.5)

- FAIRPROB: What is the probability that a given allocation is fair under the specific fairness definition?
- EXISTSPOSSIBLYFAIR: Does there exist an allocation that is fair with non-zero probability?
- EXISTSCERTAINLYFAIR: Does there exist an allocation that is fair with probability one?
- HIGHESTPROB: How to find an allocation with the highest probability of being fair under the specific fairness definition?

We focus on one fairness notion introduced in Aziz et al. [13]: weak SD proportionality and SD proportionality. Other proportional notions are also discussed in Aziz et al. [13]. But they are shown to be equivalent to either of these two fairness notions. For each fairness notion, we present hardness results and efficient algorithms for the above four questions. Specifically, we show that for both fairness notions, FAIRPROB and EXISTSPOSSIBLYFAIR can be solved in polynomial time via dynamic programming and matching. EXISTSCERTAINLYFAIR and HIGHESTPROB are NP-hard. But we can provide polynomial-time algorithms for some special cases. For instance, if the number of agents is constant or the agents are all identical, we can efficiently find an allocation satisfying the corresponding condition. Due to the high similarities in the techniques used, we omit the results for SD proportionality in this thesis. Interested readers can refer to our conference version for more details. We summarize all the theoretical findings in Table 5.1.

Finally, noted the importance of HIGHESTPROB in many real-world scenarios, we propose several heuristic algorithms to find an allocation that is fair with a high probability. Experiments in both synthetic and real datasets are conducted to evaluate their performance and computation efficiency.

5.1.2 Related Work

In most existing literature, agents are assumed to have *cardinal* and *deterministic* valuations over items [35, 40, 43, 106].

Several papers have considered ordinal preferences and weaker fairness notions [13, 37, 41, 47, 59, 60, 72, 97, 125]. Aziz et al. [13] is the most related one. They used the stochastic dominance relation between fractional allocations to define various generalized notions of proportionality and envy-freeness and investigated the computational complexity of finding a fair assignment. They assume preferences are completely known and consider possibly indifferent preferences over items. In contrast, we assume that agents have strictly different preferences for different items but their preferences may not be completely known. We name the set of items with unknown preferences a tie. We model the preferences for items in a tie using uniformly distributed linear order extensions. It is worthwhile to note that assuming indifference for items with unknown preference can be regarded as an alternative way to model the uncertain preference. But we show that the fair allocation from such a modeling perspective can only be mapped to the certainly and possibly fair allocation in weak SD proportionality and SD proportionality, respectively in our work, after some small modifications.

Another stream of related literature considers uncertainties in agents' knowledge about their valuations [14–17, 108, 111]. The most related ones are Aziz et al. [17] and Aziz et al. [16], in which they study a Pareto optimal assignment problem. In their problem, agents also express ordinal and uncertain preferences. We adopt the way how they model the uncertain preferences but study a proportionally fair allocation problem. Lumet et al. [108] considered a different fair division uncertainty model in which each item could be in either good or bad state with certain probabilities. They developed algorithms to find ex-post fair allocations assuming independent states over items and correlated valuations over agents. Different from their paper, we assume the states of the items are correlated but the agent valuations are independent.

5.2 Preliminaries

We follow the indivisible goods setting described in Chapter 2. Given that we consider ordinal and uncertain preferences in this chapter, we provide the corresponding variants of the notations here.

Given the agents set $N = [n]$ and the indivisible goods set $M = \{g_1, g_2, \ldots, g_m\}$, each agent i has a complete and transitive strict preference order $\succ_i$ over M. The *ranking* of an item for an agent is defined as the sequence of the item in the agent's preference list. Specifically, an item's ranking in an agent's preference equals the number of items preferred by the agent plus one.

Uncertainty in Preferences. In this chapter, we allow agents to express uncertainty in their preferences in terms of "ties". Specifically, we adopt an uncertainty model considered in Aziz et al. [14] and Aziz et al. [17] (termed as a *compact indifference model* in their paper). In this model, each agent i is allowed to report a weak preference list. For instance, consider such a preference list $(S_{i,1}) \succ_i \ldots \succ_i (S_{i,k_i})$. Each $S_{i,j}$ is a tied set in the weak preference list that we call the jth *equivalent class* of agent i. The actual strict preference $\succ_i$ of agent i is then chosen uniformly at random from all linear order extensions of this weak order. The choices of the linear order extensions of different agents are independent. We denote by k_i the number of equivalent classes of agent i.

Example 5.1. Consider the following allocation problem with 2 agents and 4 items: agent 1 has preference $(a, b) \succ_1 (c, d)$, agent 2 has preference $(a) \succ_2 (b, c, d)$. In this instance, the strict preference of agent 1 may be $a \succ_1 b \succ_1 c \succ_1 d$ or $a \succ_1 b \succ_1 d \succ_1 c$ or $b \succ_1 a \succ_1 c \succ_1 d$ or $b \succ_1 a \succ_1 d \succ_1 c$, each with probability 1/4, while the strict preference of agent 2 may be $a \succ_2 b \succ_2 c \succ_2 d$ or $a \succ_2 b \succ_2 d \succ_2 c$, or the rest 4 preferences generated by the permutation of b, c, d, each with probability 1/6.

Proportional Fairness. When only ordinal preferences are available, Aziz et al. [13] defined several fairness notions that generalize proportionality to the ordinal setting. In this work we consider the Weak Stochastic Dominance (SD) proportionality.

Weak Stochastic Dominance (SD) proportionality: an allocation satisfies *weak SD proportionality* if for each agent i, there exists $1 \leq k \leq m$ such that agent i is allocated at least $\lfloor \frac{k}{n} + 1 \rfloor$ of her top k items.

The above definition of weak SD proportionality is a simplification of that in Aziz et al. [13] when only considering strict preferences. This is when comparing a uniform allocation with another allocation of an agent using the original SD preference definition in Aziz et al. [13], it reduces to the comparison between $\frac{k}{n}$ and the number of items allocated to this agent in her top k favorites for every $1 \leq k \leq m$ under strict preferences.

Example 5.1(continued). For our example above, when considering weak SD proportionality, one certainly fair allocation is to assign items b, c, d to agent 1 and item a to agent 2.

5.3 Weak SD Proportionality

In this section, we consider the weak SD proportionality. First we show that given an allocation, one can efficiently compute the probability that this allocation is weak SD proportional.

Theorem 5.1. FAIRPROB *with regard to weak SD proportionality can be solved in polynomial time. Specifically, this can be solved in $O(m^3K)$ time.*

Proof. Because the uncertainty in each agent's preference is independent, it suffices to show that the probability that the fairness condition is met for each agent $i \in N$ can be computed in polynomial time. In the following, we use dynamic programming to calculate this probability.

For one specific agent $i \in N$, there are $K := k_i$ equivalent classes and the j-th equivalent class has $s_j := |S_{i,j}|$ items. First, we calculate the number of items owned by i in the j-th equivalent class of i and we denote it by $r_j \leq s_j$.

The key to this dynamic programming algorithm is to identify the subproblem structure. We use $P_{\mathsf{num},\mathsf{tot},\mathsf{dis}}$ to represent the probability that there are tot items owned by agent i arranged in the top num items with a fair verification parameter

Algorithm 3 FairProb Algorithm

Require:
m: number of items
n: number of agents
K: number of equivalent classes of agent i
$s_{1,\dots,K}$: number of items in each equivalent classes
$r_{1,\dots,K}$: number of items owned by agent i in each equivalent classes

1: $P_{\mathsf{num},\mathsf{tot},\mathsf{dis}} \leftarrow 0 \quad \forall 0 \leq \mathsf{num}, \mathsf{tot} \leq m, 0 \leq \mathsf{dis} \leq \lfloor \frac{m}{n} + 1 \rfloor$
2: $P_{0,0,1} \leftarrow 1$
3: **for** $\mathsf{num} \leftarrow 0$ to $m - 1$ **do**
4:     **for** $\mathsf{tot} \leftarrow 0$ to num **do**
5:         **for** $\mathsf{dis} \leftarrow 0$ to $\lfloor \frac{m}{n} + 1 \rfloor$ **do**
6:             **if** $P_{\mathsf{num},\mathsf{tot},\mathsf{dis}} \neq 0$ **then**
7:                 Let sta be the minimum integer such that $\sum_{j \in [\mathsf{sta}]} s_j \geq \mathsf{num} + 1$.
8:                 $\mathsf{PrevP} \leftarrow P_{\mathsf{num},\mathsf{tot},\mathsf{dis}}$
9:                 $p \leftarrow \frac{\sum_{j \in [sta]} r_j - \mathsf{tot}}{\sum_{j \in [sta]} s_j - \mathsf{num}}$
10:                 **if** $\mathsf{dis} = 0$ **then**
11:                     $P_{\mathsf{num}+1,\mathsf{tot}+1,\mathsf{dis}} \mathrel{+}= \mathsf{PrevP} \cdot p$
12:                     $P_{\mathsf{num}+1,\mathsf{tot},\mathsf{dis}} \mathrel{+}= \mathsf{PrevP} \cdot (1 - p)$
13:                 **else**
14:                     Let id be 1 if $\mathsf{num} + 1$ is a multiple of n and 0 otherwise.
15:                     $P_{\mathsf{num}+1,\mathsf{tot}+1,\max\{0,\mathsf{dis}+\mathsf{id}-1\}} \mathrel{+}= \mathsf{PrevP} \cdot p$
16:                     $P_{\mathsf{num}+1,\mathsf{tot},\mathsf{dis}+\mathsf{id}} \mathrel{+}= \mathsf{PrevP} \cdot (1 - p)$
17:                 **end if**
18:             **end if**
19:         **end for**
20:     **end for**
21: **end for**
22: **return** $P_{m,\sum_{j \in [K]} r_j,0}$

dis defined as follows:

$$\mathsf{dis} = \begin{cases} 0, & \text{if it's tranferred from the state with } \mathsf{dis}' = 0 \\ \max\left\{0, \left\lfloor \frac{\mathsf{num}}{n} + 1 \right\rfloor - \mathsf{tot}\right\}, & \text{otherwise} \end{cases}$$

Intuitively, dis shows the distance to weak SD proportionality and once it reaches 0, it will keep at 0.

The complete algorithm is shown in Algorithm 3. The key steps in this algorithm are the state transfers in Lines 3-21. We let sta represent the equivalent class that the $(\mathsf{num}+1)$-ranked item belongs to, and let p represent the probability of the item with ranking $(\mathsf{num}+1)$ owned by i under the present state in Lines 7-9.

We consider the specific transfers in two cases. Lines 11-12 solve the first case with $\mathsf{dis}=0$ where we have already reached fairness and the remaining is to arrange the order of the rest items. Lines 14-16 solve the second case with $\mathsf{dis}>0$ where dis has never met 0 before and we need to update it based on the definition. In both these two cases, we need to update the state based on the situation of whether the next item is owned by agent i. Specifically, when $\mathsf{dis}=0$, if the next item is owned by agent i, there are $\mathsf{tot}+1$ items owned by i arranged in the top $\mathsf{num}+1$ items, so we transfer the state to $P_{\mathsf{num}+1,\mathsf{tot}+1,\mathsf{dis}}$ in Line 11, otherwise only tot items in i's bundle are arranged in the top $\mathsf{num}+1$ items, which corresponds to the state $P_{\mathsf{num}+1,\mathsf{tot},\mathsf{dis}}$ in Line 12. For the case of $\mathsf{dis}>0$, on top of the change of tot, we also use id to record the change of the term $\lfloor\frac{\mathsf{num}}{n}+1\rfloor$ in the above definition of dis in Line 14. Then, we transfer the state to $P_{\mathsf{num}+1,\mathsf{tot}+1,\max\{0,\mathsf{dis}+\mathsf{id}-1\}}$ in Line 15 if the next item is owned by i, or $P_{\mathsf{num}+1,\mathsf{tot},\mathsf{dis}+\mathsf{id}}$ in Line 16 otherwise.

Finally, Line 22 returns the probability that the fairness condition is met for agent i with her all owned items.

For the time complexity, since each execution of the for loop can be done in $O(K)$ time and there are totally $O(m^3)$ rounds of the for loop, the total time complexity is $O(m^3K)$. □

We further show that ExistsPossiblyFair can also be solved in polynomial time based on Theorem 7 in Aziz et al. [13].

Theorem 5.2. ExistsPossiblyFair *with regard to weak SD proportionality can be solved in polynomial time. Specifically, this can be solved in $O(n^3+mn)$ time.*

Proof. Theorem 7 in Aziz et al. [13] showed that when agents have strict and deterministic preferences, a weak SD proportional allocation exists if and only if (1) $m=n$ and no item is least preferred by all agents, or (2) $m>n$. Furthermore, when such allocation exists, it can also be found in polynomial time.

This result allows us to derive a simple algorithm for ExistsPossiblyFair. First, when $m>n$, we can simply find an arbitrary set of linear extensions and compute a corresponding weak SD proportional allocation. Since an arbitrary set of linear extensions can be found in $O(mn)$ time and the corresponding weak SD proportional allocation can be found in $O(n^2)$ time. Thus, the time complexity in this case is $O(mn + n^2)$.

When $m = n$, we need to find a set of linear extensions in which no item is least preferred by all agents, then compute a weak SD proportional allocation with regard to this preference profile. To find such a preference profile, we can construct a bipartite graph between the agents and the items, such that an edge exists if and only if the item is not least preferred by the agent in some preference profiles. If there exists a perfect matching in this bipartite graph, this perfect matching corresponds to an allocation that is weak SD proportional with non-zero probability. If the graph does not have a perfect matching, because each agent has at least $n - 1$ edges, by Hall's marriage theorem [80], there must exist one item which is least preferred by all agents in all preference profiles. Thus, there is no allocation with non-zero fair probability. Since a perfect matching can be found in $O(n^3)$ time by applying the Hungarian algorithm [64], the time complexity in this case is $O(n^3)$. □

Next we turn to ExistsCertainlyFair problem. Unlike ExistsPossiblyFair, we show that it is NP-hard to determine whether there exists an allocation that is weak SD proportional with probability one. To prove this hardness result, we reduce from a known NP-hard problem denote as (2,2)-E3-SAT [33].

(2,2)-E3-SAT: Given a boolean formula in conjunctive normal form in which each clause has three literals and each variable occurs exactly twice positive and twice negative, decide whether this boolean formula is satisfiable.

Theorem 5.3. ExistsCertainlyFair *and* HighestProb *with regard to weak SD proportionality are NP-hard.*

Proof. We reduce from (2,2)-E3-SAT to ExistsCertainlyFair, which can also imply NP-hardness for HighestProb. Considering a (2,2)-E3-SAT instance F with s variables $X = \{x_1, \ldots, x_s\}$ and t clauses which satisfies $4s = 3t$. Based on F we construct a problem instance of ExistsCertainlyFair with $6s$ agents and $12s$ items.

The agents are divided into two sets. The first is a set of $2s$ agents $A = \{A_1, \ldots, A_{2s}\}$ where A_{2k-1} corresponds to the true valuation of x_k and A_{2k} corresponds to the false valuation of x_k for each $k \in [s]$. The second set is the set of the remaining $4s$ dummy agents $A' = \{A'_1, \ldots, A'_{4s}\}$.

The $12s$ items are divided into five sets as follows:

- A set of t items $C = \{C_1, \ldots, C_t\}$, where C_i corresponds to the i-th clause in (2,2)-E3-SAT;
- A set of $t/2$ items $T = \{T_1, \ldots, T_{t/2}\}$, given to the agents corresponding to the correct value of variables together with the items in C;
- A set of s items $S = \{S_1, \ldots, S_s\}$, compensated for the agents representing the wrong value of variables;
- A set of $6s$ dummy items $B = \{B_1, \ldots, B_{6s}\}$, where each 2 items are bound to $3s$ agents in A';
- A set of $3s$ dummy items $Q = \{Q_1, \ldots, Q_{3s}\}$, where each 3 items are bound to the remaining s agents in A'.

Next, we construct the preference lists for agents. For each agent $A_i \in A$, we denote R_i as a set of items such that: for each $k \in [s]$, R_{2k-1} and R_{2k} consist of the two items in C corresponding to the clauses containing the positive x_k and the negative x_k respectively. The preference of each agent A_i in A is: $S_{\lceil i/2 \rceil} \succ (R_i, T, B) \succ$ (others). The preference of each dummy agent A'_i in A' is: $(B) \succ$ (others).

If the (2,2)-E3-SAT instance has a satisfying assignment, we can construct an allocation as follows:

- If variable x_i is TRUE, we assign S_i to A_{2i}, otherwise we assign S_i to A_{2i-1};
- For the j-th clause, suppose the first true term in it is about x_i. Then, if this term is positive x_i, we assign C_j to A_{2i-1}, otherwise we assign C_j to A_{2i}.
- For agents in A who do not receive any S_i, we assign items in T to them so that these agents can each receive exactly two items. We know this can be satisfied because $2s = t + t/2$.
- Finally, we assign two items in B to each of the first $3s$ agents in A' and assign three items in Q to each of the remaining s agents in A' arbitrarily.

For the agents in A who get an item in S, their fairness conditions are met because they get their unique most preferred item respectively. Each of the remaining agents in A gets exactly two items that are not least preferred by them, so their fair conditions are also met. Finally, agents in A' also meet the fair conditions because either they receive two items that they do not prefer the least, or they receive three items. Thus, this allocation must satisfy weak SD proportionality.

On the other hand, assume there is an allocation that is weak SD proportional with probability one, we can construct a satisfying assignment for the (2,2)-E3-SAT instance. First, the agents in A who do not receive their most preferred item must get at least two items to meet the fair condition. Because there are at most s agents that can receive items in S, at least s agents in A need at least two items. Next, for the agents in A', they can meet the fairness condition if and only if they receive either two items in B or three items. Because of the limit of the number of items in B, at most $3s$ agents in B can meet the fair condition with only two items.

We can consider the most optimistic situation: s agents in A each get their most preferred item in S and the remaining s agents in A get exactly two items in their respective second equivalent class but not in B; $3s$ agents in A' each get exactly two items in B while the remaining s agents in A' each get three items. The number of items needed in total is $12s$, which means that this situation is necessary.

Therefore, for any $k \in [s]$, exactly one agent between A_{2k-1} and A_{2k} gets the corresponding S_k while the other one needs to get two items from $C \cup T$, whose size is $t/2 + t = 2s$. This means each item in C must be chosen by the agents in A, and this will lead to a satisfying assignment for (2,2)-E3-SAT with such value scheme: for each pair A_{2k-1} and A_{2k} with $k \in [s]$, we set $x_k = \mathsf{FALSE}$ if A_{2k-1} gets S_k otherwise we set $x_k = \mathsf{TRUE}$.

□

Next we present several positive results and short discussions on solving ExistsCertainlyFair and HighestProb for special cases.

Theorem 5.4. ExistsCertainlyFair *with regard to weak SD proportionality can be solved in polynomial time when all agents have identical preferences. Specifically, this can be solved in $O(m^2)$ time.*

When all agents have identical preferences, we can derive an optimal greedy algorithm that assigns items to agents such that each agent receives consecutive items in a preference order from the most to the least preferred items. This demonstrates the polynomial-time solvability of the ExistsCertainlyFair. In this analysis, the assumption of identical preferences is critical since it allows us to sort all items before the allocation, which is a crucial step in the follow-up analysis.

Proof. Firstly, we simplify some notions when agents have identical preferences: we use K to represent the number of equivalent classes and use S_j to represent the j-th equivalent class. Then we re-number all items with indices 1 to n such that $S_1 = \{1, \ldots, |S_1|\}$ and $S_j = \{\sum_{k=1}^{j-1} |S_k| + 1, \ldots, \sum_{k=1}^{j} |S_k|\}$ for each $1 < j \leq K$. This means for all agents, smaller-indexed items are always more preferred.

We consider the following greedy algorithm. For each agent from 1 to n, we consider the remaining items in increasing order of indices, and allocate them to this agent one by one, until the current allocation met the fairness condition for this agent with probability 1. We repeat this process until all items are fully allocated or all agents have received their desired bundles of items. In this algorithm, since we only need to enumerate the items and check the fairness condition after adding each item, the total time complexity is $O(m^2)$.

In the following, we show that if a certainly fair allocation M exists in a problem instance, the above greedy algorithm can always return such a certainly fair allocation.

Next, we perform a three-step transformation on the given certainly fair allocation M:

- For each agent i, if removing the item with largest index in her bundle can still keep her fairness condition met, we will remove that item from i's bundle. Repeat this process until no items can be removed anymore. After this step, we denote sta_i as the equivalent class the worst item in i's bundle belongs to. Then, for any agent i, she could not have her fairness condition met after throwing the present worst item and all owned items are always least preferred in their respective equivalent classes in the worst case, so there must exist $\sum_{j=1}^{\mathsf{sta}_i - 1} |S_j| < k_i \leq \sum_{j=1}^{\mathsf{sta}_i} |S_j|$ such that agent i is allocated at least $\lfloor \frac{k_i}{n} + 1 \rfloor$ of her top k_i items in the worst case mentioned above.

- Next, we find any agent who has item o that has a larger index than some thrown away item o', and we replace o with o'. Again repeat this process until no such changes can be made. We know M is still certainly fair. After this step, the present allocation must contain exactly all the items in $[m']$ for some $m' \leq m$.

- Finally, if there exists agent i and j such that the worst item in i's bundle has a larger index than the worst items in j's bundle, and agent i also has an item o_i with a smaller index than some item o_j belonging to agent j, we will be able to swap o_i and o_j while still keeping the fairness conditions for both agents. This obviously holds for agent j. For agent i, this claim is true because o_j is still not worse than the worst item in i's bundle and it still holds that agent i is allocated at least $\lfloor \frac{k_i}{n} + 1 \rfloor$ of her top k_i items in the worst case for that k_i. Repeat this until no more exchanges can be made. After this step, we know the allocation must still be certainly fair and it is a consecutive partition of $[m']$ for some $m' \leq m$.

Finally, it is easy to see if there exists a consecutive partition that is certainly fair, the allocation computed by our greedy algorithm must be also certainly fair. □

Theorem 5.5. ExistsCertainlyFair *and* HighestProb *with regard to weak SD proportionality can be solved in polynomial time when the number of agents is constant. Specifically, this can be solved in* $O(n^{4n^2+n+1} \cdot m^3)$ *time.*

When the number of agents n is a constant, we can derive the following exact algorithm. First, we enumerate the number of items assigned to each agent from each equivalent class. Next, we use a perfect matching algorithm to find a corresponding allocation scheme and apply FairProb to calculate the fair probability. The total number of enumerations is in the order of $O(n^{4n^2+n+1})$ which is a constant. However, this is no longer true if n is a super-constant. Therefore, the constant assumption is an important assumption to get a polynomial-time solvable algorithm.

Proof. Let $m = qn + r$, where q, r are integers and $0 \leq r < n$.

First, we observe that any agent who gets $q+1$ items is always weak SD proportional with probability one. Thus we can assume that no agent gets more than $q+1$ items in the allocation with the highest fair probability. This also implies that every agent gets at least $q_m = q - (n-1-r)$ items, because otherwise the total number of allocated items will be less than $q-(n-1-r)+(n-1)(q+1) = qn+r = m$.

If there exists some agent i such that $|S_{i,1}| < n \cdot q_m$, we can choose $\min\{|S_{i,1}|, q_m\}$ items from $S_{i,1}$ to be allocated to agent i, which guarantees the fair condition for agent i to be met with probability one. Next, we assign all other agents $q+1$ arbitrary items each and give the remaining items back to i. It is easy to check that this allocation is weak SD proportional with probability one.

Now we consider the case where $|S_{i,1}| \geq n \cdot q_m$ holds for every agent $i \in N$. This means that the total number of items not in the first equivalent class for any agent is no more than $n(n-1-r)+r$.

For each agent i, if there exists an integer $1 \leq j < |k_i|$, such that $1 + \sum_{t=1}^{j-1} |S_{i,t}| \geq (q'-1)n$ and $\sum_{t=1}^{j+1} |S_{i,t}| < q'n$ for some $q' \leq q+1$, then the items in the j-th and $(j+1)$-th equivalent classes can be merged into one equivalent class without affecting the fair probability of any allocation, because all items in these two equivalent classes will always have the ranking in the range between $(q'-1)n$ and $q'n-1$. We don't really care about the specific ranking in this range because we can simplify the definition of weak SD proportionality to only consider whether there exists $k \in \{n-1, 2n-1, \ldots, qn-1, m\}$ such that agent i is allocated at least $\lfloor \frac{k}{n} + 1 \rfloor$ of her top k items. We repeatedly merge such adjacent equivalent classes until none can be found. By the end of this procedure, the total number of equivalent classes in each agent's preference list is no more than $1 + 2 \cdot (n-1-r+1) = 2(n-r)+1$.

Next, we enumerate the number of items assigned to each agent i from each equivalent class. More specifically, for each agent i, we first enumerate the total number of items in her bundle and then the number $C_{i,j}$ of items in each $S_{i,j}$ assigned to agent i. For each set of numbers $\{C_{i,j}\}$, we use the following perfect matching algorithm to find candidate allocations.

- We construct the bipartite graph $G = (A \cup B, E)$ where A has $C_{i,j}$ duplicate vertices representing the positions for items in $S_{i,j}$ for each agent $i \in [n]$ and each equivalent class $j \in [k_i]$, and B contains m vertices each representing an item. For each duplicate vertex for $S_{i,j}$ in A and each vertex in B corresponding to the item in $S_{i,j}$, there is an edge between them in E;

- If there exists a perfect matching between A and B (which also implies $|A| = |B|$), we pick the allocation where each item whose corresponding vertex is matched with one duplicate vertex for one $S_{i,j}$ is assigned to agent i and it meets our requirements.

If such allocation exists, its fair probability can be computed using the algorithm designed for FairProb. Finally, we return the allocation with the highest fair probability.

Finally we analyze the time complexity of the above algorithm. The first step takes $O(nm)$ to enumerate each agent and each item. Merging equivalent classes also takes $O(nm)$ time. Next, for the enumeration step, because each number $C_{i,j} \leq n(n-1-r)+r$ when $j > 1$ and the total number of items in any bundle is between $q-(n-1-r)$ and $q+1$, the total number of possibilities is $O((n(n-1-r)+r+1)^{2(n-r)\cdot n} \cdot (n-1-r+1+1)^n) = O(n^{4n^2+n+1})$. Finally, deciding the existence of an allocation and calculating its fair probability takes $O(m^3)$ time. Thus, the overall time complexity of this algorithm is $O(n^{4n^2+n+1} \cdot m^3)$. □

5.4 Experiments

In this section, we focus on HighestProb with respect to weak SD proportionality given the importance of finding allocations with a high fair probability. Since we have shown this problem is NP-hard, we will only present several heuristic algorithms for it. The performance and computational efficiency are evaluated on both synthetic and real datasets.

5.4.1 Algorithms

We design and evaluate four algorithms for HighestProb.

- **Baseline (B)**: We assign each item to a random agent to get a random allocation. We repeat this process B times to get B random allocations and choose the one with the largest fair probability. We set $B = 4m$ in the experiment. We also tested our algorithm with B increased to $8m$, and did not observe any noticeable improvement on the algorithm's performance.

- **LocalSearch (LS)**: Two allocations are called neighbors if one allocation can be derived from the other by moving one item from some agent to another agent. In this Local Search algorithm, we start from a random allocation and iteratively move to a neighbor allocation that has a higher fair probability, until a locally optimal allocation is reached. We repeat the above process L times, each time with a new random initial allocation, and choose the final allocation with the highest fair probability. We set $L = n$ in the experiment. We also tested our algorithm with L increased to $2n$, and did not observe any noticeable improvement on the performance.

- **Matching (M)**: Since in a weak SD proportional allocation, every agent needs to receive $\lfloor \frac{r}{n} \rfloor + 1$ items with ranking not worse than r for some r. We set the value of an agent getting an item with ranking r as $\frac{1}{\lfloor \frac{r}{n} \rfloor + 1}$. For each agent i and item o, let $\mathsf{avg}(i, o)$ be the average value of i getting o.

  This algorithm runs for $\frac{m}{n} + 1$ rounds and assigns at most one item to each agent each round. In round k, let R_i be the total value that agent i has received in previous rounds, then for each item o, we set the value of agent i getting item o in this round as $(1 - R_i) \cdot \mathsf{avg}(i, o)$. In addition, we remove all edges whose values are less than a certain threshold L. Then we find a maximum weight matching between agents and items and assign the item to the matched agent. Finally, we update R_i for each agent and remove all agents who receive a total value larger than another threshold U from consideration in future rounds. After finishing $\frac{m}{n} + 1$ rounds, we allocate the remaining items to the agents with less than $\frac{m}{n} + 1$ items arbitrarily.

  In our experiment, we enumerate 5 different lower bound L from $\frac{1}{\lfloor \frac{m}{n} \rfloor + 1}$ to $\frac{1}{\lfloor \frac{m}{n} \rfloor + 1} + 0.1$ equidistant and 11 different upper bound U from 1 down to 0.7 equidistant. We run the algorithm for each pair of L and U, and choose the allocation with the largest fair probability among all solutions. We also test the algorithm with L higher than $\frac{1}{\lfloor \frac{m}{n} \rfloor + 1} + 0.1$ or with U lower than 0.7 and do not observe significant performance improvement.

- **Greedy (G)**: For each agent i and item o, we adopt notion $\mathsf{avg}(i, o)$ from the previous Matching algorithm to be the average value of i getting o. Then for each agent i and a set S of items, when calculating the R_i in the above Matching algorithm, we can find that it's always equal to $1 - \prod_{o \in S}(1 - \mathsf{avg}(i, o))$ no matter in what order the items in S are adding to agent i's bundle. Thus, we let $f_i(S) = \min(U, 1 - \prod_{o \in S}(1 - \mathsf{avg}(i,o)))$ denote the value of this agent receiving S, where U is a threshold parameter. One can check that f_i is a submodular function, and our question becomes a general submodular welfare maximization problem. We then use a greedy algorithm from Lehmann et al. [98] which always outputs a 2-approximation solution for the submodular welfare maximization problem.

In the experiment, we run this algorithm with 11 different values of U from 1 down to 0.7, and choose the allocation with the largest fair probability. We also tested our algorithm with the value of U less than 0.7, and found it did not improve the algorithm's performance.

5.4.2 Datasets

We test the above four algorithms on both synthetic and real datasets.

- **Synthetic dataset**: We enumerate n from 2 to 20 and m from n to $5n$. For each $p \in \{0.02, 0.03, 0.04, 0.05, 0.06\}$, we create 30 datasets in the following way: For each agent we create a random permutation of all items. For each pair of adjacent items in the permutation, we separate them into two equivalent classes with probability p. We choose p to be in this range because when p becomes larger, all the four algorithms can easily find an almost certainly fair assignment and there is no significant difference in their performance.
- **Real dataset**: We use the Preflib database from Mattei and Walsh [110], which is an online database of real-world preference profiles to test our algorithms. We select 11 datasets from two categories in this database:
  - *Matching Data (MD-00002)*: This category contains bidding preferences of reviewers over a subset of papers at Computer Science conferences. Each preference is an incomplete list with ties. We convert each preference into a complete list with ties by adding the remaining items as the last equivalent class. This category has 3 datasets corresponding to 3 different conferences.
  - *Matching Data (MD-00003)*: This category contains bids of students over a set of projects for student/project allocations at a university. It

has 8 datasets in total, all with complete preferences with ties, with 31-51 students and 56-155 projects.

5.4.3 Results and Discussions

Synthetic dataset. We measure the performance of each algorithm by the average fair probability across various tested datasets. Figure 5.1 shows the average fair probability generated by each algorithm for each n from 2 to 20. Each average fair probability is taken over multiple datasets with different m and p. Figure 5.2 presents the average running time of each algorithm.

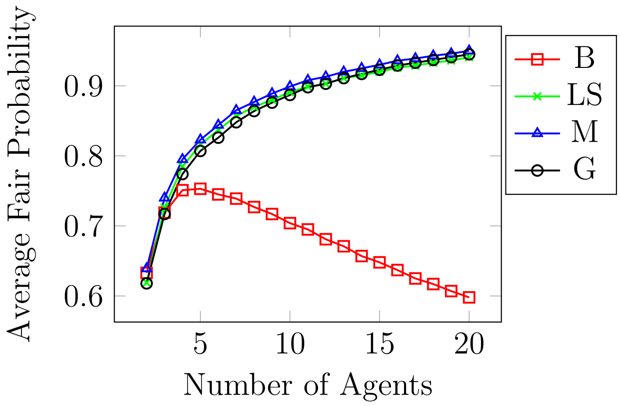


FIGURE 5.1: Algorithm performance over synthetic data

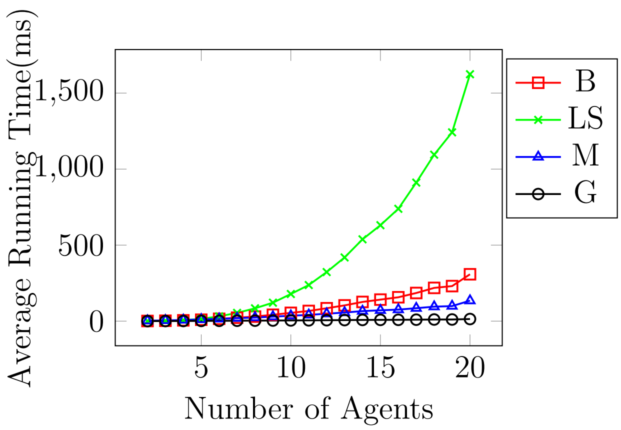


FIGURE 5.2: Algorithm running time over synthetic data

From the figures, one can see that LOCALSEARCH, MATCHING and GREEDY generate allocations with a much higher average fair probability than BASELINE when the number of agents is at least 5. Besides, when the number of agents n increases, the running time of LocalSearch increases significantly, while Baseline, Matching, and Greedy keep low running time. In summary, Greedy and Matching require the least running time and perform the best consistently for different values of n.

Real dataset. Table 5.2 presents the fair probabilities of the suggested allocations by each algorithm on 11 real datasets. Table 5.3 further provides their running times. In the real datasets, Baseline still performs the worst. The other three algorithms can generate allocations that are fair with probability 1 for most cases, with two exceptions for LocalSearch and one exception for Greedy. Matching is the only algorithm that performs consistently well. In terms of the running time, LocalSearch takes much longer than the other three. The Baseline and Matching take a little more time than Greedy. But it is still acceptable.

Table 5.2: Fair probability of all algorithms over real data (sorted by the number of agents n)

no.	n	m	B	LS	M	G
1	24	52	0.03	1.00	1.00	1.00
2	31	54	0.11	1.00	1.00	1.00
3	31	103	0.02	1.00	1.00	1.00
4	32	102	0.06	1.00	1.00	1.00
5	34	63	0.23	1.00	1.00	1.00
6	35	61	0.01	1.00	1.00	1.00
7	37	56	0.00	0.01	1.00	1.00
8	38	133	0.03	1.00	1.00	1.00
9	51	147	0.28	1.00	1.00	1.00
10	51	155	0.00	1.00	1.00	1.00
11	146	176	0.00	0.00	1.00	0.37

Table 5.3: Running time(s) of all algorithms over real data (sorted by the number of agents n)

no.	n	m	B	LS	M	G
1	24	52	0.074	0.908	0.058	0.003
2	31	54	0.078	1.017	0.075	0.005
3	31	103	1.008	29.54	0.212	0.024
4	32	102	1.127	40.02	0.264	0.025
5	34	63	0.143	0.738	0.120	0.006
6	35	61	0.172	1.798	0.140	0.006
7	37	56	0.126	2.531	0.131	0.006
8	38	133	2.148	63.73	0.388	0.039
9	51	147	2.410	9.612	0.618	0.039
10	51	155	3.250	480.7	0.687	0.055
11	146	176	3.559	9255	5.533	0.064

In conclusion, according to the results from both synthetic and real data, LocalSearch has a good performance at a high computation cost. Greedy has decent performance with the best computation efficiency. Matching is a balanced algorithm with consistently good performance and high computation efficiency.

5.5 Conclusions

In this chapter, we present algorithmic and complexity results in computing fair allocations assuming agents' preferences over items are ordinal and have uncertainties. Under the proposed fair notions, we provide polynomial-time solvable algorithms to find the probability that a given allocation is fair and determine whether there exists an allocation with non-zero fair probability. We show that it is NP-hard to see whether there exists an allocation that is fair with probability one. Finally, we show that to find an allocation with the highest fair probability is NP-hard. We further provide several heuristics for this problem. The performance of the heuristics is examined thoroughly on both synthetic and real datasets.

One possible direction for future work is to consider other fairness concepts in the context of ordinal and uncertain preferences and study approximation algorithms to find fair allocations. It is also interesting to combine the fairness notions with other properties such as stability in the uncertain ordinal preference setting.

Part III

Efficiency and Fairness in Offline Resource Allocation

Chapter 6

Price of Fairness

We have already investigated efficiency and fairness separately in the previous two parts. This raises the question: can we achieve good efficiency and fairness guarantees simultaneously? In this part, we explore topics related to balancing efficiency and fairness in offline resource allocation settings.

In this chapter, we examine the price of fairness in both indivisible goods and mixed goods settings, measuring the efficiency loss incurred when considering fairness. This chapter has been published in Li et al. [105].

6.1 Introduction

Fair division is a fundamental topic in algorithmic game theory and has attracted wide attention. In this problem, we need to allocate some resources among agents in a fair manner. The most classic notion of fairness is *envy-freeness (EF)*. When the goods are divisible, meaning that they can be divided into arbitrarily small pieces and allocated to different agents (the cake-cutting problem), an envy-free allocation always exists [3]. However, when considering indivisible goods, meaning that each of them should be allocated to an agent in its entirety, an envy-free allocation may not exist. Thus, a relaxation of envy-freeness, called *envy-freeness up to one good (EF1)*, has been proposed to circumvent this issue. Such an allocation always exists when allocating indivisible goods [106]. Besides EF1, another relaxation of envy-freeness called *envy-freeness up to any good (EFX)* has also received wide attention in recent years [e.g., 2, 4, 52, 53, 55, 73, 118]. Moreover, there have been a number of papers on *partial* EFX allocations with good properties [e.g., 32, 49, 54]. Beyond the setting concerning either divisible or indivisible goods, some recent studies have focused on fairly dividing a mixture of both divisible and indivisible resources [27, 28, 31, 36, 95, 103, 116]. Among these, Bei et al. [27] first considered the mixed-goods setting and proposed the fairness notion called *envy-freeness for mixed goods (EFM)*, which naturally combines envy-freeness and EF1 together. An EFM allocation is guaranteed to exist [27]. By strengthening EF1 to EFX when assessing the envy from others to an agent who receives only indivisible goods, we have a stronger notion than EFM, which is called *envy-freeness up to any good for mixed goods (EFXM)* [27, 116].

In addition to fairness, *efficiency*, or *social welfare*, which refers to the total utility of all the agents towards their bundles, also plays an important role in evaluating an allocation [44, 45]. The *price of fairness*, introduced independently by Bertsimas et al. [34] and Caragiannis et al. [48], is a quantitative measure indicating the loss of social welfare when a given fairness constraint is imposed. More specifically, the price of fairness is defined as the supremum ratio of the maximum social welfare among all allocations to the maximum social welfare among all fair allocations under a particular fairness property, where the supremum is taken over all possible instances. For divisible goods, the price of envy-freeness is $\Theta(\sqrt{n})$, where n is the number of agents [34]. For indivisible goods, the price of EF1 is $\Theta(\sqrt{n})$ [24, 29]. For the special case where $n = 2$, Bei et al. [29] provided a lower bound of $8/7$ and an upper bound of $\frac{2}{\sqrt{3}}$ for the price of EF1 for scaled utilities, and Bu et al. [44] gave a tight ratio of 2 for unscaled utilities. Regarding the price of EFX for indivisible goods, with $n = 2$ agents, Bei et al. [29] provided a tight bound of $3/2$ under scaled utilities, and Bu et al. [44] presented a lower bound of 2 under unscaled utilities. Bu et al. also showed tight bounds for the price of EFX for n agents for both scaled and unscaled utilities, which are $\Theta(\sqrt{n})$ and $\Theta(n)$, respectively.

In this chapter, we close gaps on the price of EF1 and EFX for indivisible-goods allocation when there are two agents [29, 44]. Moreover, for the first time, we provide tight or asymptotically tight bounds on the price of EFM and EFXM when allocating mixed divisible and indivisible goods, resolving questions left open in the literature [107]. Our results are shown in Tables 6.1 and 6.2, where agents'

TABLE 6.1: Price of Envy-Freeness for Two Agents with (Un)Scaled Utilities

Price of ...	EF1	EFM / EFXM	EFX
Scaled	$\frac{8}{7}$ (Thm. 6.1)	$\frac{3}{2}$ (Thm. 6.4)	$\frac{3}{2}$ [29]
Unscaled	2 [44]	2 (Thm. 6.3)	2 (Thm. 6.3)

TABLE 6.2: Price of Envy-Freeness for n Agents with (Un)Scaled Utilities

Price of ...	EF1	EFM / EFXM	EFX
Scaled	$\Theta(\sqrt{n})$ [24, 29]	$\Theta(\sqrt{n})$ (Thm. 6.5)	$\Theta(\sqrt{n})$ [44]
Unscaled	$\Theta(n)$ [44]	$\Theta(n)$ (Thm. 6.6)	$\Theta(n)$ [44]

utilities are *scaled* if for all $i \in N$, $u_i(M \cup D) = 1$, and *unscaled* otherwise. Below, we highlight some interesting features/observations according to our results:

- In all of the settings, tight bounds (or asymptotically tight bounds for an arbitrary number of agents) on the price of envy-freeness are now known.
- For two agents, we show that the price of EF1 is exactly 8/7, which closes the gap between 8/7 and $2/\sqrt{3}$ left open in the previous paper by Bei et al. [29].
- The price of EFM is (asymptotically) the same as the price of EFX in all the settings. This is a potential evidence that EFM, although defined in relation to EF1, is more similar to EFX in nature.

6.1.1 Additional Related Work

While the price of fairness concept captures the efficiency loss in the best fair allocation, Bei et al. [29] introduced the concept of *strong price of fairness*, which captures the efficiency loss in the *worst* allocation. The strong price of fairness has proven to provide meaningful guarantees for fairness notions defined in the form of welfare maximizers, e.g., maximum Nash welfare, maximum Egalitarian welfare, and leximin. The strong price of fairness is, however, too demanding to yield any non-trivial guarantee for fairness notions of interest in this work. To be more specific, the strong price of EF1 and the strong price of EFX are ∞ [29]. We thus only focus on the price of fairness in our work.

The interplay between fairness and efficiency has been extensively studied in the literature of fair division for both divisible and indivisible goods settings [9, 10, 22, 74]. An immediate question is whether a fairness criterion is compatible with Pareto optimality (PO) which is a rather weak economic efficiency measurement. In particular, both envy-freeness and EF1 can be combined with PO by finding the allocation that satisfies maximum Nash welfare in the divisible and indivisible settings respectively [50, 124].

Another related direction considers the problem of maximizing social welfare subject to fairness constraints. For divisible goods, this optimization problem with the envy-freeness constraints for piecewise-constant valuations can be solved optimally [56]. However, this problem has an inapproximability hardness with a polynomial factor if an additional requirement of connectivity on the received piece of cake is imposed [26]. For indivisible goods, the problem with EF1 / EFX criteria is well understood recently [20, 23, 44]. With scaled utilities, Barman et al. [23] gave inapproximability results for general numbers of agents and items while Aziz et al. [20] showed that the problem subject to the EF1 / EFX constraints is NP-hard for some special cases. Bu et al. [44] gave a complete landscape on the computational complexity and approximability of maximizing the social welfare within EF1 / EFX allocations of indivisible goods for both scaled and unscaled utilities.

6.2 Preliminaries

In this chapter, we consider the indivisible goods setting and the mixed goods setting under the multiple homogeneous goods assumption presented in Chapter 2. In this section, we introduce some additional notations related to the price of fairness, which is required for this chapter.

As we mentioned before, agents' utilities are *scaled* if for all $i \in N$, $u_i(M \cup D) = 1$, and *unscaled* otherwise. In this work, we consider both scaled and unscaled utilities.

We are now ready to define the central concept of this work—the price of fairness [24, 29, 34, 48].

Definition 6.1 (Price of Fairness P)**.** For any given fairness criteria P and any instance I, we define

$$\text{price of } P \text{ for instance } I = \frac{\mathsf{OPT}(I)}{\mathsf{SW}(\mathcal{A}^*)},$$

where $\mathcal{A}^*$ is a (partial) allocation that satisfies P and has the maximum social welfare and $\mathsf{OPT}(I)$ is the maximum social welfare over all allocations for instance I. The overall *price of* P is then defined as the supremum price of P across all instances.

Note that when we define the price of EF1 and EFX, we consider instance I with $D = \emptyset$.

Here, when defining the price of fairness, we disregard computational complexity, focusing solely on quantifying the gap between the optimal fair allocation and the optimal allocation.

Partial Allocation and Resource Monotonicity. As a remark, when defining the price of EFM, EFXM and EFX, we include partial allocations into our consideration. To illustrate the idea here, let us first introduce the concept of *resource monotonicity* with respect to social welfare[1] [see, e.g., 44]. Given a fairness property P, we say resource monotonicity *holds* for property P if for any instance, there always exists a complete allocation satisfying P that has a weakly higher social welfare than any other partial allocation satisfying P. Note that when the existence of property P is not guaranteed, resource monotonicity fails for the property.[2]

Regarding EFX, Bu et al. [44] showed that for two agents, resource monotonicity holds for EFX. In general, however, resource monotonicity *fails* for EFX [44]. Put differently, Bu et al. [44] provided an instance in which a partial EFX allocation has a higher social welfare than any complete EFX allocation. Note also that it is a major open problem whether a complete EFX allocation always exists [5]. We have the following two scenarios:

[1]Prior research has also explored the concept of resource monotonicity subject to Pareto optimality. [see, e.g., 123].

[2]For instance, resource monotonicity fails for envy-freeness when allocating indivisible goods.

- if an EFX allocation of indivisible goods always exists, its failure of resource monotonicity suggests that some partial EFX allocation may have higher social welfare than any complete EFX allocation; and
- otherwise we may not even have a complete EFX allocation.

In either case, independent of the existence of EFX, it is more natural to include partial allocations when defining the price of EFX. Since EFXM is a generalization of EFX, it is also natural to consider partial allocations in its price of fairness definition.

In terms of EFM, its existence is guaranteed [27]; however, we do not know whether resource monotonicity holds for EFM (in the mixed-goods setting) or not. If any partial EFM allocation can be extended to a complete EFM allocation with a weakly higher social welfare, it makes no difference whether or not partial allocations are included when defining the price of EFM. Otherwise, it would be more natural to use the "better" partial allocation for characterizing the price of EFM, as opposed to forcing the allocation being complete. Thus, it is again more natural to include partial allocations into our consideration.

For EF1, as noted in [44], it is easy to see that any partial EF1 allocation can be extended to a complete EF1 allocation with a weakly higher social welfare by carrying on the *envy-graph procedure*. Therefore, we do not need to consider partial allocations for the price of EF1.

6.3 Two Agents

In this section, we establish tight bounds on the price of EF1 / EFX / EFM / EFXM for two agents with scaled or unscaled utilities.

6.3.1 Price of EF1 (for Indivisible Goods)

For unscaled utilities, the price of EF1 is exactly 2 due to Bu et al. [44]. For scaled utilities, the price of EF1 is between $\frac{8}{7}$ and $\frac{2}{\sqrt{3}}$ due to Bei et al. [29]. In the following, we close the gap by showing that the price of EF1 is $\frac{8}{7}$.

Theorem 6.1. *For $n = 2$ and scaled utilities (over indivisible goods), the price of EF1 is $\frac{8}{7}$.*

Some additional notations are introduced here for the sake of clarity during this proof. Denote by T_1 the subset of goods that agent 1 values no less than agent 2, and by T_2 the remaining goods:

$$T_1 = \{g \in M \mid u_1(g) \geq u_2(g)\} \qquad \text{and} \qquad T_2 = M \setminus T_1.$$

Let the *surplus* of a bundle M', denoted by $\mathsf{SP}(M')$, be how much agent 1 values bundle M' more than agent 2; formally stated as

$$\mathsf{SP}(M') = \sum_{g \in M'} (u_1(g) - u_2(g)).$$

We first state and prove a useful proposition, and then provide the proof of Theorem 6.1. Given an allocation $(M_1, M_2, \ldots, M_n)$ of indivisible goods M, we say that an agent $i \in N$ *strongly envies* an agent j if and only if for any $g \in M_j$, $u_i(M_i) < u_i(M_j \setminus \{g\})$. It can be seen that given an allocation, if some agent strongly envies some other agent, then the allocation is not EF1. Moreover, if every agent does not strongly envy any other agent, then the allocation is EF1.

Proposition 6.1. *Suppose agent 2 strongly envies agent 1 in the allocation (T_1, T_2). If T_1 can be partitioned into T'_A and T'_B such that agent 2 does not strongly envy agent 1 in the allocation $\mathcal{A} = (T'_A, T'_B \cup T_2)$, then there exists an EF1 allocation $\mathcal{A}'$ with $\mathsf{SW}(\mathcal{A}') \geq \mathsf{SW}(\mathcal{A})$.*

Proof. If agent 1 does not strongly envy agent 2 in $\mathcal{A}$, then let $\mathcal{A}' = \mathcal{A}$ and we are done. Otherwise, we apply the following iterative "one-by-one reassignment" process:

- Suppose, at the beginning of the iteration, agent 1 has bundle T'_1 and agent 2 has bundle T'_2. Select an arbitrary good $g \in T'_2 \cap T_1$. Since agent 1 strongly envies T'_2, she would still envy the bundle if g is excluded, that is, $u_1(T'_1) < u_1(T'_2 \setminus \{g\})$.
- If $u_2(T'_2 \setminus \{g\}) \geq u_2(T'_1)$, assign good g to agent 1. Otherwise we have $u_2(T'_2 \setminus \{g\}) < u_2(T'_1)$. We now swap the two agents' bundles, i.e., let agent 1

get bundle T_2' and agent 2 get bundle T_1'. Note that given the updated allocation, agent 2 is still EF1 towards agent 1.

- If agent 1 still strongly envies agent 2's bundle, go back to step 1 and start another iteration of this process.

Then we prove that the following two invariants hold at the end of each iteration: (a) the social welfare does not decrease throughout the process; (b) agent 2 does not strongly envy agent 1's bundle.

- If agent 2 does not envy agent 1 even when g is excluded from her bundle, moving g to agent 1's bundle would increase social welfare, because $g \in T_1$, indicating that agent 1 values it more than agent 2; if otherwise, the two agents envy each other's bundle, swapping their bundle would also increase social welfare, and item g, too, is reassigned to the agent that values it more.
- If agent 2 does not envy agent 1 even when g is excluded from her bundle, adding g to agent 1's bundle would not lead to agent 2 strongly envying the bundle; if otherwise, both agents would not envy each other's bundle after swapping, and agent 2 certainly does not strongly envy agent 1's bundle after g is reassigned.

The two invariants being established, it can be seen that EF1 is guaranteed after the whole one-by-one reassignment process, and the social welfare of the resulting allocation $\mathcal{A}'$ is no less than $\mathsf{SW}(\mathcal{A})$. □

We are now ready to establish Theorem 6.1.

Proof of Theorem 6.1. For brevity, let y be $\mathsf{SP}(T_1)$. Since scaled utilities are considered here, we should have that $\mathsf{SP}(T_2) = -\mathsf{SP}(T_1) = -y$. Note that assigning each item to the agent that values it more, i.e., assigning T_1 to agent 1 and T_2 to agent 2, achieves the optimal social welfare, so for any instance I with two agents, scaled utilities, and only indivisible goods,

$$\mathsf{OPT}(I) = u_1(T_1) + u_2(T_2) = \mathsf{SP}(T_1) + u_2(T_1) + u_2(T_2) = 1 + y.$$

We divide our proof into three cases.

Case 1: $y \geq \frac{1}{2}$ This case is trivial, since the optimal allocation (allocating each item to the agent who values it more) would have already achieved envy-freeness. More formally speaking, in the optimal allocation, the bundle assigned to agent 1 is T_1, while agent 2 receives T_2. Hence, the utility of agent 1's bundle

$$u_1(T_1) = u_2(T_1) + \mathsf{SP}(T_1) \geq y \geq \frac{1}{2}.$$

Since agent 1 has received more than half of the total value of all items from her perspective, there is no chance that she would envy agent 2's bundle. Similarly, for agent 2,

$$u_2(T_2) = u_1(T_2) - \mathsf{SP}(T_2) = u_1(T_2) + y \geq \frac{1}{2}.$$

Therefore, neither would agent 2 envy agent 1's bundle, and envy-freeness is proved. Because EF1 can be achieved via an allocation optimizing social welfare, the price of EF1 for such instances is 1.

Case 2: $y \leq \frac{1}{3}$ If the optimal allocation already satisfies EF1, we are done. For this reason, we only consider instances where the optimal allocation violates EF1, and suppose without loss of generality that agent 2 strongly envies agent 1. Note that agent 1 does not strongly envy agent 2's bundle in the optimal allocation, for otherwise exchanging their bundle would lead to an allocation with higher social welfare, contradicting to optimality. Then we let agent 2 partition T_1 into two subsets "as evenly as possible", maximizing the utility of the subset with lower utility from her perspective. Call the two subsets T_A and T_B. Suppose $\mathsf{SP}(T_A) \geq \mathsf{SP}(T_B)$. We temporarily assign T_A to agent 1, and $T_B \cup T_2$ to agent 2, and it can be proved that, at this time, (a) agent 2 does not strongly envy agent 1; (b) the social welfare is no less than $\frac{y}{2} + 1$.

- If $u_2(T_B) \geq u_2(T_A)$, then agent 2 values her bundle more than agent 1's, and envy-freeness is guaranteed. If $u_2(T_B) \leq u_2(T_A)$, there should exist a certain item $g \in T_A$ such that $u_2(T_B) \geq u_2(T_A \setminus \{g\})$, for otherwise, moving this item to T_B would result in a "more even" partition. Therefore, $u_2(T_B \cup T_2) \geq u_2(T_B) \geq u_2(T_A \setminus \{g\})$.

- The lower bound of social welfare can be established as follows:

$$\begin{aligned}\mathsf{SW}(T_A, T_B \cup T_2) &= \mathsf{SP}(T_A) + u_2(T_A \cup T_B \cup T_2) \\ &\geq \frac{1}{2}\mathsf{SP}(T_1) + u_2(M) \\ &= \frac{y}{2} + 1.\end{aligned}$$

The inequality can be derived by the fact that $\mathsf{SP}(T_A) \geq \mathsf{SP}(T_B)$.

At this time, agent 1 can still strongly envy agent 2, and we apply Proposition 6.1 to derive a EF1 allocation $\mathcal{A}'$ with social welfare no less than $\frac{y}{2}+1$. Consequently, for instances with $y \leq \frac{1}{3}$, there exists an allocation $\mathcal{A}'$ with $\mathsf{SW}(\mathcal{A}) \geq \frac{y}{2} + 1$ satisfying EF1, ergo the price of EF1

$$\frac{\mathsf{OPT}(I)}{\mathsf{SW}(\mathcal{A}')} \leq \frac{1+y}{1+\frac{y}{2}} \leq \frac{8}{7}.$$

Case 3: $\frac{1}{3} \leq y \leq \frac{1}{2}$ Again, suppose without loss of generality that agent 2 strongly envies agent 1 under optimal allocation. Let agent 2 partition T_1 into three subsets "as evenly as possible", maximizing the subset with minimum utility from her perspective. Let the three subsets be T_A, T_B and T_C, and $u_2(T_A) \geq u_2(T_B) \geq u_2(T_C)$. Three subcases are studied here.

Subcase 3.1: $u_2(T_C) \geq \frac{1}{6}$. Since $u_2(T_2) = u_1(T_2) - \mathsf{SP}(T_2) \geq y \geq \frac{1}{3}$, assigning any one of T_A, T_B, and T_C to agent 2 would result in she claiming more than half of all items' total utility, eliminating the possibility of agent 2 envying agent 1. We assign the subset with smallest surplus, dubbed T_{13}, to agent 2, and the other two, dubbed T_{11} and T_{12}, to agent 1. Thus, $\mathsf{SP}(T_{11} \cup T_{12}) \geq \frac{2}{3}\mathsf{SP}(T_1)$. The social welfare of such allocation $\mathcal{A}$ can be lower bounded via the following calculation

$$\mathsf{SW}(\mathcal{A}) = \mathsf{SP}(T_{11} \cup T_{12}) + u_2(M) \geq 1 + \frac{2}{3}y.$$

Since it is still possible that agent 1 strongly envies agent 2 at this moment, we apply the "one-by-one assignment" process in Proposition 6.1, and derive an allocation $\mathcal{A}'$ satisfying EF1, with social welfare no less than $1 + \frac{2}{3}y$.

Subcase 3.2: $u_2(T_A) \leq \frac{1}{3}$. If any one of T_A, T_B, and T_C is assigned to agent 2, she would not strongly envy agent 1. Assume she gets T_C. For any item $g \in T_B$,

$u_2(T_C) \geq u_2(T_B \setminus \{g\})$ should hold, because moving g to T_C would give a more even partition otherwise. Furthermore, $u_2(T_2) \geq y \geq \frac{1}{3} \geq u_2(T_A)$. Thus, $u_2(T_C \cup T_2) \geq u_2(T_A \cup T_B \setminus \{g\})$ for any $g \in T_B$, proving the claim that agent 2 does not strongly envy agent 1. If T_A or T_B is assigned to agent 2 instead, the proof is similar. Again, assigning the subset with smallest surplus to agent 2 would result in an allocation with social welfare no less than $1 + \frac{2}{3}y$. Since it is possible that agent 1 strongly envies agent 2 at this moment, we apply the "one-by-one assignment" process in Proposition 6.1, and derive an allocation $\mathcal{A}'$ satisfying EF1, with social welfare no less than $1 + \frac{2}{3}y$.

Subcase 3.3: $u_2(T_A) \geq \frac{1}{3}$ and $u_2(T_C) \leq \frac{1}{6}$. In this case, all items in T_A must have values no less than $\frac{1}{6}$, for otherwise moving this item to T_C would bring about a more even allocation. Furthermore, observe that if there are two items in T_A, moving one of them to T_C would also result in a more even allocation. Thus, there can only be one item in T_A. Call it g_A. The optimal allocation here already satisfies EF1, because $u_2(T_2) \geq y = \frac{1}{3}$, and $u_2(T_1 \setminus \{g_A\}) = u_2(T_1) - u_2(g_A) \leq \frac{2}{3} - \frac{1}{3} = \frac{1}{3}$.

Concluding the three subcases discussed above, for any instance I with $\frac{1}{3} \leq y \leq \frac{1}{2}$, there exists an EF1 allocation $\mathcal{A}$ with $\mathsf{SW}(\mathcal{A}) \geq 1 + \frac{2}{3}y$. Therefore, the price of EF1 for instance I is

$$\frac{\mathsf{OPT}(I)}{\mathsf{SW}(\mathcal{A})} \leq \frac{1+y}{1+\frac{2}{3}y} \leq \frac{9}{8}.$$

Combining all three cases, the price of EF1 is at most $\frac{8}{7}$. Together with the lower bound provided in Bei et al. [29], we concluded that for two agents, the price of EF1 is exactly $\frac{8}{7}$. □

6.3.2 Price of EFX / EFM / EFXM

Regarding EFX, it is known that for scaled utilities, the price of EFX is $\frac{3}{2}$ due to Bei et al. [29] as well as for unscaled utilities, the price of EFX is *at least* 2 due to Bu et al. [44]. We provide here a matching upper bound and thus conclude that for unscaled utilities, the price of EFX is exactly 2. In addition, we provide a complete picture of tight bounds on the price of EFM and EFXM for two agents with scaled or unscaled utilities.

Algorithm 4 Cut-and-Choose Algorithm

Input: Fair division instance $\langle [2], M, D, \mathbf{u} \rangle$.

Output: An EFM (or EFX if $D = \emptyset$) allocation with social welfare at least $\frac{u_1(M \cup D) + u_2(M \cup D)}{2}$.

1: Let agent 1 (resp., agent 2) partition the mixed goods into two bundles, denoted by X_1, X_2 (resp., Y_1, Y_2), in the sense that her values for the two bundles are as equal as possible. Assume without loss of generality that $u_1(X_1) \geq u_1(X_2)$, $|u_1(X_1) - u_1(X_2)| \leq |u_2(Y_1) - u_2(Y_2)|$ and that between bundles X_1 and X_2, all goods of value zero for agent 1, if any, are in bundle X_2.;

2: Let agent 2 choose her preferred bundle between X_1 and X_2, and agent 1 get the other bundle. Denote by $\mathcal{A} = (A_1, A_2)$ the resulting allocation.

3: **return** Allocation $\mathcal{A}$

We start by showing that a variant of the well-known *Cut-and-Choose Algorithm* outputs an EFXM (and thus EFM) allocation with social welfare at least one half of $u_1(M \cup D) + u_2(M \cup D)$. The same idea has also been used to show the price of EFX for two agents when allocating indivisible goods; see Theorem 3.4 of Bei et al. [29]. We slightly tailor the algorithm description to allocating mixed goods.

Lemma 6.2. *Given any fair division instance $\langle [2], M, D, \mathbf{u} \rangle$, Algorithm 4 computes an EFXM allocation $\mathcal{A} = (A_1, A_2)$ with social welfare $\mathsf{SW}(\mathcal{A}) \geq \frac{u_1(M \cup D) + u_2(M \cup D)}{2}$. If $D = \emptyset$, $\mathcal{A}$ is EFX.*

Proof. For ease of exposition, we assume without loss of generality that $u_1(X_1) \geq u_1(X_2)$ and $u_2(Y_1) \geq u_2(Y_2)$. Then, according to Algorithm 4, we have

$$u_1(X_1) - u_1(X_2) \leq u_2(Y_1) - u_2(Y_2). \tag{6.1}$$

Put differently, agent 1's partition of the mixed goods is more equal.

We now show that $\mathsf{SW}(\mathcal{A}) \geq \frac{u_1(M \cup D) + u_2(M \cup D)}{2}$. First, we have $u_1(A_1) \geq u_1(X_2)$. Second, we have $u_2(A_2) \geq u_2(Y_1)$; otherwise, agent 2 could have a more equal partition of the mixed goods, a contradiction to our assumption. The social welfare of allocation $\mathcal{A}$ is lower bounded by

$$\mathsf{SW}(\mathcal{A}) = u_1(A_1) + u_2(A_2) \geq u_1(X_2) + u_2(Y_1) \geq u_1(X_1) + u_2(Y_2),$$

where the last transition is due to Equation (6.1). It implies that

$$\begin{aligned}\mathsf{SW}(\mathcal{A}) &\geq \frac{u_1(X_2) + u_2(Y_1) + u_1(X_1) + u_2(Y_2)}{2} \\ &= \frac{u_1(M \cup D) + u_2(M \cup D)}{2},\end{aligned}$$

as desired.

Finally, we show that allocation $\mathcal{A}$ is EFXM. Agent 2 gets her preferred bundle, so she is envy-free and hence EFXM. Regarding agent 1, she is envy-free (and hence EFM) if she receives bundle X_1. In the case that agent 1 gets bundle X_2, if agent 1 still has envy after removing some indivisible good or some amount of divisible goods from bundle X_1, then, by moving the good to bundle X_2, agent 1 could have created a more equal partition, a contradiction. As a result, allocation $\mathcal{A}$ is EFXM. When $D = \emptyset$, this implies that allocation $\mathcal{A}$ is EFX. □

We are now ready to show the tight bounds on price of EFX / EFM / EFXM for two agents, and start with the case of agents having unscaled utilities.

Theorem 6.3. *For $n = 2$ and unscaled utilities, the price of EFX / EFM / EFXM is 2.*

Proof. The lower bound of 2 for both the price of EFX (and thus EFXM) and the price of EFM (note that EFM generalizes EF1) follows from Theorem F.4 of Bu et al. [44].

We now show the matching upper bound. Consider an arbitrary instance $I = \langle [2], M, D, \mathbf{u} \rangle$. It is easy to see that $\mathsf{OPT}(I) \leq u_1(M \cup D) + u_2(M \cup D)$. Together with Lemma 6.2, we conclude that the price of these three fairness notions is at most 2. □

Our next result is the price of EFM and the price of EFXM for two agents with scaled utilities.

Theorem 6.4. *For $n = 2$ and scaled utilities, the price of EFM and the price of EFXM is $\frac{3}{2}$.*

Proof. Lower bound: Consider the following instance with one indivisible good g_1 and two homogeneous divisible goods d_1, d_2, and assume that the utilities are as follows:

	g_1	d_1	d_2
Agent 1's value	$1/2$	$1/2-\varepsilon$	ε
Agent 2's value	$1/2$	ε	$1/2-\varepsilon$

The optimal social welfare is $3/2 - 2\varepsilon$, achieved by assigning goods g_1 and d_1 to agent 1, and good d_2 to agent 2. On the other hand, in any EFM allocation, no agent can get both the indivisible good g_1 and any positive amount of the divisible goods. Hence, the social welfare of an EFM allocation is at most 1. Taking $\varepsilon \to 0$, we find that the price of EFM is at least $3/2$. Since EFXM is stronger than EFM, this also implies that the price of EFXM is at least $3/2$.

Upper bound: Consider an arbitrary instance. If in an optimal allocation both agents get utility at least $1/2$, this allocation is envy-free (due to the assumptions of additive and scaled utilities) and hence EFM and EFXM; therefore, in this case, the price of EFM is 1. Otherwise, the maximum social welfare is at most $1+1/2 = 3/2$. According to Lemma 6.2, Algorithm 4 returns an EFXM (and thus EFM) allocation $\mathcal{A}$ with $\mathsf{SW}(\mathcal{A}) \geq \frac{u_1(M\cup D)+u_2(M\cup D)}{2}$. Since utilities are scaled, we have $\mathsf{SW}(\mathcal{A}) \geq 1$, implying that the price of EFXM and the price of EFM is at most $3/2$. □

6.4 Arbitrary Number of Agents

In this section, we establish asymptotically tight bounds on the price of EFM and the price of EFXM for n agents, and begin with the case that agents' valuations are scaled.

Theorem 6.5. *For scaled utilities, the price of EFM and the price of EFXM are* $\Theta(\sqrt{n})$.

Proof. Since EFM generalizes EF1 and EFXM implies EFM, the lower bound $\Omega(\sqrt{n})$ follows from Bei et al. [29].

To show the upper bound $O(\sqrt{n})$, we make use of the result that the price of EFX is $\Theta(\sqrt{n})$ shown by Bu et al. [44]. The high-level idea is as follows. We first split each divisible good $d_{\overline{k}}$ into ℓ smaller goods $d_{\overline{k}}^1, d_{\overline{k}}^2, \ldots, d_{\overline{k}}^\ell$ of equal size, and we treat each of the ℓ smaller goods as an indivisible good. In other words, we are considering an instance I^ℓ with a total of $m + \overline{m}\ell$ indivisible goods. For each ℓ, we find an EFX allocation that achieves an $O(\sqrt{n})$-approximation to $\mathsf{OPT}(I^\ell)$. When $\ell \to \infty$, we have a sequence of EFX allocations which converges to a "limit allocation" that is EFXM (and thus EFM), and the limit allocation exists due to the compactness of the allocation space. This limit allocation characterizes the upper bound $O(\sqrt{n})$ for the price of EFXM, as the social welfare is a continuous function on the allocation space.

To prove the upper bound formally, we start from defining the instance I^ℓ and the allocation $\mathcal{A}^\ell$. In the instance I^ℓ, we have the same set of agents N, and a set of $m + \overline{m}\ell$ *indivisible* goods which consist of the m goods in M and $\overline{m}\ell$ goods $\{d_{\overline{k}}^1, d_{\overline{k}}^2, \ldots, d_{\overline{k}}^\ell\}_{\overline{k}=1,\ldots,\overline{m}}$ as described earlier. The result of Bu et al. [44] indicates that there exists a (partial) EFX allocation $\mathcal{A}$ for instance I^ℓ such that $\frac{\mathsf{OPT}(I^\ell)}{\mathsf{SW}(\mathcal{A})} \leq c\sqrt{n}$ for some constant c and sufficiently large n. Let $\mathcal{A}^\ell$ be such an allocation. Notice that A^ℓ is also a valid allocation for the original instance I (where each $d_{\overline{k}}$ is divisible), and we will use $\mathcal{A}^\ell$ for the same allocation in both I^ℓ and I.

Next, we will define an allocation $\mathcal{A}$ for the original instance I which is a "limit allocation" for the allocation sequence $\{\mathcal{A}^\ell\}_{\ell=1}^\infty$. To make the notion of limit valid, we need to define a metric space for the set of all allocations, and this is defined in the following natural way. First note that there are $(n+1)^m$ ways to allocate the indivisible goods (each good can be allocated to one of the n agents, or unallocated), which is finite. For each fixed allocation of the indivisible goods, an allocation of the divisible goods $\{d_1, d_2, \ldots, d_{\overline{m}}\}$ can be naturally described by a point in the following subset of $\mathbb{R}^{n\overline{m}}$:

$$\chi = \Bigg\{ (x_{i\overline{k}})_{i=1,\ldots,n;\overline{k}=1,\ldots,\overline{m}} \in \mathbb{R}^{n\overline{m}} : \sum_{i=1}^n x_{i\overline{k}} \leq 1 \text{ for each } \overline{k} \in [\overline{m}], \\ \text{and } x_{i\overline{k}} \geq 0 \text{ for each } i \in [n] \text{ and } \overline{k} \in [\overline{m}] \Bigg\}.$$

Given two allocations, the distance between them in the metric space is defined as follows:

- if their corresponding allocations for M are different, the distance is ∞;
- if their corresponding allocations for M are the same, the distance is defined by the Euclidean distance of the two points in χ describing their allocations for D.

Since χ is closed and bounded and the space of all allocations is a union of finitely many ($(n+1)^m$ to be precise) such closed and bounded sets, the Bolzano-Weierstrauss Theorem [25] implies that the allocation space contains at least one allocation that is a limit point for the sequence $\{\mathcal{A}^\ell\}_{\ell=1}^{\infty}$. Let $\mathcal{A}$ be one such limit point. In the remaining part of the proof, we will conclude the theorem by showing that 1) $\mathcal{A}$ is an EFXM allocation and 2) it satisfies the approximation guarantee $\frac{\mathsf{OPT}(I)}{\mathsf{SW}(\mathcal{A})} = O(\sqrt{n})$.

$\mathcal{A}$ is EFXM Suppose this is not the case. There exist two agents i and j such that $u_i(A_i) < u_i(A_j)$ and A_j contains some divisible good (i.e., $x_{j\overline{k}} > 0$ for some $\overline{k}$). We choose a sufficiently small value $\delta_{\overline{k}}$ such that $3\delta_{\overline{k}} \in (0, x_{j\overline{k}})$ and $u_i(A_j) - u_i(A_i) > 3\delta_{\overline{k}} \cdot u_i(d_{\overline{k}})$. In other words, removing an amount $3\delta_{\overline{k}}$ of good $d_{\overline{k}}$ from A_j will not stop agent i from envying agent j. Since u_i is a continuous function (which can be proved by a straightforward application of the definition of continuity given our definition of the metric space) and $\mathcal{A}$ is a limit point of the sequence $\{\mathcal{A}^\ell\}_{\ell=1}^{\infty}$, by considering a sufficiently small neighbourhood of $\mathcal{A}$, there exists ℓ with allocation $\mathcal{A}^\ell = (A_1^\ell, \ldots, A_n^\ell)$ such that

- $|u_i(A_i) - u_i(A_i^\ell)| < \delta_{\overline{k}} \cdot u_i(d_{\overline{k}})$,
- $|u_i(A_j) - u_i(A_j^\ell)| < \delta_{\overline{k}} \cdot u_i(d_{\overline{k}})$, and
- $\delta_{\overline{k}} > \frac{1}{\ell}$.

Points 1 and 2 above imply $u_i(A_j^\ell) - u_i(A_i^\ell) > \delta_{\overline{k}} \cdot u_i(d_{\overline{k}})$ under the condition that $u_i(A_j) - u_i(A_i) > 3\delta_{\overline{k}} \cdot u_i(d_{\overline{k}})$. Point 3 further implies that, in the instance I^ℓ, there exists an indivisible item corresponding to a small portion that is smaller than $\delta_{\overline{k}}$ of $d_{\overline{k}}$ whose removal will not stop agent i from envying agent j. This contradicts to our construction that $\mathcal{A}^\ell$ is EFX.

Approximation guarantee By our construction of the sequence with the result of Bu et al. [44], for sufficiently large n and a fixed constant c, we have $\frac{\mathsf{OPT}(I^\ell)}{\mathsf{SW}(\mathcal{A}^\ell)} \leq$

$c\sqrt{n}$ for every ℓ. It suffices to show that

$$\mathsf{OPT}(I) = \lim_{\ell \to \infty} \mathsf{OPT}(I^\ell) \qquad \text{and} \qquad \mathsf{SW}(\mathcal{A}) = \lim_{\ell \to \infty} \mathsf{SW}(\mathcal{A}^\ell).$$

The second limit follows from the continuity of the function $\mathsf{SW}(\,\cdot\,)$, where the continuity can be proved by a straightforward application of the definition of continuity. The first limit follows from the fact that $\mathsf{OPT}(I) = \mathsf{OPT}(I^\ell)$ for each ℓ. To see this, in the optimal allocation, we allocate each divisible good $d_{\bar{k}}$ as a whole to a single agent who values it the highest (with tie broken arbitrarily), so it does not matter how each divisible good is sub-divided to multiple indivisible smaller goods. □

We now proceed to show the price of EFM and the price of EFXM for unscaled utilities.

Theorem 6.6. *For unscaled utilities, the price of EFM and the price of EFXM are* $\Theta(n)$*.*

Theorem 6.6 can be proved by using the result that the price of EFX for unscaled valuations is $\Theta(n)$ from Bu et al. [44] and applying the discretization-with-limit technique used in the proof of Theorem 6.5. Here, we give an alternative constructive proof of Theorem 6.6.

When allocating indivisible goods, Lemma 1 of Barman et al. [24] proved that there always exists an EF1 allocation with an absolute welfare guarantee. We show a similar result holds when allocating mixed goods. To be more specific, by slightly tweaking Algorithm 1 (Alg-EF1-Abs) of Barman et al. [24], we can compute a partial EFXM allocation with a similar absolute welfare guarantee:

Lemma 6.7. *Given any fair division instance* $\langle N, M, D, \mathbf{u} \rangle$*, Algorithm 5 computes a partial EFXM allocation* $\mathcal{A}$ *with social welfare* $\mathsf{SW}(\mathcal{A}) \geq \frac{1}{2n+1} \cdot \sum_{i \in N} u_i(M \cup D)$*.*

We first give some intuition behind the proof. At a high level, Barman et al.'s algorithm starts from a maximum weight matching where each agent receives exactly one indivisible good, and then performs the envy-cycle elimination procedure of Lipton et al. [106]. At the end, an EF1 allocation $(A_1, A_2, \ldots, A_n)$ of indivisible goods is obtained. In many "natural scenarios", we have $u_i(A_i) \geq \frac{1}{2} u_i(A_j)$

Algorithm 5 A partial EFXM allocation with an absolute welfare guarantee

Input: Fair division instance $\langle N, M, D, \mathbf{u} \rangle$.
Output: A partial EFXM allocation $\mathcal{A} = (A_1, A_2, \ldots, A_n)$ with social welfare $\mathsf{SW}(\mathcal{A}) \geq \frac{1}{2n+1} \cdot \sum_{i \in N} u_i(M \cup D)$.

if $|M| \geq n$ **then**
    $\widetilde{M} \leftarrow M$
else
    Let $\widetilde{M}$ be the union of indivisible goods M and some dummy indivisible goods, for which each agent has value zero, such that $|\widetilde{M}| = n$.
end if
Consider the weighted bipartite graph $G = (N \cup \widetilde{M}, N \times \widetilde{M})$ with weight of each edge $(i, g) \in N \times \widetilde{M}$ setting as $u_i(g)$. Let π be a maximum-weight matching in G that matches all nodes in N.
Construct the partial allocation $\mathcal{A}' = (A'_1, A'_2, \ldots, A'_n)$ such that $A'_i = \{\pi(i)\}$ for each $i \in N$.
Use Algorithm 2.1 of Chaudhury et al. [54] to extend allocation $\mathcal{A}'$ by allocating the remaining indivisible goods and obtain a partial EFX allocation.
Use Algorithm 1 of Bei et al. [27] to allocate the divisible goods and obtain a partial EFXM allocation $\mathcal{A} = (A_1, A_2, \ldots, A_n)$, where $A_i = (\widetilde{M}_i, \mathbf{x}_i)$.
return Allocation $\mathcal{A}$

as removing one indivisible good from A_j eliminates the envy from i to j. This gives us the $2n$ approximation ratio to the optimal social welfare. The inequality $u_i(A_i) \geq \frac{1}{2} u_i(A_j)$ can only fail in the case when the removed indivisible good g from A_j is "large" so that $u_i(\{g\}) > u_i(A_j \setminus \{g\})$. However, the initial allocation with the maximum weight matching ensures that the "large indivisible goods" are "reasonably allocated" so that the inequality holds in some average sense. Barman et al. worked out the calculations to make the approximation guarantee $2n$ hold, and we find out that this set of arguments can be extended to the setting with mixed divisible and indivisible goods.

Proof. We first add some dummy indivisible goods for which each agent has value zero, if needed, so that there are at least n indivisible goods; denote by $\widetilde{M}$ the (possibly extended) set of indivisible goods. Let $\widetilde{M}^i$ be a set of n most valuable indivisible goods from $\widetilde{M}$ to each agent $i \in N$. Note that those dummy indivisible goods do not affect the social welfare of any allocation.

Consider a subgraph $G' = (N \cup \widetilde{M}, E)$ of the weighted bipartite graph G, where $E = \{(i, g) \colon i \in N, g \in \widetilde{M}^i\}$. In other words, subgraph G' only considers edges from each agent $i \in N$ to her n most valuable indivisible goods. Due to the same argument in the proof of Barman et al. [24, Lemma 1], we have

$$\mathsf{SW}(\mathcal{A}') = \sum_{i \in N} u_i(\pi(i)) \geq \frac{1}{n} \cdot \sum_{i \in N} \sum_{g \in \widetilde{M}^i} u_i(g).$$

Since each agent receives a single indivisible good, the (partial) allocation $\mathcal{A}'$ is EFXM. According to Lemmas 2.5 and 2.7 of Chaudhury et al. [54] and the analysis of Algorithm 1 of Bei et al. [27], we have $u_i(A_i) \geq u_i(A'_i)$ after executing Lines 8-9. As a result,

$$\mathsf{SW}(\mathcal{A}) = \sum_{i \in N} u_i(A_i) \geq \sum_{i \in N} u_i(A'_i) = \mathsf{SW}(\mathcal{A}') \geq \frac{1}{n} \cdot \sum_{i \in N} \sum_{g \in \widetilde{M}^i} u_i(g),$$

or, alternatively,

$$n \cdot \mathsf{SW}(\mathcal{A}) \geq \sum_{i \in N} \sum_{g \in \widetilde{M}^i} u_i(g). \tag{6.2}$$

Together with the property that no one envies the unallocated indivisible goods P, stated in Chaudhury et al. [54, Theorem 2.8], we have:

$$\mathsf{SW}(\mathcal{A}) = \sum_{i \in N} u_i(A_i) \geq \sum_{i \in N} u_i(A'_i) = \mathsf{SW}(\mathcal{A}') \geq \sum_{i \in N} u_i(P). \tag{6.3}$$

Because allocation $\mathcal{A}$ is EFXM, for each pair of agents $i, j \in N$, there exists $S_j \subseteq \widetilde{M}_j$ with $|S_j| \leq 1$ such that

$$u_i(A_i) \geq u_i(\widetilde{M}_j \setminus S_j, \mathbf{x}_j) = u_i(A_j) - u_i(S_j). \tag{6.4}$$

Recall that we may have unallocated indivisible goods P. Summing the above inequality over $j \in [n]$, we have

$$\begin{aligned} n \cdot u_i(A_i) &\geq \sum_{j \in [n]} u_i(A_j) - \sum_{j \in [n]} u_i(S_j) \\ &= u_i(\widetilde{M} \cup D \setminus P) - \sum_{j \in [n]} u_i(S_j) \\ &\geq u_i(\widetilde{M} \cup D \setminus P) - \sum_{g \in \widetilde{M}^i} u_i(g), \end{aligned}$$

where the last inequality holds because the n sets S_j's are disjoint and have cardinality at most 1 each, and $\widetilde{M}^i$ is the set of the n most valuable goods to agent i. Next, summing the above inequality over $i \in N$, we have

$$n \cdot \sum_{i \in N} u_i(A_i) = n \cdot \mathsf{SW}(\mathcal{A}) \geq \sum_{i \in N} u_i(\widetilde{M} \cup D \setminus P) - \sum_{i \in N} \sum_{g \in \widetilde{M}^i} u_i(g).$$

Finally, plugging Equations (6.2) and (6.3) into the above inequality, we have

$$\begin{aligned} & n \cdot \mathsf{SW}(\mathcal{A}) + n \cdot \mathsf{SW}(\mathcal{A}) + \mathsf{SW}(\mathcal{A}) \\ \geq & \sum_{i \in N} u_i(\widetilde{M} \cup D \setminus P) - \sum_{i \in N} \sum_{g \in \widetilde{M}^i} u_i(g) + \sum_{i \in N} \sum_{g \in \widetilde{M}^i} u_i(g) + \sum_{i \in N} u_i(P) \\ = & \sum_{i \in N} u_i(M \cup D), \end{aligned}$$

which implies

$$\mathsf{SW}(\mathcal{A}) \geq \frac{1}{2n+1} \cdot \sum_{i \in N} u_i(M \cup D),$$

as desired. □

Proof of Theorem 6.6. Since EFM generalizes EF1, the desired lower bound of $\Omega(n)$ follows from Theorem 1 of Barman et al. [24]. Since EFXM implies EFM, we obtain the same lower bound of $\Omega(n)$ for the price of EFXM.

We now show the asymptotic matching upper bound for the price of EFXM. Consider an arbitrary instance. Since $\sum_{i \in N} u_i(M \cup D)$ is a trivial upper bound on the optimal social welfare of the instance, Lemma 6.7 implies the desired upper bound of $O(n)$ for the price of EFXM.

Next, we establish the asymptotic matching upper bound for the price of EFM. Given that EFXM implies EFM, we can readily deduce an upper bound of $O(n)$ for the price of EFM. Moreover, if we want to find a complete EFM allocation, we can adapt Line 9 of Algorithm 5 in the following way:

- Use Algorithm 1 of Bei et al. [27] to extend the partial EFM allocation $\mathcal{A}'$ by allocating both the remaining indivisible goods and divisible goods into a complete EFM allocation.

Following a similar argument, we can conclude that $\mathsf{SW}(\mathcal{A}) \geq \frac{1}{2n} \cdot \sum_{i \in N} u_i(M \cup D)$.[3] Thus, the price of EFM is also $\Theta(n)$. □

6.5 Conclusions

In this chapter, we have given a complete characterization for the price of envy-freeness in various settings. The bounds we provide are tight for two agents and asymptotically tight for any number of agents. In particular, we close a gap left open in [29] by showing a tight bound for the price of EF1 for two agents. Furthermore, the price of fairness has been studied for the setting with divisible goods and the setting with indivisible goods, but it is much less understood for allocating mixed divisible and indivisible goods. The work in this chapter fills in this missing piece.

[3] To compute the social welfare lower bound for the complete EFM allocation, we do not need Equation (6.3) in the proof of Lemma 6.7.

Chapter 7

Efficiency and Fairness in Cloud Computing

In this chapter, we focus on a specific resource allocation problem for cloud computing systems, where the preferences of agents towards items are Leontief preferences instead of additive functions. In this setting, we introduce and investigate a novel approximation ratio measure called *fair-ratio*, which quantifies the efficiency gap between a fair mechanism and an "ideal" fair allocation. This chapter has been published in Bei et al. [30].

7.1 Introduction

In order to offer flexible resources and economies of scale, in cloud computing systems, a fundamental problem is to efficiently allocate heterogeneous computing resources, such as CPU time and memory, to agents with different demands. This resource allocation problem presents several significant challenges from a technical perspective. For example, how to balance the efficiency of the system and fairness among users? How to incentivize agents to participate and truthfully reveal their private information? These are all delicate issues that need to be carefully considered when designing a resource allocation algorithm.

One of the most widely used mechanisms for multi-type resource allocation is the *Dominant Resource Fairness (DRF)* mechanism proposed by [75]. This work assumes that agents in the system have *Leontief* preferences, which means they demand to receive resources of each type in fixed proportions. Under such preferences, the proposed DRF mechanism generalizes the max-min allocation by equalizing the share of the most demanded resource, called *dominant share*, for all agents. Ghodsi et al. [75] show that DRF satisfies a set of desirable properties. These include fairness properties which we describe before: (i) *share incentive (SI)*, and (ii) *envy-freeness (EF)*; efficiency properties: (iii) *Pareto optimality (PO)*, it is impossible to increase the allocation of one agent without decreasing the allocation of another agent; as well as incentive properties: (iv) *strategy-proofness (SP)*, no agent can benefit from reporting a false demand. Consequently, DRF has received significant attention with many variants proposed to tackle different restrictions occurred in practice.

Despite the above attractive properties, however, DRF is known to have poor performance in terms of utilitarian social welfare, which is defined as the sum of utilities of all agents. Many alternative mechanisms have then been proposed to tackle this issue and balance the trade-off between fairness and efficiency [38, 39, 77, 89, 90, 92, 129, 130]. Most of these mechanisms still satisfy SI, EF, and PO. However, none of them satisfy SP. Recently, Jiang and Wu [89] propose the so called 2-dominant resource fairness (2-DF) to balance fairness and efficiency. Different from other mechanisms, 2-DF satisfies SP and PO, but does not satisfy SI and EF generally. On the other hand, Parkes et al. [117] justify this worst-case performance of DRF by showing that any mechanism satisfying any of the three properties SI, EF, and SP cannot guarantee more than $\frac{1}{m}$ of the optimal social welfare, which is also what DRF can achieve. Here m denotes the number of resource types. This characterization seems to suggest that from a worst-case viewpoint, DRF has the best possible social welfare guarantee among all fair or truthful mechanisms.

In this work, we aim to design new mechanisms that satisfy the same set of properties with DRF but with better efficiency guarantees. In order to get around the theoretical barrier set by [117], we first propose and justify a new benchmark to measure the social welfare guarantee of a mechanism. Note that Parkes et al. [117] and many other works use the *approximation ratio*, which is defined as the worst-case ratio between the *optimal social welfare among all allocations* and the mechanism's social welfare, as the performance measure of a mechanism. However, since SI and EF are both fairness properties that place significant constraints on feasible allocations, it is not surprising that any allocation satisfying SI or EF would incur a large approximation ratio of m. On the other hand, one can show that any mechanism satisfying SI has approximation ratio at most m. This means all mechanisms satisfying SI and EF will have the same worst-case approximation ratio, which renders the approximation ratio notion meaningless in systems where these fairness conditions are hard constraints that must be satisfied. Since fairness is a hard constraint in many practical applications, we argue that it is more reasonable to compare the mechanism's social welfare to the optimal social welfare among *all allocations that satisfy SI and EF*. To this end, we modify the approximation ratio definition and propose this according variant. The new definition allows us to get pass the lower bound barrier from [117] and design mechanisms with better social welfare approximation ratio guarantees.

7.1.1 Our results

We design new resource allocation mechanisms that satisfy properties such as SI, EF, PO, and SP, and at the same time achieve high efficiency. The efficiency is measured by two objectives: *social welfare*, defined as the sum of utilities of all agents, and *utilization*, defined as the minimum utilization rate among all resources. Social welfare is an indicator commonly used to measure efficiency, while improving utilization rate is also an important goal for cloud providers for cost-saving (see, e.g., Amazon[1], IBM[2]). In academia, utilization has been studied by [91, 92, 101]. For the performance measure, we define *fair-ratio* for social welfare (resp. utilization) of a mechanism as the worst-case ratio between the social welfare (resp. utilization) achieved by the optimal mechanism *satisfying SI and EF* and that by the mechanism. See formal definitions in Section 7.2.

We first focus on the setting where all agents' dominant resources fall into two types. This is the most basic and arguably also the most important setting in cloud computing and other application domains such as high performance computing. For example, most existing commercial cloud computing services, such as Azure, Amazon EC2, and Google Cloud, work with only two (dominant) resources:

[1]https://aws.amazon.com/blogs/aws/cloud-computing-server-utilization-the-environment/
[2]https://www.ibm.com/cloud/learn/cloud-computing

Table 7.1: Fair-ratio results for $m = 2$ resources overview.

	Social Welfare		Utilization	
DRF (Lemma 7.1)	2	$(2-\alpha)$	∞	$(\frac{1}{\alpha})$
UNB (Theorem 7.2)	$\frac{3}{2}$	$(1+\alpha)$	2	$(\frac{1}{1-\alpha})$
BAL (Theorem 7.3)	$\frac{4}{3}$	$(\frac{4-2\alpha}{3-\alpha})$	2	$(\frac{2}{1+\alpha})$
BAL* (Theorem 7.4)	$\left[\frac{4-2\alpha}{3-\alpha}, \frac{4-2\alpha}{3-\alpha-\frac{1}{n}}\right]$		$\left[\frac{2}{1+\alpha}, \frac{2}{1+\alpha-\frac{1}{n}}\right]$	

CPU and memory. Two-resource setting can also be used to model the coupled CPU-GPU architectures where CPU and GPU are integrated into a single chip to achieve high performance computing [129]. In this setting, we present three new mechanisms UNB, BAL, and BAL*, all with better fair-ratio guarantees than DRF. Different from DRF which equalizes the dominant share of all agents, the idea behind our new mechanisms is to partition all agents into two groups according to their dominant resources and carefully increase the share of agents with the smallest fraction of their non-dominant resource in each group. Mechanism UNB satisfies all four properties (SI, EF, PO, and SP) and has a fair-ratio of $\frac{3}{2}$ for social welfare and 2 for utilization. Mechanism BAL further improves the fair-ratio for social welfare to $\frac{4}{3}$. However, BAL satisfies SI, EF, and PO, but not SP. Finally, we generalize BAL to a new mechanism BAL* which satisfies all the four properties and has the same asymptotic fair-ratio as BAL when the number of agents n goes to infinity. We further provide a more fine-grained analysis of the fair-ratio parameterized by a *minority population ratio* parameter $\alpha \in (0, \frac{1}{2}]$, which is defined as the fraction of agents in the smaller group classified by their dominant resources. Table 7.1 lists a summary of the fair-ratios of different mechanisms in the worst case and in terms of α. We also compare our mechanisms with DRF by conducting experiments on both synthetic and real-world data. Our results match well with the theoretical bounds of fair-ratios and show that both UNB and BAL* achieve better social welfare and utilization than DRF.

Next we move to the general situation with $m \geq 2$ resources. We first give a family $\mathcal{F}$ of mechanisms, containing DRF as a special case, that satisfy all the four properties. This answers the question posed by [75] that *"whether DRF is the only possible strategy-proof policy for multi-resource fairness, given other desirable* *properties such as Pareto efficiency"*. Unfortunately, as we will see in the next part, for general m all mechanisms that satisfy the four properties will have the same fair-ratio as DRF. Nevertheless, we show that a generalization of UNB still satisfies the four properties and its fair-ratio is always weakly better than DRF.

7.1.2 Related work

Since its introduction by [75], DRF has been extended in multiple directions, including the setting with weighted agents or indivisible tasks [117], the setting when resources are distributed over multiple servers with placement constraints [128, 134] or without placement constraints [70, 133], a dynamic setting when agents arrive at different times [94] and the case when agents' demands are limited [101, 114]. In contrast to these works, we consider the original setting and aim to design mechanisms with better efficiency guarantees than DRF. Notably, Li et al. [101] generalize DRF to the limited demand setting, and study the approximation ratio of the generalized mechanism by comparing it with the optimal allocation satisfying PO, SI and EF. Essentially, their results implies that for two resources, the fair-ratio of DRF is 2 for social welfare and ∞ for utilization, which can be seen as a special case of our more fine-grained result in Lemma 7.1 parameterized by α. Dolev et al. [61] advocate a different fairness notion called *Bottleneck Based Fairness (BBF)* for multi-resource allocation with Leontief preferences and show that a BBF allocation always exists. Gutman and Nisarr [79] extend DRF and BBF for a larger family of utilities and give a polynomial time algorithm to compute a BBF solution. Characterization of mechanisms satisfying a set of desirable properties under Leontief preferences has been studied in economics literature [71, 100, 115]. However, they consider different properties than what we consider.

7.2 Preliminaries

In this chapter, we consider the cloud computing model presented in Chapter 2. In this section, we introduce some additional notations needed for this chapter.

7.2.1 Dominant Resource Fairness (DRF)

The *DRF* mechanism [75] works by maximizing and equalizing the dominant shares of all agents, subject to the feasible constraint. Let x be the dominant share of each agent, DRF solves the following linear program:

$$\begin{aligned} \text{maximize} \quad & x \\ \text{subject to} \quad & \sum_{i \in N} x \cdot d_{ir} \leq 1, \quad \forall r \in R \end{aligned}$$

This linear program can be rewritten as $x^* = \frac{1}{\max_{r \in R} \sum_{i \in N} d_{ir}}$. Then, for agent i the allocation $\mathbf{A}_i = x^* \cdot \mathbf{d}_i$.

7.2.2 Properties of mechanisms

Beyond the two fairness notions SI and EF presented in Chapter 2, we are also interested in the following properties of a resource allocation mechanism.

Before stating the properties, we first formally define the *mechanism*. Denote the set of all instances by $\mathcal{I}$, and the set of all feasible allocations by $\mathcal{A}$. A *mechanism* is a function $f : \mathcal{I} \to \mathcal{A}$ that maps every instance to a feasible allocation. We use $f_i(\mathbf{I})$ to denote the allocation vector to agent i under instance $\mathbf{I}$. A mechanism is non-wasteful if the allocation $\mathbf{A}$ of the mechanism on any instance satisfies that for each agent $i \in N$ there exists $y \in \mathbb{R}_+$ such that $A_{ir} = y \cdot d_{ir}, \forall r \in R$. In words, for each agent, the amount of allocated resources are proportional to its normalized demand vector. In this chapter, we only consider non-wasteful mechanisms.

Definition 7.1 (Pareto Optimality (PO)). An allocation $\mathbf{A}$ is PO if it is not dominated by another allocation $\mathbf{A}'$, i.e., there is no $\mathbf{A}'$ such that $\exists i_0 \in N$: $u_{i_0}(\mathbf{A}'_{i_0}) > u_{i_0}(f_{i_0}(\mathbf{I}))$ and $\forall i \in N : u_i(\mathbf{A}'_{i_0}) \geq u_i(f_i(\mathbf{I}))$. A mechanism f is PO if for any instance $\mathbf{I} \in \mathcal{I}$ the allocation $f(\mathbf{I})$ is PO.

It is easy to verify that a non-wasteful mechanism satisfies PO if and only if at least one resource is used up in the allocation returned by the mechanism.

Definition 7.2 (Strategyproofness (SP)). A mechanism f is SP if no agent can benefit by reporting a false demand vector, i.e., $\forall \mathbf{I} \in \mathcal{I}, \forall i \in N, \forall \mathbf{d}'_i, u_i(f_i(\mathbf{I})) \geq$

$u_i(f_i(\mathbf{I}'))$, where $\mathbf{I}'$ is the resulting instance by replacing agent i's demand vector by $\mathbf{d}'_i$.

7.2.3 Approximation ratio

Since we have already defined *social welfare (SW)* in Chapter 2, it suffices to define the utilization. As in [101], we define *utilization* of an allocation $\mathbf{A}$ as the minimum utilization rate of m resources,

$$\mathrm{U}(\mathbf{A}) = \min_{r \in R} \sum_{i \in N} A_{ir}.$$

As discussed in the introduction, we use a revised notion of approximation ratio to measure the efficiency performance of a mechanism, where we use the optimal *fair* allocation as the benchmark instead of the original benchmark which is based on the optimal allocation.

Definition 7.3. The *fair-ratio for social welfare (resp. utilization)* of a mechanism f is defined as, among all instances $\mathbf{I} \in \mathcal{I}$, the maximum ratio of the optimal social welfare (resp. utilization) among all allocations that satisfy SI and EF over the social welfare (resp. utilization) of $f(\mathbf{I})$, i.e.,

$$\mathrm{FR}_{\mathsf{SW}} = \max_{\mathbf{I} \in \mathcal{I}} \frac{\max_{\mathbf{A} \text{is SI,EF}} \mathsf{SW}(\mathbf{A})}{\mathsf{SW}(f(\mathbf{I}))} \quad \text{and} \quad \mathrm{FR}_{\mathrm{Util}} = \max_{\mathbf{I} \in \mathcal{I}} \frac{\max_{\mathbf{A} \text{is SI,EF}} \mathrm{U}(\mathbf{A})}{\mathrm{U}(f(\mathbf{I}))}.$$

7.3 Two Types of Resources

In this section we focus on the case where there are only two competing resources. More specifically, we assume that among the m types of resources, there exists $r_1, r_2 \in R$, such that for any agent i and any other resource $r \neq r_1, r_2$, we have $d_{ir_1} \geq d_{ir}$ and $d_{ir_2} \geq d_{ir}$. This means in any allocation, other resources will not run out before r_1 or r_2 runs out. Thus it is equivalent to assume that R contains only two resources r_1 and r_2.

We partition all agents into two groups G_1 and G_2, where $G_i (i = 1, 2)$ consists of all agents whose dominant resource is r_i. Agents with demand vector $(1, 1)$

are considered to be in G_1. Denote $n_1 = |G_1|$ and $n_2 = |G_2|$. Without loss of generality, we assume that $n_1 \geq \frac{n}{2}$ (otherwise we can rename the two resources).

We now let

$$\alpha := \frac{n_2}{n} \in (0, \frac{1}{2}]$$

be the fraction of agents in the smaller group and we call α the *minority population ratio*. We assume that $\alpha > 0$, because when $\alpha = 0$ the only allocation satisfying SI is to give every agent $\frac{1}{n}$ of the first resource (and the corresponding amount of the second resource). As we will see in the following, α is crucial in analyzing the fair-ratio of a mechanism.

We start by analyzing the fair-ratio of DRF.

Lemma 7.1. *With 2 resources, for instances with minority population ratio α, we have*

$$\mathrm{FR}_{\mathsf{SW}}(\mathrm{DRF}) = 2 - \alpha \quad \textit{and} \quad \mathrm{FR}_{\mathrm{Util}}(\mathrm{DRF}) = \frac{1}{\alpha}.$$

For a better illustration of the results in this chapter, we defer all the proofs to the end of this chapter.

When α approaches 0, we have $\mathrm{FR}_{\mathsf{SW}}(\mathrm{DRF}) \to 2$ and $\mathrm{FR}_{\mathrm{Util}}(\mathrm{DRF}) \to \infty$. Notice that with 2 resources $\mathrm{FR}_{\mathsf{SW}}(f)$ for any mechanism f satisfying SI is at most 2 as the mechanism can always achieve at least 1 in SW.

In the following, we present two new mechanisms with the same set of properties as DRF but with better fair-ratios.

7.3.1 Mechanism UNB

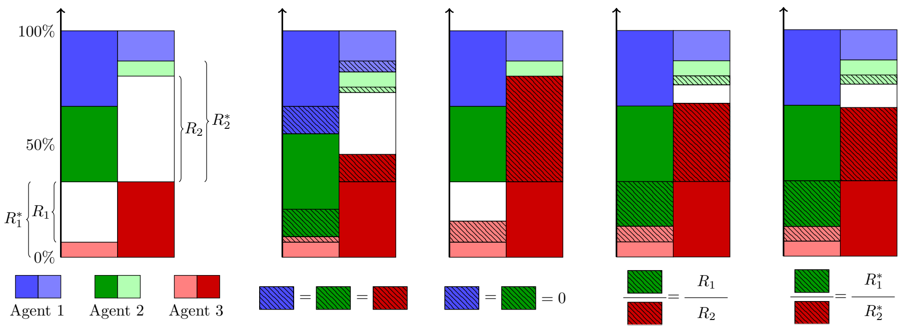
The more detailed analysis of Lemma 7.1 shows that when the population of two groups are unbalanced, i.e., when α is close to 0, it is better to allocate more resources to agents in the minor group G_2 with smaller $d_{i,1}$. This idea leads to mechanism UNB, described in Algorithm 6. The mechanism has two steps. In step 1, the mechanism allocates every agent $\frac{1}{n}\mathbf{d}_i$ of resources such that each agent has a dominant share of $\frac{1}{n}$, which ensures SI. In step 2, the mechanism repeats the following process till one resource is used up: Select a set of agents from G_2 who have the smallest fraction t_1 of resource r_1, denoted by P, and increase their

Algorithm 6 UNB$(\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_n)$

$C \leftarrow (c_1, c_2) = (1, 1)$ // remaining resources
$G_1 \leftarrow \{i \mid d_{i,1} = 1\}$; $G_2 \leftarrow \{i \mid d_{i,1} < 1\}$
for all $i \in N$ **do**
  $\mathbf{A}_i \leftarrow \frac{1}{n}\mathbf{d}_i$ // every agent receives $\frac{1}{n}$ dominant share
  $C \leftarrow C - \mathbf{A}_i$
end for
while $c_1 > 0$ and $c_2 > 0$ **do**
  $P \leftarrow \arg\min_{i \in G_2} A_{i,1}$ // agents with the smallest fraction of resource r_1
  $\delta_0 \leftarrow \min\limits_{i \in N \setminus P} A_{i,1} - \min\limits_{i \in P} A_{i,1}$ // increasing step when 2nd smallest fraction of resource r_1 is reached
  $\delta_1 \leftarrow \frac{C_1}{|P|}$, $\delta_2 \leftarrow \frac{C_2}{\sum_{i \in P} \frac{1}{d_{i,1}}}$ // increasing step when resource r_1 (or r_2) is used up
  $\delta^* \leftarrow \min\{\delta_0, \delta_1, \delta_2\}$
  for all $i \in P$ **do**
    $\mathbf{A}_i \leftarrow \mathbf{A}_i + (\delta^*, \frac{\delta^*}{d_{i,1}})$ // increase resource r_1 by the same δ^*
    $C \leftarrow C - (\delta^*, \frac{\delta^*}{d_{i,1}})$
  end for
end while
return A

fractions of resource r_1 at the same speed (δ^*) till the fraction reaches the second smallest fraction t_2 in G_2 ($\delta^* = \delta_0$) or one resource is used up ($\delta^* = \delta_1$ for resource r_1 and $\delta^* = \delta_2$ for resource r_2).

Example 7.1. Consider an instance with 3 agents who have demand vectors $\mathbf{d}_1 = (1, \frac{2}{5})$, $\mathbf{d}_2 = (1, \frac{1}{5})$ and $\mathbf{d}_3 = (\frac{1}{5}, 1)$. We compare the allocation under UNB and DRF. Notice that DRF can also be viewed as a two-step mechanism, where in step 1 every agent gets $\frac{1}{n}$ dominant share (the same as UNB) and in step 2 we increase the dominant share of every agent at the same speed till one resource is used up. For the above instance, in step 1 all 3 agents get $\frac{1}{3}$ dominant share, and the remaining resource is $C = (\frac{4}{15}, \frac{7}{15})$, corresponding to Figure 7.1a. In step 2, under DRF, all agents have the same dominant share $x^* = \frac{1}{\max\{\frac{11}{5}, \frac{8}{5}\}} = \frac{5}{11}$ and the final allocation vectors are $A_1 = (\frac{5}{11}, \frac{2}{11})$, $A_2 = (\frac{5}{11}, \frac{1}{11})$ and $A_3 = (\frac{1}{11}, \frac{5}{11})$, corresponding to Figure 7.1b. Under UNB, we increase the allocation of agent 3, who currently has the smallest fraction $\frac{1}{15}$ of resource r_1, till the second resource r_2 is used up and we have $A_3 = (\frac{4}{25}, \frac{4}{5})$, corresponding to Figure 7.1c. The SW under DRF is $\frac{5}{11} \times 3 \approx 1.36$, while the SW under UNB is $\frac{1}{3} + \frac{1}{3} + \frac{4}{5} \approx 1.47$.



(A) Step 1 (B) DRF Step 2 (C) UNB Step 2 (D) BAL Step 2 (E) BAL* Step 2

FIGURE 7.1: Allocations under DRF, UNB, BAL and BAL* in Example 1. The shaded area represents the added parts in respective Step 2.

We show that UNB satisfies all four properties and has a better fair-ratio than DRF.

Theorem 7.2. *With* 2 *resources, mechanism* UNB *can be implemented in polynomial time, satisfies SI, EF, PO, and SP, and has*

$$\text{FR}_{\textsf{SW}}(\text{UNB}) = 1 + \alpha \quad \textit{and} \quad \text{FR}_{\text{Util}}(\text{UNB}) = \frac{1}{1-\alpha}.$$

Specifically, mechanism UNB *can be implemented in* $O(n^2)$ *time and* $O(n)$ *space.*

Because $\alpha \in (0, 1/2]$, we have $\text{FR}_{\textsf{SW}}(\text{UNB}) \leq 3/2$ and $\text{FR}_{\text{Util}}(\text{UNB}) \leq 2$, both of which are significantly better than DRF.

The intuition behind $\text{FR}_{\textsf{SW}}(\text{UNB}) \leq 1 + \alpha$ is that under UNB agents in G_1 get at most α less utility than the optimal allocation and agents in G_2 get no less utility than the optimal allocation. For the lower bound $\text{FR}_{\textsf{SW}}(\text{UNB}) \geq 1 + \alpha$, consider instances where after step 1 the remaining resource is $C = (\alpha - \epsilon, \epsilon)$ with $\epsilon \to 0$. In step 2 UNB can only increase the allocations of agents in G_2 and get SW at most $1 + \epsilon$, while the optimal allocation can increase the allocation of agents in G_1 with the smallest $d_{i,2}$ and get SW of $1 + \alpha - \epsilon$.

7.3.2 Mechanism BAL

According to Theorem 7.2, UNB has the worst performance when the population of two groups are balanced, i.e., when α is close to $\frac{1}{2}$, because in step 2 it only

Algorithm 7 BAL($\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_n$)

$C \leftarrow (c_1, c_2) = (1, 1)$ // remaining resources
$G_1 \leftarrow \{i \mid d_{i,1} = 1\}$; $G_2 \leftarrow \{i \mid d_{i,1} < 1\}$
for all $i \in N$ **do**
  $\mathbf{A}_i \leftarrow \frac{1}{n}\mathbf{d}_i$ // every agent receives $\frac{1}{n}$ dominant share
  $C \leftarrow C - \mathbf{A}_i$
end for
$(R_1, R_2) \leftarrow C$ // remaining resources after step 1
while $c_1 > 0$ and $c_2 > 0$ **do**
  $P_1 \leftarrow \arg\min_{i \in G_1} A_{i,2}$
  $P_2 \leftarrow \arg\min_{i \in G_2} A_{i,1}$
  $(\delta_1^*, \delta_2^*) \leftarrow$ CalcStep () // calculate increasing steps
  for $k = 1, 2$ **do**
    for all $i \in P_k$ **do**
      $\mathbf{A}_i \leftarrow \mathbf{A}_i + \frac{\delta_k^*}{d_{i,3-k}}\mathbf{d}_i$ // increase the non-dominant resource by the same δ_k^*
      $C \leftarrow C - \frac{\delta_k^*}{d_{i,3-k}}\mathbf{d}_i$
    end for
  end for
end while
return A

increases allocations of agents in one group (G_2). In this case, a better strategy in step 2 is to increase allocations of agents from both groups.

Following this intuition, we propose mechanism BAL, described in Algorithm 7. Mechanism BAL also has two steps. Step 1 is the same as UNB, where every agent gets $\frac{1}{n}$ dominant share. In step 2, the mechanism increases allocations of agents from both groups, and within each group the method is the same as in UNB, that is, within each group, only agents who have the smallest amount of the non-dominant resource will be allocated more resources, and they will be allocated the same fraction (δ_1^* or δ_2^*) of the non-dominant resource. In addition, BAL controls the relative allocation rates (δ_1^*, δ_2^*) of two groups such that the ratio between the increased dominant shares of two groups is proportional to the ratio between the remaining amounts of two resources after step 1. Formally, let ΔS_1 and ΔS_2 be the sum of increased dominant share of agents in G_1 and G_2 in step 2 respectively. Let $R_1 = 1 - \frac{n_1}{n} - \frac{1}{n}\sum_{i \in G_2} d_{i,1}$ and $R_2 = 1 - \frac{n_2}{n} - \frac{1}{n}\sum_{i \in G_1} d_{i,2}$ be the amount of remaining resources after step 1, then BAL ensures that

$$\frac{\Delta S_1}{\Delta S_2} = \frac{R_1}{R_2}. \tag{7.1}$$

Algorithm 8 CalcStep ()

for $k = 1, 2$ **do**
  $\delta_k \leftarrow \min\limits_{i \in N \setminus P_k} A_{i,3-k} - \min\limits_{i \in P_k} A_{i,3-k}$ // increasing step when 2nd smallest fraction of resource r_{3-k} is reached
  $D_k \leftarrow \sum_{i \in P_k} \frac{1}{d_{i,3-k}}$
  $\overline{\delta_k} \leftarrow \frac{c_{3-k}}{|P_k| + D_k \frac{R_{3-k}}{R_k}}$ // increasing step when resource r_{3-k} is used up
  $\delta_k^* \leftarrow \min\{\delta_k, \overline{\delta_k}\}$
end for
if $\frac{\delta_1^* D_1}{\delta_2^* D_2} \leq \frac{R_1}{R_2}$ **then**
  $\delta_2^* \leftarrow \delta_1^* \cdot \frac{D_1}{D_2} \cdot \frac{R_2}{R_1}$ // decrease δ_2^* according to δ_1^*
else
  $\delta_1^* \leftarrow \delta_2^* \cdot \frac{D_2}{D_1} \cdot \frac{R_1}{R_2}$ // decrease δ_1^* according to δ_2^*
end if
return (δ_1^*, δ_2^*)

This condition is crucial to guarantee the good performance of BAL.

To compute the increasing steps (δ_1^*, δ_2^*) (CalcStep () in Line 10 of Algorithm 7), we calculate the largest increasing steps (δ_1^*, δ_2^*) such that condition (7.1) is satisfied and one of the following conditions is satisfied: (a) one resource has been used up; (b) one agent has to be added into P_1 or P_2. The concrete algorithm is given in Algorithm 8.

We now show that BAL satisfies SI, EF, and PO, and its fair-ratio for SW is at most $\frac{4}{3}$.

Theorem 7.3. *With 2 resources, mechanism* BAL *can be implemented in polynomial time, satisfies SI, EF, and PO, and has*

$$\mathrm{FR}_{\mathsf{SW}}(\mathrm{BAL}) = \frac{4-2\alpha}{3-\alpha} \quad \textit{and} \quad \mathrm{FR}_{\mathrm{Util}}(\mathrm{BAL}) = \frac{2}{1+\alpha}.$$

Specifically, mechanism BAL *can be implemented in* $O(n^2)$ *time and* $O(n)$ *space.*

The intuition behind $\mathrm{FR}_{\mathsf{SW}}(\mathrm{BAL}) \leq \frac{4-2\alpha}{3-\alpha}$ is that compared with the optimal allocation, where in step 2 the sum of increased dominant share of agents in G_1 and G_2 are ΔS_1^* and ΔS_2^* respectively, we can show that either $\Delta S_1 \geq \Delta S_1^*$ or $\Delta S_2 \geq \Delta S_2^*$, and $\Delta S_i \geq \frac{1}{2} \Delta S_i^*$ for any $i \in \{1, 2\}$. Combining them with the fact that $\max\{\Delta S_1^*, \Delta S_2^*\} \leq 1-\alpha$, we get

$$\mathrm{FR}_{\mathsf{SW}}(\mathrm{BAL}) = \frac{1+\Delta S_1^*+\Delta S_2^*}{1+\Delta S_1+\Delta S_2} \leq \max_{i\in\{1,2\}} \frac{1+\Delta S_i^*}{1+\frac{1}{2}\Delta S_i^*} \leq \frac{2-\alpha}{1+\frac{1-\alpha}{2}} = \frac{4-2\alpha}{3-\alpha}.$$

For the lower bound $\mathrm{FR}_{\mathsf{SW}}(\mathrm{BAL}) \geq \frac{4-2\alpha}{3-\alpha}$, consider instances where after step 1 the remaining resource is $C = (\epsilon, 1-\alpha-\epsilon)$ with $\epsilon \to 0$. In step 2 the optimal allocation can allocate all the remaining resource to agent i^* in G_2 who has demand vector $(\frac{\epsilon}{1-\alpha-\epsilon}, 1)$ and get SW of $2-\alpha-\epsilon$, while for BAL, because of the condition (7.1), we can only give about half of the remaining resource r_1 to i^* and the other half to agents in G_1 such that $\frac{\Delta S_1}{R_1} = \frac{\Delta S_2}{R_2} \approx \frac{1}{2}$, where the SW is about $1+\frac{1-\alpha}{2} = \frac{1}{2}(3-\alpha)$.

However, BAL does not satisfy SP as the agent with the minimum $d_{i,1}$ (or minimum $d_{i,2}$) could influence the ratio $\frac{R_1}{R_2}$ by modifying its demand vector to get more resources in step 2, as shown in the following example.

Example 7.2. Consider an instance with two agents who have demand vectors $\mathbf{d}_1 = (1, \frac{1}{2})$ and $\mathbf{d}_2 = (\frac{1}{4}, 1)$. According to BAL, in step 1 agent 1 gets $(\frac{1}{2}, \frac{1}{4})$ and agent 2 gets $(\frac{1}{8}, \frac{1}{2})$. Then the remaining resources is $(\frac{3}{8}, \frac{1}{4})$ and the increasing speed ratio is $\frac{3}{2}$. In step 2, agent 1 gets $(\frac{3}{14}, \frac{3}{28})$ and agent 2 gets $(\frac{1}{28}, \frac{1}{7})$, and resource 2 is used up. Overall agent 2 gets $(\frac{9}{56}, \frac{9}{14})$. However, if agent 2 reports another demand vector $\mathbf{d}_2' = (\frac{1}{2}, 1)$, then both agents will get the same dominant share $\frac{2}{3}$ under BAL. In particular, agent 2 will get $(\frac{1}{3}, \frac{2}{3})$, which is strictly better than $(\frac{9}{56}, \frac{9}{14})$. Therefore, BAL is not SP.

7.3.3 Mechanism BAL*

Though BAL is not SP, we can make BAL satisfy SP with a small modification. In the following we propose a slightly different mechanism BAL* that replaces the condition (7.1) by the following condition:

$$\frac{\Delta S_1}{\Delta S_2} = \frac{R_1^*}{R_2^*} = \frac{R_1 + \frac{1}{n} d_{i^*,1}}{R_2 + \frac{1}{n} d_{j^*,2}}, \tag{7.2}$$

where i^* is an agent in G_2 with the minimum $d_{i,1}$ and j^* is an agent in G_1 with the minimum $d_{i,2}$. That is, the ratio between ΔS_1 and ΔS_2 is proportional to the ratio between the remaining amounts of two resources when all agents except i^* and j^* get $\frac{1}{n}$ dominant share. Intuitively, for agent i^*, this modification prevents it from increasing $d_{i^*,1}$ to influence $\frac{R_1^*}{R_2^*}$, unless $d_{i^*,1}$ becomes larger than the second smallest $d_{i,1}$, for which case we can show that i^* cannot benefit.

We show that BAL* satisfies all four properties including SP, and its fair-ratio is very close to that of BAL.

Theorem 7.4. *With 2 resources, mechanism* BAL* *can be implemented in polynomial time, satisfies SI, EF, PO, and SP, and has*

$$\mathrm{FR}_{\mathsf{SW}}(\mathrm{BAL}^*) \in \left[\frac{4-2\alpha}{3-\alpha}, \frac{4-2\alpha}{3-\alpha-\frac{1}{n}}\right]; \quad \mathrm{FR}_{\mathrm{Util}}(\mathrm{BAL}^*) \in \left[\frac{2}{1+\alpha}, \frac{2}{1+\alpha-\frac{1}{n}}\right].$$

Specifically, mechanism BAL* *can be implemented in* $O(n^2)$ *time and* $O(n)$ *space.*

Example 7.1 (continued). We compare BAL and BAL* for the instance in Example 7.1. Step 1 is the same as before and we have $\frac{R_1}{R_2} = \frac{\frac{4}{15}}{\frac{7}{15}} = \frac{4}{7}$ and $\frac{R_1^*}{R_2^*} = \frac{\frac{4}{15}+\frac{1}{15}}{\frac{7}{15}+\frac{1}{15}} = \frac{5}{8}$. In step 2, under BAL, we increase the allocation of agent 2 by $(\frac{16}{81}, \frac{16}{405})$, and that of agent 3 by $(\frac{28}{405}, \frac{28}{81})$ such that resource r_1 is used up. Notice that $\Delta S_1 = \frac{16}{81}$ and $\Delta S_2 = \frac{28}{81}$ satisfy $\frac{\Delta S_1}{\Delta S_1} = \frac{R_1}{R_2}$. Under BAL*, we increase the allocation of agent 2 by $(\frac{20}{99}, \frac{4}{99})$, and that of agent 3 by $(\frac{32}{495}, \frac{32}{99})$. Notice that $\Delta S_1 = \frac{20}{99}$ and $\Delta S_2 = \frac{32}{99}$ satisfy $\frac{\Delta S_1}{\Delta S_1} = \frac{R_1^*}{R_2^*}$. These two corresponds to Figure 7.1d and 7.1e. The SW under BAL and BAL* is ≈ 1.54 and ≈ 1.53 respectively, which is larger than 1.36 under DRF and 1.47 under UNB.

Figure 7.2 shows fair-ratios of DRF, UNB, BAL, and BAL* (when $n \to \infty$) as a function of α. Notice that all three new mechanisms have better fair-ratio than DRF for any $\alpha \in (0, \frac{1}{2})$. Among new mechanisms, UNB has better fair-ratio than BAL (BAL*) when α is close to 0 while BAL (BAL*) has better fair-ratio than UNB when α is close to 0.5.

7.3.4 Experimental evaluation

The above analysis of fair-ratio shows that our mechanism UNB and BAL* have better performance than DRF from the worst-case perspective. In this section, we compare the performance of DRF, UNB and BAL* when $m = 2$ using both synthetic instances and real-world instances based on Google cluster-usage traces [121]. Our results are shown in Figure 7.3, where we plot the ratio between the optimal

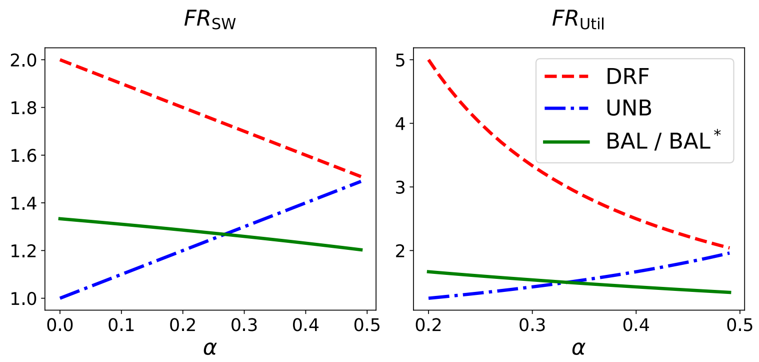


FIGURE 7.2: Fair-ratio of mechanisms as a function of α. As $\text{FR}_{\text{Util}}(DRF) \to \infty$ when $\alpha \to 0$, for better visualization, we only show FR_{Util} for $\alpha \in [0.2, 0.5]$.

allocation (satisfying SI and EF) and the allocation under compared mechanisms. Our results match well with the above fair-ratios and show that both UNB and BAL* achieve better social welfare and utilization than DRF.

Random instances with different α. First we compare mechanisms on random instances with fixed $n = 100$ and different $\alpha \in \{0.05, 0.10, \ldots, 0.50\}$. For each α, we average over 1000 instances to get the data point. To control the value of α, we choose $n(1-\alpha)$ agents and set $d_{i,1} = 1$ for them, and for the remaining agents we set $d_{i,2} = 1$. The other entries of the demand vectors are sampled uniformly from $\{0.01, 0.02, \ldots, 1.00\}$.

The result is shown in the first row of Figure 7.3. For SW, BAL* is very close to the optimal solution (the ratio is close to 1) and BAL* is always better than DRF for different values of α. UNB also outperforms DRF for most values of α except when $\alpha \in [0.45, 0.5]$. Comparing UNB and BAL*, similarly to the crossing point of their theoretical fair-ratios in Figure 7.2, their performance on random instances also cross when $\alpha \approx 0.25$ in Figure 7.3, confirming that when $\alpha \to 0$, UNB is better than BAL*, and when $\alpha \to 0.5$, BAL* is better than UNB. When $\alpha \geq 0.2$, the performance trend of three mechanisms matches well with the fair-ratio. More precisely, when α increases, BAL* and DRF perform better while UNB performs worse. The comparison of three mechanisms in utilization is almost the same as in SW.

Instances generated from Google trace. Next we test mechanisms on instances that are generated according to the real demands of tasks from the Google

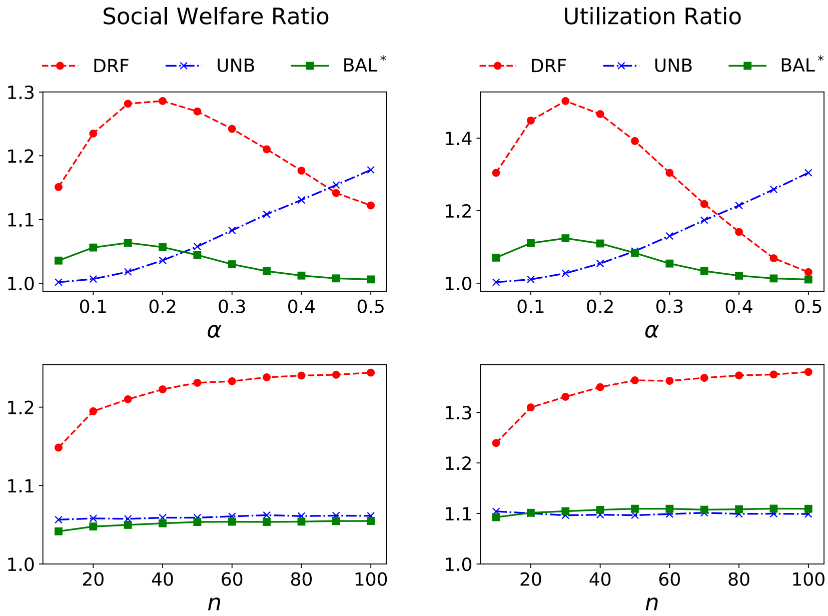


FIGURE 7.3: Performance ratio between the optimal allocation and allocations under DRF, UNB, BAL* on synthetic instances with different α (1st row) and real-world instances with different n (2nd row).

traces. The Google traces record the demands for CPU and memory of each submitted task. We normalize these demands to get a pool of normalized demand vectors. Then we generate instances by randomly sampling demand vectors from this pool. We compare mechanisms on instances with different number $n \in \{10, 20, \ldots, 100\}$ of agents. For each n, we average over 1000 instances to get the data point.

The result is shown in the second row of Figure 7.3. For both SW and utilization, UNB and BAL* outperform DRF and the improvements are more than 10%. The performance of UNB and BAL* are very close, because in the demand vector pool more agents (about 67%) have CPU as the dominant resource and hence the generated instances have α close to 0.33. Notice that the fair-ratios of UNB and BAL* are indeed very close when $\alpha = 0.33$ (see Figure 7.2).

7.4 Multiple Types of Resources

We move to the general case with $m \geq 2$ types of resources.

7.4.1 A family of mechanisms

We start by presenting a large family of mechanisms that satisfy the four desired properties SI, EF, PO, and SP, which includes DRF as a special case. This is in response to the question asked in [75] that *"whether DRF is the only possible strategy-proof policy for multi-resource fairness, given other desirable properties such as Pareto efficiency"*. Although many mechanisms based on DRF have been proposed for different settings and there are characterizations of mechanisms satisfying desirable properties under Leontief preferences [71, 100, 115], to the best of our knowledge, there is no work that directly answers this question.

We call a function g that maps vectors $\mathbf{v} \in [0,1]^m$ to $\mathbb{R}$ *monotone*, if it satisfies that for any two vectors $\mathbf{v}_1, \mathbf{v}_2$ with $\mathbf{v}_1 > \mathbf{v}_2$, $g(\mathbf{v}_1) > g(\mathbf{v}_2)$. Here $\mathbf{v}_1 > \mathbf{v}_2$ means $\mathbf{v}_1$ is element-wise strictly larger than $\mathbf{v}_2$. Denote $\mathcal{G}$ the set of all monotone functions. Now we define a family of mechanisms $\mathcal{F}$ based on monotone functions. For each monotone function $g \in \mathcal{G}$, we define a mechanism $f_g \in \mathcal{F}$ as follows. The mechanism contains two steps that have the same flavor as UNB. In step 1, every agent receives $\frac{1}{n}$ dominant share. In step 2, we increase the allocation for agents that have the minimum value of $g(\mathbf{A}_i)$ till some resource is used up. We show that all mechanisms in $\mathcal{F}$ satisfy the four desired properties.

Theorem 7.5. *For any m, every mechanism $F_g \in \mathcal{F}$ satisfies SI, EF, PO, and SP.*

With the large family of mechanisms at hand, the next question is to check if there exists any mechanism from $\mathcal{F}$ that can achieve better efficiency than DRF. Unfortunately, as we will see in the next part, all mechanisms from $\mathcal{F}$ will have the same approximation guarantee for general m. This means from a worst-case analysis point of view, no mechanism has a provable better SW or utilization than DRF. Thus a more fine-grained analysis is needed to find better mechanisms. In the next section, we analyze a special mechanism from $\mathcal{F}$, which can be seen as a generalization of UNB, by considering two parameters.

7.4.2 Generalization of UNB

Similar to the case with 2 resources, we first partition all agents into m groups $G_i (i \in [m])$ according to their dominant resources and choose an arbitrary group (say G_1) as a special group. Then, we let $\alpha := 1 - \frac{|G_1|}{n}$ be the fraction of agents not in G_1, and let $\beta := \sum_{i \in N \setminus G_1} \frac{d_{i,1}}{n\alpha}$ be the average demand of agents not in G_1 for resource r_1.

UNB can be generalized as follows. In step 1, each agent gets $\frac{1}{n}$ dominant share. In step 2, we increase the allocation of agents who have the smallest fraction of resource r_1 in the same speed for resource r_1, till some resource is used up. With slight abuse of notation, we still call this generalized mechanism UNB. Note that this mechanism is equivalent to the mechanism from the family $\mathcal{F}$ with monotone function $g(\mathbf{v}) = \mathbf{v}_1$. We prove the fair-ratio of UNB and DRF parameterized by α and β in the following theorem.

Theorem 7.6. *With $m \geq 3$ resources, mechanism* UNB *can be implemented in polynomial time, satisfies SI, EF, PO, and SP, has* $\mathrm{FR}_{\mathrm{Util}}(\mathrm{UNB}) = \mathrm{FR}_{\mathrm{Util}}(\mathrm{DRF}) = \infty$, *and*

$$\mathrm{FR}_{\mathsf{SW}}(\mathrm{UNB}) = \max \left\{ m - \alpha\beta - (1-\alpha), \frac{m - \alpha\beta}{1 + \frac{1-\beta}{\beta}\alpha} \right\},$$

compared to

$$\mathrm{FR}_{\mathsf{SW}}(\mathrm{DRF}) = \max \left\{ m - \alpha\beta - (1-\alpha), (m - \alpha\beta)(1 - \alpha(1-\beta)) \right\}.$$

Specifically, mechanism UNB *can be implemented in $O(n^2 m)$ time and $O(nm)$ space.*

In particular, one can show that for any $\alpha, \beta \in (0,1)$, $1 - \alpha(1-\beta) > \frac{1}{1 + \frac{1-\beta}{\beta}\alpha}$. This means $\mathrm{FR}_{\mathsf{SW}}(\mathrm{UNB})$ is always weakly better than $\mathrm{FR}_{\mathsf{SW}}(\mathrm{DRF})$.

7.5 Conclusions

In this chapter, we investigate the multi-type resource allocation problem. Generalizing the classic DRF mechanism, we propose several new mechanisms in the two-resource setting and in the general m-resource setting. The new mechanisms satisfy the same set of desirable properties as DRF but with better efficiency guarantees.

For future works, we hope to extend these mechanisms to handle more realistic assumptions, such that when agents have limited demands or indivisible tasks, and when agents arrive at different times.

7.6 Appendix

We list some proofs for theorems in this chapter here.

7.6.1 Proof of Lemma 7.1

We first show that fair-ratio for SW is $2-\alpha$. For the lower bound, we build an instance with minority population ratio α and n agents as follows. The first group G_1 consists of $n(1-\alpha)$ agents who have the same demand vector $(1, \varepsilon)$, where $\varepsilon = \frac{1}{n}$. The second group G_2 consists of $n\alpha$ agents, where except for one special agent i^* whose demand vector is $(\frac{\varepsilon}{2}, 1)$, all other agents have the same demand vector $(1-\varepsilon, 1)$. We choose n large enough such that $n\alpha \geq 2$. The idea is that under DRF since all agents must have the same dominant share, their dominant share is close to $\frac{1}{n}$ because of the limit of the first resource and hence SW will be close to 1, while there exists an allocation that satisfies SI, EF, and PO, and has SW close to $2-\alpha$ by giving roughly $1-\alpha$ dominant share to the special agent i^*.

Formally, under DRF, the first resource will be used up and the dominant share of every agent is

$$\frac{1}{n(1-\alpha)+(n\alpha-1)(1-\varepsilon)+\frac{\varepsilon}{2}} \leq \frac{1}{n-2},$$

so SW of DRF is at most $\frac{n}{n-2}$. However, if we give $\frac{1}{n}$ dominant share to every agent except for agent i^* and give i^* the bundle $(x\frac{\varepsilon}{2}, x)$, where $x = (1-\alpha)(1-\frac{1}{n})$, such that the second resource is used up, then the SW is

$$1 - \frac{1}{n} + x \geq 2 - \alpha - \frac{2}{n}.$$

It is easy to verify that the above allocation, denoted by $\mathbf{A}^*$ satisfies SI, EF, and PO. For EF, notice that the special agent i^* receives $x\frac{\varepsilon}{2} + \frac{1}{n^2} \leq \frac{1}{2n} + \frac{1}{n^2}$ of the first resource while all other agents in G_2 receive $\frac{1-\frac{1}{n}}{n}$ of the first resource. So fair-ratio for SW of DRF is at least

$$\frac{2-\alpha-\frac{2}{n}}{\frac{n}{n-2}} \xrightarrow{n\to\infty} 2-\alpha.$$

For the upper bound, for any instance **I** with SW $s \geq 1$ under DRF, we show that SW of any allocation satisfying SI is upper bounded by $(2-\alpha)s$. Thus, fair-ratio for SW is upper bounded by $2-\alpha$. Let **A** be an arbitrary allocation on **I** satisfying SI. If the first resource is used up under DRF, then agents in G_1 get $(1-\alpha)s$ of the first resource and agents in G_2 get $1-(1-\alpha)s$ of the first resource. Notice that under DRF every agents has $\frac{s}{n}$ dominant share while in **A** every agent gets at least $\frac{1}{n}$ dominant share. Thus, in **A** agents in G_2 get at least $\frac{1}{s}(1-(1-\alpha)s)$ of the first resource. Then SW of **A** is upper bounded by

$$1-\frac{1}{s}(1-(1-\alpha)s)+1 = 3-\alpha-\frac{1}{s} \leq (2-\alpha)s,$$

where the last inequality follows by $s \geq 1$. Analogously, if the second resource is used up under DRF, we have that SW of **A** is upper bounded by $3-(1-\alpha)-\frac{1}{s} \leq (2-\alpha)s$.

We then show that fair-ratio for utilization is $\frac{1}{\alpha}$. For the upper bound, since DRF satisfies SI, each resource is used at least α, so fair-ratio for utilization is upper bounded by $\frac{1}{\alpha}$. For the lower bound, using the same instance used above for SW, we have that in the allocation under DRF the second resource is not used up and at most $\frac{n}{n-2}\alpha \leq \alpha + \frac{2}{n-2}$ of the second resource is used. However, in $\mathbf{A}^*$ the first resource is not used up and at least $1-\alpha+(n\alpha-1)\frac{1-\frac{1}{n}}{n} \geq 1-\frac{2}{n}$ of the first resource is used. So fair-ratio for utilization of DRF is lower bounded by

$$\frac{1-\frac{2}{n}}{\alpha+\frac{2}{n-2}} \xrightarrow{n\to\infty} \frac{1}{\alpha}.$$

7.6.2 Proof of Theorem 7.2

To formally prove this theorem, we solve this by proving the following four lemmas.

Lemma 7.7. *For the situation with* 2 *resources, mechanism* UNB *satisfies SI, EF, PO, and SP.*

Proof. We first show UNB satisfies SI, EF and PO. SI and PO are clearly satisfied since all agents have dominant share at least $\frac{1}{n}$ and the mechanism stops only when one resource is used up. EF is satisfied in the first step when all agents get the same dominant share $\frac{1}{n}$. After that, the mechanism allocates more resources to agents in G_2, so there is no envy from G_2 to G_1. Note that the mechanism stops before any agent in G_2 receiving more than $\frac{1}{n}$ of resource 1. Since all agents in G_1 have $\frac{1}{n}$ of resource 1, there is no envy from G_1 to G_2. Within G_2, the mechanism only allocate resources to agents who have the smallest fraction of resource 1, so no envy will occur.

It remains to show SP is satisfied. It is easy to see that no agent has an incentive to change the group they belong to as in the final allocation the fraction of the non-dominant resource is at most $\frac{1}{n}$ for all agents in both G_1 and G_2. In addition, agents in G_1 always get $\frac{1}{n}$ of the first resource, so agents in G_1 have no incentive to lie. Finally, we consider agents in G_2. Suppose that there exists an agent $i_0 \in G_2$ who can benefit by reporting a false demand vector. Let x_i be the fraction of the first resource of agent $i \in G_2$ in the truthful outcome and denote $x^* = \min_{i \in G_2} x_i$. Let $\overline{x_i}$ be the fraction of the first resource of agent $i \in G_2$ in the manipulated outcome and denote $\overline{x^*} = \min_{i \in G_2} \overline{x_i}$. We have that agent i_0 receives more of both resources in the manipulated outcome, so $x_{i_0} < \overline{x_{i_0}}$. Since agent i_0 receives more dominant resource in the manipulated outcome, i_0 must be one of agents who have the smallest fraction of the first resource in the manipulated outcome. Thereby, $\overline{x_{i_0}} = \overline{x^*}$ and then $x^* \leq x_{i_0} < \overline{x_{i_0}} = \overline{x^*}$. However, this means that agent i_0 receives more of both resources in the manipulated outcome while all other agents receive at least the same amount of resources, which contradicts that the truthful allocation is PO. This finished the proof for SP. □

Lemma 7.8. *For the situation with* 2 *resources, the fair-ratio of* UNB *is upper bounded by* $1 + \alpha \leq \frac{3}{2}$ *for SW and upper bounded by* $\frac{1}{1-\alpha}$ *for utilization.*

Proof. We first study the fair-ratio for SW. We distinguish the case where the first resource is used up and where the second resource is used up under UNB. For the first case, we show the allocation $\mathbf{A}$ of UNB actually maximizes SW subject to SI, EF, and PO. For any allocation $\mathbf{A}'$ satisfying SI, EF, and PO, let S_1' be the sum of the first resource received by agents in G_1. Since $\mathbf{A}'$ satisfying SI, we have $S_1' \geq \frac{n_1}{n}$. If $S_1' > \frac{n_1}{n}$, then the sum of dominant shares of agents in G_1 is increased by $S_1' - \frac{n_1}{n}$ compared with $\mathbf{A}$, while the sum of dominant shares of agents in G_2 is decreased by more than $S'_1 - \frac{n_1}{n}$ since the demand vectors of agents in G_2 satisfy that $d_{i,1} < d_{i,2}$. Then $\mathrm{SW}(\mathbf{A}') \leq \mathrm{SW}(\mathbf{A})$. If $S'_1 = \frac{n_1}{n}$, then all agents in G_2 must receive $\frac{1}{n}$ of the first resource, i.e., $\mathbf{A} = \mathbf{A}'$, since otherwise at least one agent in G_2 receives more than $\frac{1}{n}$ of the first resource and the second resource (due to the demand vector) and then this agent will be envied by all agents in G_1. Therefore, allocation $\mathbf{A}$ maximizes SW subject to SI, EF, and PO.

For the second case where the second resource is used up, let S_2 be the sum of dominant shares of agents in G_2 in $\mathbf{A}$. Then $\mathrm{SW}(\mathbf{A}) = \frac{n_1}{n} + S_2$ and $S_2 \geq 1 - \frac{n_1}{n}$. For any allocation satisfying SI, all agents in G_1 receive at least the same amount of receives as in $\mathbf{A}$, so the sum of dominant shares of agents in G_2 is at most S_2 and the maximum SW of allocations satisfying SI of this instance is upper bounded by $1 + S_2$. Then the fair-ratio for SW of UNB is upper bounded by

$$\frac{1 + S_2}{\frac{n_1}{n} + S_2} \leq \frac{1 + 1 - \frac{n_1}{n}}{\frac{n_1}{n} + 1 - \frac{n_1}{n}} = 1 + \alpha \leq \frac{3}{2},$$

where the first inequality follows from $S_2 \geq 1 - \frac{n_1}{n}$ and the second $\alpha \leq \frac{1}{2}$.

We then study the fair-ratio for utilization. We also distinguish the two cases as above. For the first case where resource 1 is used up under UNB, we show that the allocation $\mathbf{A}$ of UNB actually maximizes utilization subject to SI, EF, and PO. For any allocation $\mathbf{A}'$ satisfying SI, EF, and PO, let $S'_1 \geq \frac{n_1}{n}$ be the sum of the first resource received by agents in G_1. Notice that the most efficient way (maximizing the use of the second resource 2) of using a fixed amount of resource 1 for agents in G_2 subject to EF is to evenly distribute resource 1 among agents in G_2. If $S'_1 > \frac{n_1}{n}$, since the dominant resource of agents in G_1 is resource 1 and the dominant resource of agents in G_2 is resource 2, the utilization of resource 2 must be decreased. If $S'_1 = \frac{n_1}{n}$, then all agents in G_2 must receive $\frac{1}{n}$ of resource 1 due to EF, i.e., $\mathbf{A} = \mathbf{A}'$. Therefore, allocation $\mathbf{A}$ maximizes utilization subject to SI, EF, and PO. For the second case, since resource 2 is used up and UNB satisfies SI, at least $1 - \alpha$ of resource 1 is used, so the fair-ratio for utilization of UNB is upper bounded by $\frac{1}{1-\alpha}$. □

Lemma 7.9. *For the situation with* 2 *resources, the fair-ratio of* UNB *is lower bounded by* $1 + \alpha$ *for SW and lower bounded by* $\frac{1}{1-\alpha}$ *for utilization.*

Proof. We first study the fair-ratio for SW. We build an instance with minority population ratio α and n agents as follows. The first group G_1 consists of $n(1-\alpha)$ agents, where except for the first agent whose demand vector is $(1, \varepsilon)$, all other agents have the same demand vector $(1, 1-\varepsilon)$, where $\varepsilon = \frac{1}{n}$. The second group G_2 consists of $n\alpha$ agents who have the same demand vector $(\varepsilon, 1)$. Under UNB, in step 2 we will increase the allocations of agents in G_2. However, the remaining amount of resource 2 after step 1 is only $1-\alpha-\frac{n(1-\alpha)-1}{n}(1-\varepsilon)-\frac{\varepsilon}{n} \leq \frac{2}{n}$. So the SW under UNB is upper bounded by $\frac{n+2}{n}$. On the other hand, if we give $\frac{1}{n}$ dominant share to every agent except for the first agent and give the first agent the bundle $(\frac{n-1}{n}\alpha, \frac{n-1}{n^2}\alpha)$ such that the first resource is used up, then the SW is $1+\frac{n-1}{n}\alpha$. It is easy to verify that the above allocation, denoted by $\mathbf{A}^*$, satisfies SI, EF, and PO. For EF, notice that the first agent receives $\frac{n-1}{n^2}\alpha < \frac{1}{2n}$ of the second resource while all other agents receive at least $\frac{1}{2n}$ of the second resource. So the fair-ratio for SW of DRF is at least

$$\frac{1+\frac{n-1}{n}\alpha}{\frac{n+2}{n}} \xrightarrow{n\to\infty} 1+\alpha.$$

For the fair-ratio for utilization, we use the same instance as above. Under UNB, at most $1-\alpha+\frac{\alpha}{n}+\frac{2}{n} \leq 1-\alpha+\frac{3}{n}$ of resource 1 is used as the SW under UNB is upper bounded by $\frac{n+2}{n}$. In $\mathbf{A}^*$, resource 1 is used up and at least $1-\frac{2}{n}$ of resource 2 is used, so the fair-ratio for SW of DRF is at least

$$\frac{1-\frac{2}{n}}{1-\alpha+\frac{3}{n}} \xrightarrow{n\to\infty} \frac{1}{1-\alpha}.$$

□

Lemma 7.10. *For the situation with 2 resources,* UNB *can be implemented in polynomial time. Specifically, mechanism* UNB *can be implemented in $O(n^2)$ time and $O(n)$ space.*

Proof. Notice that $|P|$ is increasing in each round of step 2, so the number of rounds of step 2 is at most n. Since each round of step 2 can be implemented in $O(n)$ time, UNB can be implemented in $O(n^2)$ time. Since only the amount of each resource allocated to each agent needs to be stored, the space complexity is $O(n)$. □

7.6.3 Proof of Theorem 7.3

For the formal proof of this theorem, we will only prove the upper bound for the fair-ratio here, and the proof of the complexity, the lower bound and that BAL satisfies SI, EF and PO will be deferred to the proof of Theorem 7.4, where we show these two results for both BAL and BAL*.

Note that $R_1 \leq \alpha \leq 1-\alpha$ and $R_2 \leq 1-\alpha$. If $R_1 = 0$ or $R_2 = 0$, then all agents have the same dominant share $\frac{1}{n}$ and the allocation of BAL is already optimal. So in the following analysis we assume $R_1 > 0$ and $R_2 > 0$, which also implies that $\Delta S_1 > 0$ and $\Delta S_2 > 0$. Denote the allocation under BAL by $\mathbf{A}$ and the SW-maximizing allocation satisfying SI, EF, and PO by $\mathbf{A}^*$. Denote the sum of dominant shares of agents in G_1 and G_2 in $\mathbf{A}^*$ by $(1-\alpha) + \Delta S_1^*$ and $\alpha + \Delta S_2^*$, respectively. Here, we suppose $\mathrm{SW}(\mathbf{A}^*) > \mathrm{SW}(\mathbf{A})$ without loss of generality.

Lemma 7.11. *For any $k \in \{1, 2\}$, if resource k is used up in $\mathbf{A}$, then $\Delta S_k^* \leq \Delta S_k$. Consequently, we have that either $\Delta S_1^* \leq \Delta S_1$, or $\Delta S_2^* \leq \Delta S_2$.*

Proof. Due to symmetry, it suffices to show the result for $k = 1$. Suppose that resource 1 is used up in $\mathbf{A}$. Denote the sum of resource 1 received by agents in G_2 in $\mathbf{A}$ and $\mathbf{A}^*$ by x and x^* respectively. Notice that at step 2 of BAL we always increase allocations of agents in G_2 with the smallest $A_{i,1}$ subject to EF, which is the most efficient way of using x of resource 1 to maximize the received amount of resource 2. Therefore, for any other allocation satisfying SI and EF within G_2, if agents in G_2 receive at most x of resource 1, then they can receive at most $\alpha + \Delta S_2$ of resource 2. We use this property to show that $\Delta S_1^* \leq \Delta S_1$. The intuition is that if $\Delta S_1^* > \Delta S_1$, then for $\mathbf{A}^*$ compared with $\mathbf{A}$, the decreased dominant share of agents in G_2 will exceed the increased dominant share of agents in G_1.

Suppose towards a contradiction that $\Delta S_1^* > \Delta S_1$ and let $\delta = \Delta S_1^* - \Delta S_1 > 0$. We prepare some inequalities. First, since resource 1 is used up in $\mathbf{A}$, we have that

$$x^* \leq x - \delta. \tag{7.3}$$

Next, since $d_{i,1} < 1$ for $i \in G_2$, in $\mathbf{A}^*$ the amount of resource 1 received by agents in G_2 is less than the amount of resource 2, that is,

$$x^* < \alpha + \Delta S_2^*. \tag{7.4}$$

Finally, since $\mathrm{SW}(\mathbf{A}^*) = 1 + \Delta S_1^* + \Delta S_2^* > \mathrm{SW}(\mathbf{A}) = 1 + \Delta S_1 + \Delta S_2$, we have $\Delta S_2 - \Delta S_2^* < \Delta S_1^* - \Delta S_1 = \delta$, that is,

$$\Delta S_2^* + \delta > \Delta S_2. \tag{7.5}$$

Now, based on $\mathbf{A}^*$, we construct a partial allocation $\mathbf{A}'$ for G_2 by multiplying each $\mathbf{A}_i^*$ by γ for all $i \in G_2$, where $\gamma = \frac{x}{x^*}$. In $\mathbf{A}'$ every agent in G_2 has at least $\frac{1}{n}$ dominant share since $\mathbf{A}^*$ satisfies SI and the allocations of agents in G_2 are increased from $\mathbf{A}^*$ to $\mathbf{A}'$. EF is also satisfied within G_2 since $\mathbf{A}^*$ satisfies EF and the allocations of agents in G_2 are multiplied by the same factor γ to get $\mathbf{A}'$. However, in $\mathbf{A}'$ the sum of resource 2 received by agents in G_2 is

$$\gamma(\alpha+\Delta S_2^*) = \frac{x}{x^*}(\alpha+\Delta S_2^*) \overset{(7.3)}{\geq} \alpha+\Delta S_2^*+\frac{\delta}{x^*}(\alpha+\Delta S_2^*) \overset{(7.4)}{>} \alpha+\Delta S_2^*+\delta \overset{(7.5)}{>} \alpha+\Delta S_2.$$

Then, in $\mathbf{A}'$ agents in G_2 have x of resource 1 but more than $\alpha + \Delta S_2$ of resource 2, which is a contradiction. Therefore, $\Delta S_1^* \leq \Delta S_1$.

Since $\mathbf{A}$ is PO, at least one resource is used up, then we have that either $\Delta S_1^* \leq \Delta S_1$, or $\Delta S_2^* \leq \Delta S_2$. □

Lemma 7.12. *For the situation with 2 resources, fair-ratio for SW of* BAL *is upper bounded by* $\frac{2-\alpha}{1+\frac{1-\alpha}{2}} \leq \frac{4}{3}$.

Proof. Let $\beta = \frac{\Delta S_1}{R_1} = \frac{\Delta S_2}{R_2}$. We distinguish two cases when $\beta \geq \frac{1}{2}$ and when $\beta < \frac{1}{2}$. For the first case, we have that $\Delta S_i \geq \frac{1}{2}R_i \geq \frac{1}{2}\Delta S_i^*$ for $i = 1, 2$. Therefore,

$$\frac{1 + \Delta S_1^* + \Delta S_2^*}{1 + \Delta S_1 + \Delta S_2} \leq \frac{1 + \Delta S_1^* + \Delta S_2^*}{1 + \frac{1}{2}(\Delta S_1^* + \Delta S_2^*)} \leq \frac{1+1}{1+\frac{1}{2}} \leq \frac{4}{3},$$

where the second to last inequality is due to $\Delta S_1^* + \Delta S_2^* \leq R_1 + R_2 \leq 1$. To get a more refined upper bound as a function of α, notice that according to Lemma 7.11, either $\Delta S_1^* \leq \Delta S_1$, or $\Delta S_2^* \leq \Delta S_2$. Then,

$$\frac{1 + \Delta S_1^* + \Delta S_2^*}{1 + \Delta S_1 + \Delta S_2} \leq \max_{i\in\{1,2\}} \frac{1 + \Delta S_i^*}{1 + \Delta S_i} \leq \max_{i\in\{1,2\}} \frac{1 + \Delta S_i^*}{1 + \frac{1}{2}\Delta S_i^*} \leq \frac{2-\alpha}{1 + \frac{1-\alpha}{2}},$$

where the last inequality is due to $\max\{\Delta S_1^*, \Delta S_2^*\} \leq \max\{R_1, R_2\} \leq 1 - \alpha$.

For the second case when $\beta < \frac{1}{2}$, let $i_1 \in \{1, 2\}$ be the resource that is used up in $\mathbf{A}$ and let $i_2 = 3 - i_1$ represents another resource. We show that $\Delta S_{i_2}^* \leq 2\Delta S_{i_2}$. Since resource i_1 is used up and $\Delta S_{i_1} < \frac{R_{i_1}}{2}$, agents in G_{i_2} use more than $\frac{R_{i_1}}{2}$ of resource i_1 in step 2 of BAL. Since the average d_{i,i_1} of agents in G_{i_2} with the smallest A_{i,i_1} is increasing in step 2 of BAL, or equivalently D_k in Algorithm 7 is decreasing, even if we allocate all R_{i_1} of resource i_1 to agents in G_2, they can use at most $2\Delta S_{i_2}$ of resource i_2. So $\Delta S^*_{i_2} \leq 2\Delta S_{i_2}$. Combining $\Delta S^*_{i_1} \leq \Delta S_{i_1}$ from Lemma 7.11 we have that

$$\frac{1+\Delta S^*_{i_1}+\Delta S^*_{i_2}}{1+\Delta S_{i_1}+\Delta S_{i_2}} \leq \frac{1+\Delta S^*_{i_2}}{1+\Delta S_{i_2}} \leq \frac{1+\Delta S^*_{i_2}}{1+\frac{\Delta S^*_{i_2}}{2}} \leq \frac{2-\alpha}{1+\frac{1-\alpha}{2}} \leq \frac{4}{3},$$

where the second to the last inequality is due to $\Delta S^*_{i_2} \leq \max\{R_1, R_2\} \leq 1-\alpha$. Therefore, the fair-ratio for SW of BAL is upper bounded by $\frac{2-\alpha}{1+\frac{1-\alpha}{2}}$, which reaches its maximum value $\frac{4}{3}$ when $\alpha = 0$. □

Lemma 7.13. *For the situation with* 2 *resources, the fair-ratio for utilization of* BAL *is upper bounded by* $\frac{1}{1-\frac{1-\alpha}{2}} \leq 2$.

Proof. Let $\beta = \frac{\Delta S_1}{R_1} = \frac{\Delta S_2}{R_2}$. We distinguish two cases when $\beta \geq \frac{1}{2}$ and when $\beta < \frac{1}{2}$. For the first case, we have that $\Delta S_i \geq \frac{1}{2}R_i$ for $i = 1, 2$. Then at least $1 - \frac{R_1}{2} \geq 1 - \frac{\alpha}{2} \geq 1 - \frac{1-\alpha}{2}$ of resource 1 is used and at least $1 - \frac{R_2}{2} \geq 1 - \frac{1-\alpha}{2}$ of resource 2 is used. Therefore, the fair-ratio for utilization of BAL is upper bounded by $\frac{1}{1-\frac{1-\alpha}{2}}$.

For the second case when $\beta < \frac{1}{2}$, we further distinguish two cases when resource 1 is used up and when resource 2 is used up in $\mathbf{A}$. We now use $\mathbf{A}^*$ to represent the utilization-maximizing allocation satisfying SI, EF, and PO. Following the same use of notations, in $\mathbf{A}^*$ agents in G_1 receive $1-\alpha+\Delta S^*_1$ of resource 1 and $\alpha+\Delta S^*_2$ of resource 2. Let $y_1 = \sum_{i\in G_1} \frac{d_{i,2}}{n}$ be the amount of resource 2 received by agents in G_1 and $y_2 = \sum_{i\in G_2} \frac{d_{i,1}}{n}$ be the amount of resource 1 received by agents in G_2 when every agent receives $\frac{1}{n}$ dominant share.

If resource 1 is used up, as shown in the proof for SW (Lemma 7.12), we have that $\Delta S^*_2 \leq 2\Delta S_2$. Let y^*_1 be the mount of resource 2 received by G_1 in $\mathbf{A}^*$. Notice that the most efficient way of using resource 1 to maximize the received amount of resource 2 for G_1 subject to EF is to evenly distribute resource 1. Since agents in G_1 use $1-\alpha$ of resource 1 when every agent gets $\frac{1}{n}$ dominant share and $1-\alpha \geq \frac{1}{2}$, we have $y^*_1 \leq 2y_1$. Therefore, the fair-ratio for utilization (determined by resource

2) is upper bounded by

$$\frac{\alpha + 2\Delta S_2 + y_1^*}{\alpha + \Delta S_2 + y_1} \leq \frac{1}{\alpha + \frac{1-\alpha}{2}} = \frac{1}{1 - \frac{1-\alpha}{2}},$$

where the inequality is due to $2\Delta S_2 + y_1^* \leq 1 - \alpha$ and $\Delta S_2 + y_1 \geq \frac{1}{2}(2\Delta S_2 + y_1^*)$.

If resource 2 is used up, as shown in the proof for SW, we have that $\Delta S_1^* \leq 2\Delta S_1$. Let y_2^* be the mount of resource 1 received by G_2 in $\mathbf{A}^*$. Next we need to upper bound y_2^*. Since $\frac{\Delta S_2}{R_2} = \beta < \frac{1}{2}$ and resource 2 is used up, in $\mathbf{A}$ agents in G_1 receive more than $\frac{R_2}{2}$ of resource 2 while they receive less than $\frac{R_1}{2}$ of their dominant resource 1 as $\beta < \frac{1}{2}$. Then from $\frac{R_1}{2} \geq \frac{R_2}{2}$ we have $R_2 \leq R_1 \leq \alpha$. It follows that $\Delta S_2^* \leq \frac{R_2}{2} \leq \frac{\alpha}{2}$. Since agents in G_2 use α of resource 2 when every agent gets $\frac{1}{n}$ dominant share, we have $y_2^* \leq \frac{3y_2}{2}$. Therefore, the fair-ratio for utilization (determined by resource 1) is upper bounded by

$$\frac{1 - \alpha + 2\Delta S_2 + y_2^*}{1 - \alpha + \Delta S_2 + y_2} \leq \frac{1}{1 - \frac{\alpha}{2}} \leq \frac{1}{1 - \frac{1-\alpha}{2}},$$

where the first inequality is due to $2\Delta S_2 + y_2^* \leq \alpha$ and $\Delta S_2 + y_2 \geq \frac{1}{2}(2\Delta S_2 + y_2^*)$. □

We can finish the proof by summarizing the two lemmas above.

7.6.4 Proof of Theorem 7.4

Lemma 7.14. BAL *and* BAL* *satisfy SI, EF, and PO.*

Proof. The proof is very similar to the proof for UNB. SI and PO are clearly satisfied since all agents have dominant share at least $\frac{1}{n}$ and the mechanism stops only when one resource is used up. EF is satisfied in step 1 as all agents have the same dominant share $\frac{1}{n}$. Note that the mechanism stops before any agent receiving more than $\frac{1}{n}$ of the non-dominant resource while all agents have at least $\frac{1}{n}$ of the dominant resource, so there is no envy between G_1 and G_2. Within each group, in step 2 the mechanism only allocates resources to agents who have the smallest fraction of the non-dominant resource, so no envy will occur. □

Lemma 7.15. BAL* *satisfies SP.*

Proof. Same as UNB, no agent has an incentive to change the group they belong to as the fraction of the non-dominant resource cannot exceed $\frac{1}{n}$. Then, we show that agents in G_1 can never benefit by reporting a false demand vector. The proof for agents in G_2 is analogous due to symmetry. Suppose some agent $i \in G_1$, whose true demand vector is $(1, d_{i,2})$, reports a false demand vector $(1, d'_{i,2})$. Let $\Delta S'_1$ and $\Delta S'_2$ be the sum of increased dominant shares of agents in G_1 and G_2, respectively, at step 2 of BAL* when i reports $(1, d'_{i,2})$. We distinguish two cases when $\frac{\Delta S'_1}{\Delta S'_2} \leq \frac{\Delta S_1}{\Delta S_2}$ and $\frac{\Delta S'_1}{\Delta S'_2} > \frac{\Delta S_1}{\Delta S_2}$.

For the first case, suppose, for a contradiction, that agent i's utility is increased by reporting $(1, d'_{i,2})$. Then the allocation of both resources to agent i is strictly increased. Specifically, agent i receives more amount of resource 2. According to step 2 of BAL*, all other agents in G_1 should receive at least the same of resource 2 and resource 1. To sum up, agents in G_1 receive more of resource 1 and resource 2, then $\Delta S'_1 > \Delta S_1$ and consequently, $\Delta S'_2 < \Delta S_2$, which contradicts with $\frac{\Delta S'_1}{\Delta S'_2} \leq \frac{\Delta S_1}{\Delta S_2}$.

We proceed by considering the second case when $\frac{\Delta S'_1}{\Delta S'_2} > \frac{\Delta S_1}{\Delta S_2}$. According to condition (7.2), $\frac{\Delta S'_1}{\Delta S'_2} > \frac{\Delta S_1}{\Delta S_2}$ implies $d'_{i,2} > d_{i,2}$ and agent i is not the agent in G_1 with the minimum $d_{i,2}$ when it reports $(1, d'_{i,2})$, as otherwise R^*_2 does not change and we should have $\frac{\Delta S'_1}{\Delta S'_2} = \frac{\Delta S_1}{\Delta S_2}$. Let $\delta = \frac{1}{n}(d'_{i,2} - \max\{d_{i,2}, \min_{j \in G_1 - i} d_{j,2}\})$ be the influence on R^*_2 when agent i reports $(1, d'_{i,2})$. For the manipulated instance, we have

$$\frac{\Delta S'_1}{\Delta S'_2} = \frac{R^{*\prime}_1}{R^{*\prime}_2} = \frac{R^*_1}{R^*_2 - \delta}, \tag{7.6}$$

On the other hand, we have

$$\frac{\Delta S_1}{R^*_1} = \frac{\Delta S_2}{R^*_2} > \frac{\Delta S_2 - \delta}{R^*_2 - \delta} \Rightarrow \frac{R^*_2 - \delta}{R^*_1} > \frac{\Delta S_2 - \delta}{\Delta S_1}. \tag{7.7}$$

Combining Equations 7.6 and 7.7, we have

$$\frac{\Delta S'_2}{\Delta S'_1} > \frac{\Delta S_2 - \delta}{\Delta S_1}. \tag{7.8}$$

Now suppose that agent i's utility is increased by reporting $(1, d'_{i,2})$. We show in the following that $\Delta S'_1 \geq \Delta S_1$ and $\Delta S'_2 \leq \Delta S_2 - \delta$, which contradicts with Equation 7.8.

Let $\mathbf{A}$ be the truthful allocation and let $\mathbf{A}'$ be the manipulated allocation. Since agent i's utility is increased by reporting $(1, d'_{i,2})$ with $d'_{i,2} = \max\{d_{i,2}, \min_{j\in G_1-i} d_{j,2}\}+ n\delta$, we have $A'_{i,1} > A_{i,1} \geq \frac{1}{n}$ and

$$A'_{i,2} = A'_{i,1} \cdot d'_{i,2} > A_{i,1} \cdot (d_{i,2} + n\delta) \geq A_{i,2} + \delta.$$

Let x be the amount of resource 2 received by agents in G_1 who have the minimum amount of resource 2 in the truthful allocation, and let x' be that in the manipulated allocation. Since agent i receives more amount of resource 2 in $\mathbf{A}'$, it must be one of the agents in G_1 who have the minimum amount of resource 2 in $\mathbf{A}'$. So $x' = A'_{i,2}$. For $\mathbf{A}$ we have $x \leq A_{i,2}$. Therefore, $x' \geq x + \delta$, which implies that $A'_{j,1} \geq A_{j,1}$ and $A'_{j,2} \geq A_{j,2}$ for all $j \in G_1 - i$.

Let $j^* \in G_1$ be one of the agents in G_1 with the minimum $d_{j,2}$ when i reports $(1, d'_{i,2})$. Recall that $j^* \neq i$. We can prove $A'_{j^*,2} - A_{j^*,2} \geq \delta$, and then

$$A'_{j^*,1} - A_{j^*,1} \geq \delta$$

as $d_{j^*,1} = 1 \geq d_{j^*,2}$. First, we have j^* must be one of the agents in G_1 who have the minimum amount of resource 2 in $\mathbf{A}'$. Then, we consider two case when $A_{j^*,2} = x$ and $A_{j^*,2} > x$. In the first case $A_{j^*,2} = x$, it can be proved by $A'_{j^*,2} = x' \geq x + \delta = A_{j^*,2} + \delta$. In the second case, j^* still receives $\frac{1}{n}$ of resource 1 in $\mathbf{A}$, so

$$A'_{j^*,2} = x' = A'_{i,2} = A'_{i,1} d'_{i,2} \geq A'_{i,1}(d_{j^*,2} + n\delta) \geq \frac{1}{n}(d_{j^*,2} + n\delta) = A_{j^*,2} + \delta.$$

To sum up, all agents in G_1 receive more resources in $\mathbf{A}'$ and specifically we have $A'_{i,2} \geq A_{i,2} + \delta$ and $A'_{j^*,1} - A_{j^*,1} \geq \delta$, so

$$\sum_{j\in G_1} (A'_{j,1} - A_{j,1}) \geq \delta \text{ and } \sum_{j\in G_1} (A'_{j,2} - A_{j,2}) \geq \delta.$$

From $\sum_{j\in G_1}(A'_{j,1} - A_{j,1}) \geq \delta$ we get $\Delta S'_1 \geq \Delta S_1 + \delta \geq \Delta S_1$, as required. Next we show $\Delta S'_2 \leq \Delta S_2 - \delta$ for G_2. If in $\mathbf{A}$ resource 2 is used up, then $\Delta S'_2 \leq \Delta S_2 - \delta$ because $\sum_{j\in G_1}(A'_{j,2} - A_{j,2}) \geq \delta$. If in $\mathbf{A}$ resource 1 is used up, then according to $\sum_{j\in G_1}(A'_{j,1} - A_{j,1}) \geq \delta$, we have $\sum_{j\in G_2} A'_{j,1} \leq \sum_{j\in G_2} A_{j,1} - \delta$, and hence $\Delta S'_2 \leq \Delta S_2 - \delta$ as $d_{j,1} \leq d_{j,2} = 1$ for $j \in G_2$. Therefore, we have $\Delta S'_2 \leq \Delta S_2 - \delta$, no matter which resource is used up in $\mathbf{A}$. □

As for the fair-ratio, we show that SW of BAL* is very close to BAL.

Lemma 7.16. *For any instance* $\mathbf{I}$ *with* n *agents and* $m = 2$ *resources, we have*

$$\frac{SW(\mathrm{BAL}(\mathbf{I}))}{SW(\mathrm{BAL}^*(\mathbf{I}))} \le 1 + \frac{1}{n} \text{ and } U(\mathrm{BAL}(\mathbf{I})) \le U(\mathrm{BAL}^*(\mathbf{I})) + \frac{1}{n}.$$

Proof. We first study SW. Denote the sum of increased dominant shares of agents in G_1 and G_2 in step 2 of BAL* by ΔS_1^* and ΔS_2^*, respectively. Let $R_1^* = R_1 + \frac{1}{n} d_{i^*,1}$ and $R_2^* = R_2 + \frac{1}{n} d_{j^*,2}$, where $i^* \in G_2$ and $j^* \in G_1$ are defined in the same way as in condition (7.2), then

$$\frac{\Delta S_1}{\Delta S_2} = \frac{R_1}{R_2} \text{ and } \frac{\Delta S_1^*}{\Delta S_2^*} = \frac{R_1^*}{R_2^*}.$$

Assume without loss of generality that $\Delta S_1 \ge \Delta S_1^*$. Then $\Delta S_2 \le \Delta S_2^*$ since the only difference between BAL and BAL* is the different increasing speeds for two groups at step 2. Since

$$\frac{\mathrm{SW}(\mathrm{BAL}(\mathbf{I}))}{\mathrm{SW}(\mathrm{BAL}^*(\mathbf{I}))} = \frac{1 + \Delta S_1 + \Delta S_2}{1 + \Delta S_1^* + \Delta S_2^*} \le \frac{1 + \Delta S_1}{1 + \Delta S_1^*},$$

it suffices to show that

$$\frac{1 + \Delta S_1}{1 + \Delta S_1^*} \le 1 + \frac{1}{n}.$$

Let

$$\beta = \frac{\Delta S_1}{R_1} = \frac{\Delta S_2}{R_2} \text{ and } \beta^* = \frac{\Delta S_1^*}{R_1^*} = \frac{\Delta S_2^*}{R_2^*}.$$

Since $\Delta S_1 \ge \Delta S_1^*$ and $R_1 < R_1^* = R_1 + \frac{1}{n} d_{i^*,1}$, we have $\beta > \beta^*$. Since

$$\frac{\Delta S_2^* - \frac{1}{n} d_{j^*,2}}{R_2} = \frac{\Delta S_2^* - \frac{1}{n} d_{j^*,2}}{R_2^* - \frac{1}{n} d_{j^*,2}} < \beta^* < \beta = \frac{\Delta S_2}{R_2},$$

we have $\Delta S_2 > \Delta S_2^* - \frac{1}{n} d_{j^*,2}$. Therefore, compared with BAL*, the sum of resource 2 received by agents in G_2 is decreased by at most $\frac{1}{n} d_{j^*,2}$ under BAL, i.e.,

$$\sum_{i \in G_2} A_{i,2} > \sum_{i \in G_2} A_{i,2}^* - \frac{1}{n} d_{j^*,2}. \tag{7.9}$$

Since the dominant resource for agents in G_2 is resource 2, the sum of resource 1 received by agents in G_2 is also decreased by at most $\frac{1}{n}d_{j^*,2}$ under BAL, i.e.,

$$\sum_{i \in G_2} A_{i,1} > \sum_{i \in G_2} A^*_{i,1} - \frac{1}{n} d_{j^*,2}. \tag{7.10}$$

If resource 1 is used up in BAL, by (7.10) we have $\Delta S_1 \leq \Delta S_1^* + \frac{1}{n}d_{j^*,2} \leq \Delta S_1^* + \frac{1}{n}$. If resource 2 is used up in BAL, by (7.9) we have

$$\sum_{i \in G_1} A_{i,2} < \sum_{i \in G_1} A^*_{i,2} + \frac{1}{n} d_{j^*,2}.$$

Since $j^* \in G_1$ is the agent in G_1 with the minimum $d_{i,2}$, with the increased $\frac{1}{n}d_{j^*,2}$ of resource 2, resource 1 can be increased by at most $\frac{1}{n}d_{j^*,2} \cdot \frac{1}{d_{j^*,2}} = \frac{1}{n}$. So $\Delta S_1 \leq \Delta S_1^* + \frac{1}{n}$. In both cases we have $\Delta S_1 \leq \Delta S_1^* + \frac{1}{n}$. So

$$\frac{1 + \Delta S_1}{1 + \Delta S_1^*} \leq \frac{1 + \Delta S_1^* + \frac{1}{n}}{1 + \Delta S_1^*} \leq 1 + \frac{1}{n},$$

as required.

We then study utilization. Assume without loss of generality that $\Delta S_1 \geq \Delta S_1^*$, i.e., agents in G_1 receive more amounts of both resources in BAL and agents in G_2 receive less amounts of both resources in BAL. In the above we have already shown that $\Delta S_1 \leq \Delta S_1^* + \frac{1}{n}$ fro resource 1 and $\sum_{i \in G_1} A_{i,2} < \sum_{i \in G_1} A^*_{i,2} + \frac{1}{n}d_{j^*,2}$ for resource 2, so the utilization of both resources is increased by at most $\frac{1}{n}$ under BAL. Therefore, $U(\text{BAL}(\mathbf{I})) \leq U(\text{BAL}^*(\mathbf{I})) + \frac{1}{n}$. □

The upper bound for the fair-ratio of BAL* can be improved if we consider it directly.

Lemma 7.17. *For the situation with* 2 *resources, fair-ratio of* BAL* *is upper bounded by* $\frac{2-\alpha}{1+\frac{1-\alpha}{2}-\frac{1}{2n}}$ *for SW and upper bounded by* $\frac{1}{1-\frac{1-\alpha}{2}-\frac{1}{2n}}$ *for utilization.*

Proof. The proof is similar to the proof for BAL (Lemma 7.12 and 7.13). In this proof, denote by $\mathbf{A}$ the allocation of BAL* and $\mathbf{A}^*$ the SW maximization (or utilization maximization) satisfying SI, EF, and PO. Denote in the step 2 of BAL* the sum of dominant shares of agents in G_1 and G_2 in $\mathbf{A}^*$ by ΔS_1 and ΔS_2, respectively. Let R_1 and R_2 be the remaining amount of two resources after the step 1 of BAL*. Let $R_1^* = R_1 + \frac{1}{n}d_{i^*,1}$ and $R_2^* = R_2 + \frac{1}{n}d_{j^*,2}$, where $i^* \in G_2$ is the agent in G_2 with the minimum $d_{i,1}$ and $j^* \in G_1$ is the agent in G_1 with the minimum $d_{i,2}$. Denote the sum of dominant shares of agents in G_1 and G_2 in $\mathbf{A}^*$ by $(1-\alpha) + \Delta S_1^*$ and $\alpha + \Delta S_2^*$, respectively.

We first consider SW. Let $\beta = \frac{\Delta S_1}{R_1^*} = \frac{\Delta S_2}{R_2^*}$. We distinguish two cases when $\beta \geq \frac{1}{2}$ and when $\beta < \frac{1}{2}$. For the first case, we have that $\Delta S_i \geq \frac{1}{2}R_i^* \geq \frac{1}{2}R_i \geq \frac{1}{2}\Delta S_i^*$ for $i = 1, 2$. Notice that the result in Lemma 7.11 also holds for the allocation $\mathbf{A}$ by BAL*, i.e., either $\Delta S_1^* \leq \Delta S_1$, or $\Delta S_2^* \leq \Delta S_2$. Therefore,

$$\frac{1 + \Delta S_1^* + \Delta S_2^*}{1 + \Delta S_1 + \Delta S_2} \leq \max_{i\in\{1,2\}} \frac{1 + \Delta S_i^*}{1 + \Delta S_i} \leq \max_{i\in\{1,2\}} \frac{1 + \Delta S_i^*}{1 + \frac{1}{2}\Delta S_i^*} \leq \frac{2-\alpha}{1 + \frac{1-\alpha}{2}},$$

where the last inequality is due to $\max\{\Delta S_1^*, \Delta S_2^*\} \leq \max\{R_1, R_2\} \leq 1 - \alpha$.

For the second case when $\beta < \frac{1}{2}$, without loss of generality, assume resource 1 is used up in $\mathbf{A}$. Recall that for BAL we can show that $\Delta S_2^* \leq 2\Delta S_2$. However, this does not hold for BAL*. To get a similar bound for BAL*, we interpret BAL* as a mechanism consists of 3 steps: In step 1 all agents except i^* receive $\frac{1}{n}$ dominant share; In step 2 agent i^* receive $\frac{1}{n}$ dominant share; The step 3 is the same as the original step 2. Now we compare $\Delta S_2 + \frac{1}{n}$ and $\Delta S_2^* + \frac{1}{n}$, where the additional $\frac{1}{n}$ can be imagined as the amount of resource 2 received by i^* in step 2 of BAL*. Since $\beta < \frac{1}{2}$, in step 3 agents in G_1 receive more than $\frac{R_1^*}{2}$ of resource 1. Then the remaining at most $\frac{R_1^*}{2}$ of resource 1 is allocated to agents in G_2 in step 2 and step 3. In other words, agents in G_2 receive at most $\frac{R_1^*}{2}$ of resource 1 and $\Delta S_2 + \frac{1}{n}$ of resource 2 in step 2 and step 3. Then even if we allocate all R_1^* of resource 1 to agents in G_2 in step 2 and step 3, they can use at most $2(\Delta S_2 + \frac{1}{n})$ of resource 2 since $i^* \in G_2$ is the agent in G_2 with the minimum $d_{i,1}$ and D_k in Algorithm 7 is decreasing. It follows that $\Delta S_2^* + \frac{1}{n} \leq 2(\Delta S_2 + \frac{1}{n})$. Note that if resource 2 is used up, we can show $\Delta S_1^* + \frac{1}{n} \leq 2(\Delta S_1 + \frac{1}{n})$. Since Lemma 7.11 also holds for the allocation $\mathbf{A}$ by BAL*, we have $\Delta S_1^* \leq \Delta S_1$. Therefore, the fair-ratio for SW of BAL is upper bounded by

$$\frac{1+\Delta S_1^*+\Delta S_2^*}{1+\Delta S_1+\Delta S_2} \le \frac{1+\Delta S_2^*}{1+\Delta S_2} = \frac{1-\frac{1}{n}+\Delta S_2^*+\frac{1}{n}}{1-\frac{1}{n}+\Delta S_2+\frac{1}{n}} \le \frac{1-\frac{1}{n}+\Delta S_2^*+\frac{1}{n}}{1-\frac{1}{n}+\frac{\Delta S_2^*+\frac{1}{n}}{2}}$$
$$\le \frac{1-\frac{1}{n}+(1-\alpha+\frac{1}{n})}{1-\frac{1}{n}+\frac{1-\alpha+\frac{1}{n}}{2}}$$
$$= \frac{2-\alpha}{1+\frac{1-\alpha}{2}-\frac{1}{2n}}.$$

Next we consider utilization. We also distinguish two cases when $\beta \ge \frac{1}{2}$ and when $\beta < \frac{1}{2}$. For the first case, we have that $\Delta S_i \ge \frac{R_i^*}{2} \ge \frac{R_i}{2}$ for $i = 1, 2$. Then at least $1 - \frac{R_1}{2} \ge 1 - \frac{\alpha}{2} \ge 1 - \frac{1-\alpha}{2}$ of resource 1 is used and at least $1 - \frac{R_2}{2} \ge 1 - \frac{1-\alpha}{2}$ of resource 2 is used. Therefore, the fair-ratio for utilization of BAL is upper bounded by $\frac{1}{1-\frac{1-\alpha}{2}}$.

For the second case when $\beta < \frac{1}{2}$, we further distinguish two cases when resource 1 is used up and when resource 2 is used up in $\mathbf{A}$. Let $y_1 = \sum_{i\in G_1} \frac{d_{i,2}}{n}$ be the amount of resource 2 received by agents in G_1 and $y_2 = \sum_{i\in G_2} \frac{d_{i,1}}{n}$ be the amount of resource 1 received by agents in G_2 when every agent receives $\frac{1}{n}$ dominant share.

If resource 1 is used up, as shown in the above we have that $\Delta S_2^* + \frac{1}{n} \le 2(\Delta S_2 + \frac{1}{n})$. Let y_1^* be the mount of resource 2 received by G_1 in $\mathbf{A}^*$. As shown in Lemma 7.12 we have $y_1^* \le 2y_1$. Therefore, the fair-ratio for utilization (determined by resource 2) is upper bounded by

$$\frac{\alpha-\frac{1}{n}+\Delta S_2^*+\frac{1}{n}+y_1^*}{\alpha-\frac{1}{n}+\Delta S_2+\frac{1}{n}+y_1} \le \frac{\alpha-\frac{1}{n}+\Delta S_2^*+\frac{1}{n}+y_1^*}{\alpha-\frac{1}{n}+\frac{\Delta S_2^*+\frac{1}{n}+y_1^*}{2}} \le \frac{\alpha-\frac{1}{n}+1-\alpha+\frac{1}{n}}{\alpha-\frac{1}{n}+\frac{1-\alpha+\frac{1}{n}}{2}} = \frac{1}{1-\frac{1-\alpha}{2}-\frac{1}{2n}}.$$

If resource 2 is used up, as shown in the above we have that $\Delta S_1^* + \frac{1}{n} \le 2(\Delta S_1 + + \frac{1}{n})$. Let y_2^* be the mount of resource 1 received by G_2 in $\mathbf{A}^*$. As shown in Lemma 7.12 we have $y_2^* \le \frac{3y_2}{2} \le 2y_2$. Therefore, the fair-ratio for utilization (determined by resource 1) is upper bounded by

$$\frac{1-\alpha-\frac{1}{n}+\Delta S_1^*+\frac{1}{n}+y_2^*}{1-\alpha-\frac{1}{n}+\Delta S_1+\frac{1}{n}+y_2} \le \frac{1-\alpha-\frac{1}{n}+\Delta S_1^*+\frac{1}{n}+y_2^*}{1-\alpha-\frac{1}{n}+\frac{\Delta S_1^*+\frac{1}{n}+y_2^*}{2}} \le \frac{1-\alpha-\frac{1}{n}+\alpha+\frac{1}{n}}{1-\alpha-\frac{1}{n}+\frac{\alpha+\frac{1}{n}}{2}} = \frac{1}{1-\frac{1-\alpha}{2}-\frac{1}{2n}}.$$

□

Lemma 7.18. *For the situation with* 2 *resources, the fair-ratio of* BAL *and* BAL* *is lower bounded by* $\frac{2-\alpha}{1+\frac{1-\alpha}{2}}$ *for SW and lower bounded by* $\frac{1}{1-\frac{1-\alpha}{2}}$ *for utilization.*

Proof. We first study SW. We build an instance with minority population ratio α and n agents as follows. The first group G_1 consists of $n(1-\alpha)$ agents who have the same demand vector $(1, \varepsilon)$, where $\varepsilon = \frac{1}{n^2}$. The second group G_2 consists of $n\alpha$ agents, where except for one special agent i^* whose demand vector is $(\frac{1}{n(1-\alpha)}, 1)$, all other agents have the same demand vector $(1-\varepsilon, 1)$. The idea is that under BAL and BAL* the special agent i^* can get only about $\frac{1-\alpha}{2}$ of the second resource and the SW is about $1 + \frac{1-\alpha}{2}$ while there exists an allocation that satisfies SI, EF, and PO, and has SW about $2 - \alpha - \frac{1}{n}$ by giving roughly $1-\alpha$ dominant share to the special agent i^*.

Formally, we first give $\frac{1}{n}$ dominant share to every agent except for agent i^*. Then the remaining amount of two resources are $R_1^0 = \alpha - (n\alpha - 1)\frac{1-\frac{1}{n^2}}{n} \geq \frac{1}{n}$ and $R_2^0 = (1-\alpha)(1-\frac{1}{n^2}) + \frac{1}{n} \geq 1-\alpha$. We can give agent i^* the bundle $(\frac{1-\frac{1}{n^2}}{n}, (1-\frac{1}{n^2})(1-\alpha))$ and allocate remaining resources evenly to agents in G_1. The SW is lower bounded by

$$1 - \frac{1}{n} + (1 - \frac{1}{n^2})(1-\alpha) \geq 2 - \alpha - \frac{2}{n}.$$

It is easy to verify that the above allocation, denoted by $\mathbf{A}^*$, satisfies SI and EF.

Under BAL, the remaining resources after step 1 are

$$R_1 = \alpha - \frac{\frac{1}{n(1-\alpha)} + (n\alpha - 1)(1 - \frac{1}{n^2})}{n} < \frac{1}{n} \text{ and } R_2 = (1-\alpha)(1 - \frac{1}{n^2}) < 1 - \alpha.$$

If in step 2 we give the special agent i^* a bundle $(\frac{1}{2n}, \frac{1-\alpha}{2})$, then the increased dominant shares for two groups ΔS_1 and ΔS_2 satisfy that $\Delta S_1 \leq R_1 - \frac{1}{2n} < \frac{R_1}{2}$ and $\Delta S_2 = \frac{1-\alpha}{2} > \frac{R_2}{2}$. This means in step 2 the dominant share of the special agent i^* is increased by at most $\frac{1-\alpha}{2}$. Then the SW under BAL is upper bounded by $1 + R_1 + \frac{1-\alpha}{2} \leq 1 + \frac{1-\alpha}{2} + \frac{1}{n}$ and the fair-ratio for SW is lower bounded by

$$\frac{2 - \alpha - \frac{2}{n}}{1 + \frac{1-\alpha}{2} + \frac{1}{n}} \xrightarrow{n\to\infty} \frac{2-\alpha}{1 + \frac{1-\alpha}{2}}.$$

For BAL*, we have

$$\frac{R_1^*}{R_2^*} = \frac{R_1 + \frac{1}{n^3(1-\alpha)}}{R_2 + \frac{1}{n^3}} \geq \frac{R_1 + \frac{1}{n^3}}{R_2 + \frac{1}{n^3}} \geq \frac{R_1}{R_2},$$

where the last inequality follows by $R_1 < R_2$. This means that under BAL* the special agent gets less resources in step 2 than under BAL, i.e., $\Delta S_2^* \leq \Delta S_2$. Using the same argument for BAL we get that the fair-ratio for SW of BAL* is also lower bounded by $\frac{2-\alpha}{1+\frac{1-\alpha}{2}}$.

For utilization, we use the same instance in the above. In $\mathbf{A}^*$, when we give agent i^* the bundle $(\frac{1-\frac{1}{n}}{n^2}, (1-\frac{1}{n})(1-\alpha))$ and every other agent $\frac{1}{n}$ dominant share, the remaining amount of resource 1 is at most $R_1^0 \leq \frac{\alpha}{n^2} + \frac{1}{n} \leq \frac{2}{n}$ and the remaining amount of resource 1 is at most $R_2^0 - (1-\frac{1}{n^2})(1-\alpha) = \frac{1}{n}$. Thus, utilization of $\mathbf{A}^*$ is at least $1 - \frac{2}{n}$.

Under BAL, the remaining resources after step 1 are $R_1 < \frac{1}{n}$ and $R_2 = (1-\alpha)(1-\frac{1}{n^2})$, and we have shown that $\Delta S_2 \leq \frac{1-\alpha}{2}$. Since $R_1 < \frac{1}{n}$, agents in G_2 receive at most $\frac{1}{n}$ of resource 2 in step 2. Thus, at least $R_2 - \frac{1-\alpha}{2} - \frac{1}{n} \geq \frac{1-\alpha}{2} - \frac{2}{n}$ of resource 2 is not used under BAL. Then the fair-ratio for utilization of BAL is lower bounded by

$$\frac{1-\frac{2}{n}}{1-\frac{1-\alpha}{2}-\frac{2}{n}} \xrightarrow{n\to\infty} \frac{1}{1-\frac{1-\alpha}{2}}.$$

Under BAL*, we have shown that $\Delta S_2^* \leq \Delta S_2 \leq \frac{1-\alpha}{2}$. Then using the same argument for BAL, at least $R_2 - \frac{1-\alpha}{2} - \frac{1}{n} \geq \frac{1-\alpha}{2} - \frac{2}{n}$ of resource 2 is not used under BAL* and we get the same lower bound $\frac{1}{1-\frac{1-\alpha}{2}}$ for BAL*. □

Lemma 7.19. *For the situation with* 2 *resources,* BAL *and* BAL* *can be implemented in polynomial time. Specifically, mechanism* BAL *and* BAL* *can be implemented in $O(n^2)$ time and $O(n)$ space.*

Proof. Notice that both $|P_1|$ and $|P_2|$ are non-decreasing and at least one of them is increasing in each round of step 2, so the number of rounds of step 2 is at most n. Since each round of step 2 can be implemented in $O(n)$ time, BAL and BAL* can be implemented in $O(n^2)$ time. Since only the amount of each resource allocated to each agent needs to be stored, the space complexity is $O(n)$. □

7.6.5 Proof of Theorem 7.5

SI and PO are clearly satisfied. For SP, suppose there exists a mechanism $F_g \in \mathcal{F}$ that is not SP. Let i^* be the agent who can benefit by reporting a false demand vector $\mathbf{d}'_{i^*}$ instead of the true demand vector $\mathbf{d}_{i^*}$ in an instance $\mathbf{I}$. Denote the truthful outcome by $\mathbf{A}$ and the manipulated outcome by $\mathbf{A}'$. Let $t = \min_{i \in N} g(\mathbf{A}_i)$ and $P = \{i \in N \mid g(\mathbf{A}_i) = t\}$. Let t' and P' be the corresponding notations for $\mathbf{A}'$. Note that for any agent i, since the allocation is non-wasteful, we have $u_i(\mathbf{A}'_i) > u_i(\mathbf{A}_i) \Leftrightarrow \mathbf{A}'_i > \mathbf{A}_i \Leftrightarrow g(\mathbf{A}'_i) > g(\mathbf{A}_i)$, where $\mathbf{A}'_i > \mathbf{A}_i$ means that $A'_{i,j} > A_{i,j}$ for all $j \in R$. Then from $u_{i^*}(\mathbf{A}'_{i^*}) > u_{i^*}(\mathbf{A}_{i^*})$ we get $g(\mathbf{A}'_{i^*}) > g(\mathbf{A}_{i^*})$ and $\mathbf{A}'_{i^*} > \mathbf{A}_{i^*}$. Since $\mathbf{A}'_{i^*} > \mathbf{A}_{i^*} \geq \frac{1}{n}\mathbf{d}_{i^*}$, it must be that $i^* \in P'$ and $t \leq g(\mathbf{A}_{i^*}) < g(\mathbf{A}'_{i^*}) = t'$. Consequently, for any $i \in P'$, we have $g(\mathbf{A}'_i) = t' \geq \max\{t, g(\frac{1}{n}\mathbf{d}_i)\} \geq g(\mathbf{A}_i)$, and for any $i \in N \setminus P'$, we have $g(\mathbf{A}'_i) = g(\mathbf{A}_i) = g(\frac{1}{n}\mathbf{d}_i) > t' > t$. Thus, we have $g(\mathbf{A}'_i) \geq g(\mathbf{A}_i) \Leftrightarrow \mathbf{A}'_i \geq \mathbf{A}_i$ for all $i \in N$ and $\mathbf{A}'_{i^*} > \mathbf{A}_{i^*}$. This contradicts with that $\mathbf{A}$ is PO. This finishes the proof for SP.

Next we show EF. Suppose there exists a mechanism $F_g \in \mathcal{F}$ that is not EF. Let i and i' be two agents such that i envies i' in an allocation $\mathbf{A}$ produced by F_g, i.e., $u_i(\mathbf{A}_{i'}) > u_i(\mathbf{A}_i)$. Then we have $\mathbf{A}_{i'} > \mathbf{A}_i$ and hence $g(\mathbf{A}_{i'}) > g(\mathbf{A}_i)$. Let $t = \min_{j \in N} g(\mathbf{A}_j)$ and $P = \{j \in N \mid g(\mathbf{A}_j) = t\}$. Since $g(\mathbf{A}_{i'}) > g(\mathbf{A}_i)$, we have $i' \notin P$ and then $\mathbf{A}_{i'} = \frac{1}{n}\mathbf{d}_{i'}$. Let k^* be the dominant resource of agent i, we have

$$A_{i,k^*} \geq \frac{1}{n} \geq \frac{1}{n} d_{i',k^*} = A_{i',k^*},$$

which contradicts with $\mathbf{A}_{i'} > \mathbf{A}_i$. This finishes the proof for EF.

7.6.6 Proof of Theorem 7.6

Let $\mathbf{A}$ be the allocation under UNB. We differentiate two cases according to whether there exists a resource other than resource 1 that is used up in $\mathbf{A}$. If there exists such an resource, assume this resource is resource 2. Denote $x = \sum_{i \in G_1} A_{i,2}$.

Note that $\sum_{i \in G_1} A_{i,1} = \sum_{i \in G_1} \frac{1}{n} = 1 - \alpha$ and $x \leq \sum_{i \in G_1} A_{i,1} = 1 - \alpha$. Then

$$\begin{aligned}\mathsf{SW}(A) &= \sum_{i \in G_1} A_{i,1} + \sum_{j \in \{2,\ldots,m\}} \sum_{i \in G_j} A_{i,j} \\ &\geq 1 - \alpha + \sum_{j \in \{2,\ldots,m\}} \sum_{i \in G_j} A_{i,2} \\ &= 1 - \alpha + 1 - x.\end{aligned}$$

Since the SW of the optimal allocation is upper bounded by $m - x - \alpha\beta$, we have

$$\mathrm{FR}_{\mathsf{SW}}(\mathrm{UNB}) \leq \frac{m - x - \alpha\beta}{1 - \alpha + 1 - x} \leq \frac{m - (1-\alpha) - \alpha\beta}{1 - \alpha + 1 - (1 - \alpha)} = m - (1 - \alpha) - \alpha\beta,$$

where the first inequality holds since $m - \alpha\beta \geq 2 - \alpha\beta \geq 2 - \alpha$.

To show $\mathrm{FR}_{\mathsf{SW}}(\mathrm{UNB}) \geq m - (1-\alpha) - \alpha\beta$, consider the following instance. We set one parameter ε that is close to 0. In G_1, all agents have the demand vector $(1, 1-\varepsilon, \varepsilon, \ldots, \varepsilon)$ except one agent whose demand vector is $(1, \varepsilon, \varepsilon, \ldots, \varepsilon)$. G_2 consists of $n\alpha - (m-2) - 1$ agents who have the same demand vector $(\beta, 1, \varepsilon, \ldots, \varepsilon)$ and one agent who has the demand vector $(\varepsilon^2, 1, \varepsilon, \ldots, \varepsilon)$. Each of the remaining $m - 2$ agents has a different dominant resource for the remaining $m - 2$ resources and they demand ε for all non-dominant resources. Under UNB, every agent gets $\frac{1}{n}$ dominant share when ε approaches 0. In the optimal allocation, when ε approaches 0, the agent in G_1 with demand vector $(1, \varepsilon, \varepsilon, \ldots, \varepsilon)$ gets $1 - \alpha\beta$ dominant share; each agent in G_2 gets $\frac{1}{n}$ dominant share; each remaining agent gets 1 dominant share. So $\mathrm{FR}_{\mathsf{SW}}(\mathrm{UNB}) \geq m - (1-\alpha) - \alpha\beta$ when ε approaches 0. This instance also shows that $\mathrm{FR}_{\mathrm{Util}}(\mathrm{UNB}) = \infty$.

For the second case when only resource 1 is used up, Since UNB always increases the allocations of agents with the smallest fraction of resource 1, we have that

$$\frac{\sum_{j \in \{2,\ldots,m\}} \sum_{i \in G_j} A_{i,j}}{\sum_{j \in \{2,\ldots,m\}} \sum_{i \in G_j} A_{i,1}} \geq \frac{\sum_{j \in \{2,\ldots,m\}} \sum_{i \in G_j} d_{i,j}}{\sum_{j \in \{2,\ldots,m\}} \sum_{i \in G_j} d_{i,1}} = \frac{\alpha}{\alpha\beta} = \frac{1}{\beta}.$$

Since $\alpha = \sum_{j \in \{2,\ldots,m\}} \sum_{i \in G_j} A_{i,1}$, we have $\sum_{j \in \{2,\ldots,m\}} \sum_{i \in G_j} A_{i,j} \geq \frac{\alpha}{\beta}$. Then

$$\begin{aligned}\mathsf{SW}(A) &= \sum_{i \in G_1} A_{i,1} + \sum_{j \in \{2,\ldots,m\}} \sum_{i \in G_j} A_{i,j} \\ &\geq 1 - \alpha + \frac{\alpha}{\beta}.\end{aligned}$$

Then

$$\mathrm{FR}_{\mathsf{SW}}(\mathrm{UNB}) \leq \frac{m-\alpha\beta}{1-\alpha+\frac{\alpha}{\beta}} = \frac{m-\alpha\beta}{1+\frac{1-\beta}{\beta}\alpha}.$$

To show $\mathrm{FR}_{\mathsf{SW}}(\mathrm{UNB}) \geq \frac{m-\alpha\beta}{1+\frac{1-\beta}{\beta}\alpha}$, consider the following instance. In G_1, all agents have the demand vector $(1, \varepsilon, \ldots, \varepsilon)$, where ε is close to 0. For each resource $j \in [2, m]$, G_j consists of $\frac{n\alpha}{m-1}$ agents, where one agent has a special demand vector $(\frac{\beta}{\sqrt{n}}, \varepsilon^2, \ldots, \varepsilon^2, 1, \varepsilon^2, \ldots, \varepsilon^2)$, and the remaining agents have the same demand vector $(\beta, \varepsilon, \ldots, \varepsilon, 1, \varepsilon, \ldots, \varepsilon)$. The fair-ratio approaches the desired lower bound when n approaches ∞. Combining the two cases, we have $\mathrm{FR}_{\mathsf{SW}}(\mathrm{UNB}) = \max\{m-(1-\alpha)-\alpha\beta, \frac{m-\alpha\beta}{1+\frac{1-\beta}{\beta}\alpha}\}$.

Similarly, we can get the fair-ratio for DRF. The lower bounds can be proved using the same instances for UNB. Notice that

$$1-\alpha(1-\beta) - \frac{1}{1+\frac{1-\beta}{\beta}\alpha} = \frac{(\beta+\frac{1}{\beta}-2)(1-\alpha)\alpha}{1+\frac{1-\beta}{\beta}\alpha} > 0,$$

since $\alpha \in (0, 1-\frac{1}{m}]$ and $\beta \in (0,1)$. Thus, $\mathrm{FR}_{\mathsf{SW}}(\mathrm{UNB})$ is upper bounded by $\mathrm{FR}_{\mathsf{SW}}(DRF)$.

For the complexity, we can follow the proof of the complexity of UNB in two resources and achieve a similar result. Since there are m resources in this setting, UNB can be implemented in $O(n^2m)$ time and $O(nm)$ space.

Chapter 8

Conclusions

In this thesis, we begin by addressing the efficiency issue in resource allocation. We focus on the fully online matching model and propose an efficient algorithm that performs well both theoretically and numerically. For fully online matching model, it can be widely applied in various domains, including ride-sharing, online chess game platform and kidney exchange, which is discussed in the introduction of Chapter 3.

In the second part, we shift our attention to the fairness issue in resource allocation. We focus on the fair division setting, which is a specific offline resource allocation problem. We consider the mixed goods model and the indivisible goods model under ordinal and uncertain preferences. We develop algorithms that can output a fair allocation in both of these models. These models can also be applied in multiple applications, including the course assignment in school and the inheritance division.

In the final part, we tackle a more challenging problem: can we find an allocation that is both fair and efficient? To explore this, we consider the indivisible goods model, the mixed goods model, and the cloud computing model. In the first two models, we investigate the trade-off between efficiency and fairness by analyzing the efficiency loss when imposing fairness constraints. In the cloud computing model, we aim to find an efficient allocation among fair allocations and develop effective algorithms with good performance guarantees.

For future work, the following are some potential directions worth considering.

- Could we adapt some fairness notions used in the fair division problem to the online matching setting?

  As introduced in Chapter 1, some papers have considered the fairness issue in the online matching setting. However, some other applicable choices of fairness notions should also be considered in the online resource allocation setting. Considering fairness beyond the classical online matching setting, such as in the fully online matching model, is also an important direction.

- Returning to the offline resource allocation setting, can we combine other efficiency measures with classical fairness notions? Achieving optimal social welfare while maintaining fairness may be overly challenging in practical applications, as highlighted by the significant gap discussed in Chapter 6. Instead, aiming for a weaker efficiency guarantee alongside fairness could be a more practical approach. For instance, Pareto optimality offers a less stringent efficiency criterion compared to maximizing social welfare. Investigating the compatibility of Pareto optimality with fairness notions such as EFX or EFM presents an important and promising direction for future research.

- In classical fair division, there are typically no restrictions on the set of items allocated to an agent. However, introducing cardinality or budget constraints could significantly impact the algorithms used to find fair allocations. This is an important area of exploration, as such constraints are often present in real-world applications. For example, in course assignments at schools, students may be limited in the number of courses they can take. Incorporating these constraints into the allocation process can enhance its practicality and flexibility in real-world scenarios.

To summarize, we believe that efficiency and fairness are two fundamental issues in resource allocation that warrant deeper and broader exploration.

Bibliography

[1] Gagan Aggarwal, Gagan Goel, Chinmay Karande, and Aranyak Mehta. Online vertex-weighted bipartite matching and single-bid budgeted allocations. In *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms*, pages 1253–1264. SIAM, 2011. 17

[2] Hannaneh Akrami, Noga Alon, Bhaskar Ray Chaudhury, Jugal Garg, Kurt Mehlhorn, and Ruta Mehta. EFX: A simpler approach and an (almost) optimal guarantee via rainbow cycle number. In *Proceedings of the 24th ACM Conference on Economics and Computation (EC)*, page 61, 2023. 84

[3] Noga Alon. Splitting necklaces. *Advances in Mathematics*, 63(3):247–253, 1987. 83

[4] 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. 84

[5] 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*, 322:103965, 2023. 2, 42, 87

[6] Ali Aouad and Omer Saritac. Dynamic Stochastic Matching Under Limited Time. SSRN Scholarly Paper ID 3497624, Social Science Research Network, Rochester, NY, December 2019. URL `https://papers.ssrn.com/abstract=3497624`. 18, 32

[7] Itai Ashlagi, Yossi Azar, Moses Charikar, Ashish Chiplunkar, Ofir Geri, Haim Kaplan, Rahul Makhijani, Yuyi Wang, and Roger Wattenhofer. Min-cost Bipartite Perfect Matching with Delays. page 20, 2016. 18

[8] Itai Ashlagi, Maximilien Burq, Chinmoy Dutta, Patrick Jaillet, Amin Saberi, and Chris Sholley. Edge Weighted Online Windowed Matching. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 729–742, Phoenix AZ USA, June 2019. ACM. ISBN 978-1-4503-6792-9. doi: 10.1145/3328526.3329573. URL `https://dl.acm.org/doi/10.1145/3328526.3329573`. 18, 32

[9] Yonatan Aumann and Yair Dombb. The efficiency of fair division with connected pieces. *ACM Transactions on Economics and Computation*, 3(4): 1–16, 2015. 86

[10] Yonatan Aumann, Yair Dombb, and Avinatan Hassidim. Computing socially-efficient cake divisions. In *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 343–350, 2013. 86

[11] Yossi Azar and Amit Jacob Fanani. Deterministic min-cost matching with delays. *Theory of Computing Systems*, pages 1–21, 2020. 18

[12] Yossi Azar, Ashish Chiplunkar, and Haim Kaplan. Polylogarithmic bounds on the competitiveness of min-cost perfect matching with delays. In *Proceedings of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1051–1061. SIAM, 2017. 18

[13] Haris Aziz, Serge Gaspers, Simon Mackenzie, and Toby Walsh. Fair assignment of indivisible objects under ordinal preferences. *Artificial Intelligence*, 227:71–92, 2015. 63, 64, 65, 66, 68

[14] Haris Aziz, Péter Biró, Serge Gaspers, Ronald De Haan, Nicholas Mattei, and Baharak Rastegari. Stable matching with uncertain linear preferences. In *International Symposium on Algorithmic Game Theory*, pages 195–206. Springer, 2016. 62, 64, 65

[15] Haris Aziz, Péter Biró, Tamás Fleiner, Serge Gaspers, and Ronald de Haan. Stable matching with uncertain pairwise preferences. 2017.

[16] Haris Aziz, Ronald De Haan, and Baharak Rastegari. Pareto optimal allocation under uncertain preferences. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pages 1472–1474. International Foundation for Autonomous Agents and Multiagent Systems, 2017. 62, 64

[17] Haris Aziz, Peter Biro, Ronald de Haan, and Baharak Rastegari. Pareto optimal allocation under compact uncertain preferences. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1740–1747, 2019. 62, 64, 65

[18] Haris Aziz, Peter Biro, Jérôme Lang, Julien Lesca, and Jérôme Monnot. Efficient reallocation under additive and responsive preferences. *Theoretical Computer Science*, 790:1–15, 2019. 62

[19] 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. 42

[20] 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, 2023. 86

[21] Siddharth Barman and Sanath Kumar Krishnamurthy. On the proximity of markets with integral equilibria. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 1748–1755, 2019. 42

[22] Siddharth Barman, Sanath Kumar Krishnamurthy, and Rohit Vaish. Finding fair and efficient allocations. In *Proceedings of the ACM Conference on Economics and Computation (EC)*, pages 557–574, 2018. 42, 86

[23] Siddharth Barman, Ganesh Ghalme, Shweta Jain, Pooja Kulkarni, and Shivika Narang. Fair division of indivisible goods among strategic agents. In *Proceedings of the 18th International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pages 1811–1813, 2019. 86

[24] Siddharth Barman, Umang Bhaskar, and Nisarg Shah. Optimal bounds on the price of fairness for indivisible goods. In *Proceedings of the 16th International Conference on Web and Internet Economics (WINE)*, pages 356–369, 2020. 84, 85, 86, 99, 100, 101, 102

[25] Robert G. Bartle and Donald R. Sherbert. *Introduction to Real Analysis*, volume 2. Wiley New York, 2000. 98

[26] Xiaohui Bei, Ning Chen, Xia Hua, Biaoshuai Tao, and Endong Yang. Optimal proportional cake cutting with connected pieces. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 1263–1269, 2012. 86

[27] Xiaohui Bei, Zihao Li, Jinyan Liu, Shengxin Liu, and Xinhang Lu. Fair division of mixed divisible and indivisible goods. *Artificial Intelligence*, 293: 103436, 2021. 10, 11, 39, 40, 41, 42, 43, 44, 45, 46, 56, 58, 84, 88, 100, 101, 102

[28] Xiaohui Bei, Shengxin Liu, Xinhang Lu, and Hongao Wang. Maximin fairness with mixed divisible and indivisible goods. *Autonomous Agents and Multi-Agent Systems*, 35(2):34:1–34:21, 2021. 42, 84

[29] Xiaohui Bei, Xinhang Lu, Pasin Manurangsi, and Warut Suksompong. The price of fairness for indivisible goods. *Theory of Computing Systems*, 65(7): 1069–1093, 2021. 84, 85, 86, 88, 93, 94, 96, 103

[30] Xiaohui Bei, Zihao Li, and Junjie Luo. Fair and efficient multi-resource allocation for cloud computing. In Kristoffer Arnsfelt Hansen, Tracy Xiao Liu, and Azarakhsh Malekian, editors, *Web and Internet Economics - 18th International Conference, WINE 2022, Troy, NY, USA, December 12-15, 2022, Proceedings*, volume 13778 of *Lecture Notes in Computer Science*, pages 169–186. Springer, 2022. doi: 10.1007/978-3-031-22832-2_10. URL `https://doi.org/10.1007/978-3-031-22832-2_10`. 105

[31] Xiaohui Bei, Shengxin Liu, and Xinhang Lu. Fair division with subjective divisibility. In *Proceedings of the Conference on Web and Internet Economics (WINE)*, page 677, 2023. 42, 84

[32] Ben Berger, Avi Cohen, Michal Feldman, and Amos Fiat. Almost full EFX exists for four agents. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI)*, pages 4826–4833, 2022. 84

[33] Piotr Berman, Marek Karpinski, and A Scott. Approximation hardness of short symmetric instances of max-3sat. Technical report, 2004. 69

[34] Dimitris Bertsimas, Vivek F. Farias, and Nikolaos Trichakis. The price of fairness. *Operations Research*, 59(1):17–31, 2011. 84, 86

[35] Ivona Bezáková and Varsha Dani. Allocating indivisible goods. *ACM SIGecom Exchanges*, 5(3):11–18, 2005. 64

[36] Umang Bhaskar, A. R. Sricharan, and Rohit Vaish. On approximate envy-freeness for indivisible chores and mixed resources. In *Proceedings of the 24th International Conference on Approximation Algorithms for Combinatorial Optimization Problems (APPROX)*, pages 1:1–1:23, 2021. 42, 84

[37] Anna Bogomolnaia and Hervé Moulin. A new solution to the random assignment problem. *Journal of Economic theory*, 100(2):295–328, 2001. 62, 64

[38] Thomas Bonald and James Roberts. Enhanced cluster computing performance through proportional fairness. *Performance Evaluation*, 79:134–145, 2014. 106

[39] Thomas Bonald and James Roberts. Multi-resource fairness: Objectives, algorithms and performance. In *Proceedings of the 2015 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, pages 31–42, 2015. 106

[40] Sylvain Bouveret and Jérôme Lang. Efficiency and envy-freeness in fair division of indivisible goods: Logical representation and complexity. *Journal of Artificial Intelligence Research*, 32:525–564, 2008. 64

[41] Sylvain Bouveret, Ulle Endriss, Jérôme Lang, et al. Fair division under ordinal preferences: Computing envy-free allocations of indivisible goods. In *ECAI*, pages 387–392, 2010. 64

[42] Steven J. Brams and Alan D. Taylor. An envy-free cake division protocol. *The American Mathematical Monthly*, 102(1):9–18, 1995. 42

[43] Steven J. Brams, Michal Feldman, John Lai, Jamie Morgenstern, and Ariel D. Procaccia. On maxsum fair cake divisions. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI)*, pages 1285–1291, 2012. 64

[44] Xiaolin Bu, Zihao Li, Shengxin Liu, Jiaxin Song, and Biaoshuai Tao. On the complexity of maximizing social welfare within fair allocations of indivisible goods. *CoRR*, abs/2205.14296, 2023. 84, 85, 86, 87, 88, 93, 95, 97, 98, 99

[45] Xiaolin Bu, Zihao Li, Shengxin Liu, Jiaxin Song, and Biaoshuai Tao. Fair division with allocator's preference. In *Proceedings of the Conference on Web and Internet Economics (WINE)*, 2023. 84

[46] Eric Budish. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6): 1061–1103, 2011. 10, 40, 42, 58

[47] Rainer E Burkard, Mauro Dell'Amico, and Silvano Martello. *Assignment problems*. Springer, 2009. 62, 64

[48] Ioannis Caragiannis, Christos Kaklamanis, Panagiotis Kanellopoulos, and Maria Kyropoulou. The efficiency of fair division. *Theory of Computing Systems*, 50(4):589–610, 2012. 84, 86

[49] Ioannis Caragiannis, Nick Gravin, and Xin Huang. Envy-freeness up to any item with high Nash welfare: The virtue of donating items. In *Proceedings of the ACM Conference on Economics and Computation (EC)*, pages 527–545, 2019. 84

[50] 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. 10, 40, 42, 86

[51] Ioannis Caragiannis, Jugal Garg, Nidhi Rathi, Eklavya Sharma, Giovanna Varricchio, et al. New fairness concepts for allocating indivisible items. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2554–2562, 2023. 59

[52] Bhaskar Ray Chaudhury, Jugal Garg, and Kurt Mehlhorn. EFX exists for three agents. In *Proceedings of the 21st ACM Conference on Economics and Computation (EC)*, pages 1–19, 2020. 84

[53] Bhaskar Ray Chaudhury, Jugal Garg, Kurt Mehlhorn, Ruta Mehta, and Pranabendu Misra. Improving EFX guarantees through rainbow cycle number. In *Proceedings of the ACM Conference on Economics and Computation (EC)*, pages 310–311, 2021. 84

[54] Bhaskar Ray Chaudhury, Telikepalli Kavitha, Kurt Mehlhorn, and Alkmini Sgouritsa. A little charity guarantees almost envy-freeness. *SIAM Journal on Computing*, 50(4):1336–1358, 2021. 84, 100, 101

[55] George Christodoulou, Amos Fiat, Elias Koutsoupias, and Alkmini Sgouritsa. Fair allocation in graphs. In *Proceedings of the 24th ACM Conference on Economics and Computation (EC)*, pages 473–488, 2023. 84

[56] Yuga J. Cohler, John K. Lai, David C. Parkes, and Ariel D. Procaccia. Optimal envy-free cake cutting. In *Proceedings of the 25th AAAI Conference on Artificial Intelligence (AAAI)*, pages 626–631, 2011. 86

[57] Natalie Collina, Nicole Immorlica, Kevin Leyton-Brown, Brendan Lucier, and Neil Newman. Dynamic Weighted Matching with Heterogeneous Arrival and Departure Rates. *arXiv:2012.00689 [cs]*, January 2021. URL http://arxiv.org/abs/2012.00689. arXiv: 2012.00689. 18, 20

[58] Vincent Conitzer, Rupert Freeman, and Nisarg Shah. Fair public decision making. In *Proceedings of the ACM Conference on Economics and Computation (EC)*, pages 629–646, 2017. 10, 40, 42

[59] Manlove David. *Algorithmics of matching under preferences*, volume 2. World Scientific, 2013. 62, 64

[60] Stephen Demko and Theodore P Hill. Equitable distribution of indivisible objects. *Mathematical Social Sciences*, 16(2):145–158, 1988. 62, 64

[61] Danny Dolev, Dror G. Feitelson, Joseph Y. Halpern, Raz Kupferman, and Nathan Linial. No justified complaints: on fair sharing of multiple resources. In *Innovations in Theoretical Computer Science 2012*, pages 68–75, 2012. 109

[62] Joanna Drummond and Craig Boutilier. Preference elicitation and interview minimization in stable matchings. In Carla E. Brodley and Peter Stone, editors, *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada*, pages 645–653. AAAI Press, 2014. URL http://www.aaai.org/ocs/index.php/AAAI/AAAI14/paper/view/8585. 62

[63] Alexander Eckl, Anja Kirschbaum, Marilena Leichter, and Kevin Schewior. A stronger impossibility for fully online matching. *Operations Research Letters*, 49(5):802–808, 2021. 18

[64] Jack Edmonds and Richard M Karp. Theoretical improvements in algorithmic efficiency for network flow problems. *Journal of the ACM (JACM)*, 19 (2):248–264, 1972. 69

[65] Yuval Emek, Shay Kutten, and Roger Wattenhofer. Online matching: haste makes waste! In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 333–344, 2016. 18

[66] Seyed Esmaeili, Sharmila Duppala, Davidson Cheng, Vedant Nanda, Aravind Srinivasan, and John P. Dickerson. Rawlsian Fairness in Online Bipartite Matching: Two-Sided, Group, and Individual. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(5):5624–5632, June 2023. ISSN 2374-3468. doi: 10.1609/aaai.v37i5.25698. URL https://ojs.aaai.org/index.php/AAAI/article/view/25698. Number: 5. 2

[67] Jon Feldman, Aranyak Mehta, Vahab Mirrokni, and Shan Muthukrishnan. Online stochastic matching: Beating 1-1/e. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pages 117–126. IEEE, 2009. 16, 17

[68] Duncan Karl Foley. Resource allocation and the public sector. *Yale Economics Essays*, 7(1):45–98, 1967. 10, 39, 42

[69] Rupert Freeman, Sujoy Sikdar, Rohit Vaish, and Lirong Xia. Equitable allocations of indivisible goods. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 280–286, 2019. 42

[70] Eric Friedman, Ali Ghodsi, and Christos-Alexandros Psomas. Strategyproof allocation of discrete jobs on multiple machines. In *Proceedings of the 15th ACM conference on Economics and Computation (EC)*, pages 529–546, 2014. 109

[71] Eric J Friedman, Ali Ghodsi, Scott Shenker, and Ion Stoica. Strategyproofness, leontief economies and the kalai-smorodinsky solution, 2011. 109, 121

[72] Peter Gardenfors. Assignment problem based on ordinal preferences. *Management Science*, 20(3):331–340, 1973. 62, 64

[73] Jugal Garg and Aniket Murhekar. Computing fair and efficient allocations with few utility values. *Theoretical Computer Science*, 962:113932, 2023. 84

[74] Jugal Garg and Aniket Murhekar. Computing Pareto-optimal and almost envy-free allocations of indivisible goods. *Journal of Artificial Intelligence Research*, 2023. 86

[75] Ali Ghodsi, Matei Zaharia, Benjamin Hindman, Andy Konwinski, Scott Shenker, and Ion Stoica. Dominant resource fairness: Fair allocation of multiple resource types. In *Proceedings of the 8th USENIX conference on Networked Systems Design and Implementation (NSDI)*, pages 323–336, 2011. 106, 108, 109, 110, 121

[76] Laurent Gourvès, Jérôme Monnot, and Lydia Tlilane. Near fairness in matroids. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, pages 393–398, 2014. 42

[77] Robert Grandl, Ganesh Ananthanarayanan, Srikanth Kandula, Sriram Rao, and Aditya Akella. Multi-resource packing for cluster schedulers. *ACM SIGCOMM Computer Communication Review*, 44(4):455–466, 2014. 106

[78] LLC Gurobi Optimization. Gurobi optimizer reference manual. `https://www.gurobi.com`, 2022. Accessed: 2022-10-01. 33

[79] Avital Gutman and Noam Nisarr. Fair allocation without trade. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 816–823, 2012. 109

[80] Philip Hall. On representatives of subsets. In *Classic Papers in Combinatorics*, pages 58–62. Springer, 2009. 69

[81] Russell Hensley, Asutosh Padhi, and Jeff Salazar. Cracks in the ridesharing market—and how to fill them. *McKinsey & Company*, 2017. 1

[82] Hadi Hosseini, Zhiyi Huang, Ayumi Igarashi, and Nisarg Shah. Class Fairness in Online Matching, March 2022. URL `http://arxiv.org/abs/2203.03751`. arXiv:2203.03751 [cs, econ]. 2

[83] Zhiyi Huang and Xinkai Shu. Online Stochastic Matching, Poisson Arrivals, and the Natural Linear Program. *arXiv:2103.13024 [cs]*, March 2021. URL `http://arxiv.org/abs/2103.13024`. arXiv: 2103.13024. 16, 17, 28

[84] Zhiyi Huang, Binghui Peng, Zhihao Gavin Tang, Runzhou Tao, Xiaowei Wu, and Yuhao Zhang. Tight Competitive Ratios of Classic Matching Algorithms in the Fully Online Model. In *Proceedings of the 2019 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, Proceedings, pages 2875–2886. Society for Industrial and Applied Mathematics, January 2019. doi: 10.1137/1.9781611975482.178. URL `https://epubs.siam.org/doi/abs/10.1137/1.9781611975482.178`. 18

[85] Zhiyi Huang, Ning Kang, Zhihao Gavin Tang, Xiaowei Wu, Yuhao Zhang, and Xue Zhu. Fully Online Matching. *Journal of the ACM*, 67(3):17:1–17:25, May 2020. ISSN 0004-5411. doi: 10.1145/3390890. URL `https://doi.org/10.1145/3390890`. 16, 18

[86] Zhiyi Huang, Zhihao Gavin Tang, Xiaowei Wu, and Yuhao Zhang. Fully Online Matching II: Beating Ranking and Water-filling. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1380–1391, 2020. doi: 10.1109/FOCS46700.2020.00130. ISSN: 2575-8454. 16, 18

[87] Zhiyi Huang, Xinkai Shu, and Shuyi Yan. The Power of Multiple Choices in Online Stochastic Matching. *arXiv:2203.02883 [cs]*, March 2022. URL `http://arxiv.org/abs/2203.02883`. arXiv: 2203.02883. 16, 17

[88] Patrick Jaillet and Xin Lu. Online Stochastic Matching: New Algorithms with Better Bounds. *Mathematics of Operations Research*, 39(3):624–646, August 2014. ISSN 0364-765X, 1526-5471. doi: 10.1287/moor.2013.0621. URL `http://pubsonline.informs.org/doi/abs/10.1287/moor.2013.0621`. 16

[89] Suhan Jiang and Jie Wu. Multi-resource allocation in cloud data centers: A trade-off on fairness and efficiency. *Concurrency and Computation: Practice and Experience*, 33(6):e6061, 2021. 106

[90] Youngmi Jin and Michiaki Hayashi. Efficiency comparison between proportional fairness and dominant resource fairness with two different type resources. In *2016 Annual Conference on Information Science and Systems (CISS)*, pages 643–648, 2016. 106

[91] Youngmi Jin and Michiaki Hayashi. Trade-off between fairness and efficiency in dominant alpha-fairness family. *INFOCOM 2018 - IEEE Conference on Computer Communications Workshops*, pages 391–396, 2018. 107

[92] Carlee Joe-Wong, Soumya Sen, Tian Lan, and Mung Chiang. Multiresource allocation: Fairness–efficiency tradeoffs in a unifying framework. *IEEE/ACM Transactions on Networking*, 21(6):1785–1798, 2013. 106, 107

[93] Richard M Karp, Umesh V Vazirani, and Vijay V Vazirani. An optimal algorithm for on-line bipartite matching. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 352–358, 1990. 15, 17

[94] Ian Kash, Ariel D Procaccia, and Nisarg Shah. No agent left behind: Dynamic fair division of multiple resources. *Journal of Artificial Intelligence Research*, 51:579–603, 2014. 109

[95] Yasushi Kawase, Koichi Nishimura, and Hanna Sumita. Fair allocation with binary valuations for mixed divisible and indivisible goods. *CoRR*, abs/2306.05986, 2023. 39, 42, 84

[96] Brian Kropp, Jessica Knight, and Jonah Shepp. How fair is your workplace? *Harvard Business Review*, 2022. 2

[97] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 62, 64

[98] Benny Lehmann, Daniel Lehmann, and Noam Nisan. Combinatorial auctions with decreasing marginal utilities. *Games and Economic Behavior*, 55(2): 270–296, 2006. 77

[99] Bo Li, Zihao Li, Shengxin Liu, and Zekai Wu. Allocating mixed goods with customized fairness and indivisibility ratio. *CoRR*, abs/2404.18132, 2024. doi: 10.48550/ARXIV.2404.18132. URL `https://doi.org/10.48550/arXiv.2404.18132`. 39

[100] Jin Li and Jingyi Xue. Egalitarian division under leontief preferences. *Economic Theory*, 54(3):597–622, 2013. 109, 121

[101] Weidong Li, Xi Liu, Xiaolu Zhang, and Xuejie Zhang. Multi-resource fair allocation with bounded number of tasks in cloud computing systems. In *National Conference of Theoretical Computer Science (NCTCS)*, pages 3–17, 2017. 107, 109, 111

[102] Zihao Li, Xiaohui Bei, and Zhenzhen Yan. Proportional allocation of indivisible resources under ordinal and uncertain preferences. In James Cussens and Kun Zhang, editors, *Uncertainty in Artificial Intelligence, Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence, UAI 2022, 1-5 August 2022, Eindhoven, The Netherlands*, volume 180 of *Proceedings of Machine Learning Research*, pages 1148–1157. PMLR, 2022. URL `https://proceedings.mlr.press/v180/li22d.html`. 61

[103] Zihao Li, Shengxin Liu, Xinhang Lu, and Biaoshuai Tao. Truthful fair mechanisms for allocating mixed divisible and indivisible goods. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2808–2816, 2023. 42, 84

[104] Zihao Li, Hao Wang, and Zhenzhen Yan. Fully online matching with stochastic arrivals and departures. In Brian Williams, Yiling Chen, and Jennifer Neville, editors, *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 12014–12021. AAAI Press, 2023. doi: 10.1609/AAAI.V37I10.26417. URL `https://doi.org/10.1609/aaai.v37i10.26417`. 15

[105] Zihao Li, Shengxin Liu, Xinhang Lu, Biaoshuai Tao, and Yichen Tao. A complete landscape for the price of envy-freeness. In Mehdi Dastani, Jaime Simão Sichman, Natasha Alechina, and Virginia Dignum, editors, *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2024, Auckland, New Zealand, May 6-10, 2024*, pages 1183–1191. International Foundation for Autonomous Agents and Multiagent Systems / ACM, 2024. doi: 10.5555/3635637.3662975. URL `https://dl.acm.org/doi/10.5555/3635637.3662975`. 42, 83

[106] 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. 10, 40, 42, 45, 48, 64, 83, 99

[107] Shengxin Liu, Xinhang Lu, Mashbat Suzuki, and Toby Walsh. Mixed fair division: A survey. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 22641–22649, 2024. Senior Member Presentation Track. 2, 39, 42, 84

[108] Charles Lumet, Sylvain Bouveret, and Michel Lemaître. Fair division of indivisible goods under risk. In *Proceedings of the 20th European Conference on Artificial Intelligence (ECAI'12)*, volume 242, pages 564–569. IOS Press, 2012. 64

[109] Will Ma, Pan Xu, and Yifan Xu. Fairness Maximization among Offline Agents in Online-Matching Markets, September 2021. URL `http://arxiv.org/abs/2109.08934`. arXiv:2109.08934 [cs]. 2

[110] Nicholas Mattei and Toby Walsh. Preflib: A library for preferences http://www. preflib. org. In *International Conference on Algorithmic DecisionTheory*, pages 259–270. Springer, 2013. 77

[111] Vijay Menon and Kate Larson. Robust and approximately stable marriages under partial information. In *International Conference on Web and Internet Economics*, pages 341–355. Springer, 2018. 64

[112] Hervé Moulin. Fair division in the internet age. *Annual Review of Economics*, 11(1):407–441, 2019. 42

[113] Aniket Murhekar and Jugal Garg. On fair and efficient allocations of indivisible goods. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 5595–5602, 2021. 42

[114] Sushirdeep Narayana and Ian A Kash. Fair and efficient allocations with limited demands. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*, volume 35, pages 5620–5627, 2021. 109

[115] Antonio Nicoló. Efficiency and truthfulness with leontief preferences. a note on two-agent, two-good economies. *Review of Economic Design*, 8(4):373–382, 2004. 109, 121

[116] Koichi Nishimura and Hanna Sumita. Envy-freeness and maximum Nash welfare for mixed divisible and indivisible goods. *CoRR*, abs/2302.13342v2, 2023. 10, 11, 42, 84

[117] David C Parkes, Ariel D Procaccia, and Nisarg Shah. Beyond dominant resource fairness: Extensions, limitations, and indivisibilities. *ACM Transactions on Economics and Computation (TEAC)*, 3(1):1–22, 2015. 106, 107, 109

[118] Benjamin Plaut and Tim Roughgarden. Almost envy-freeness with general valuations. *SIAM Journal on Discrete Mathematics*, 34(2):1039–1068, 2020. 10, 84

[119] Baharak Rastegari, Anne Condon, Nicole Immorlica, and Kevin Leyton-Brown. Two-sided matching with partial information. In Michael J. Kearns, R. Preston McAfee, and Éva Tardos, editors, *Proceedings of the fourteenth ACM Conference on Electronic Commerce, EC 2013, Philadelphia, PA, USA, June 16-20, 2013*, pages 733–750. ACM, 2013. doi: 10.1145/2492002.2482607. URL `https://doi.org/10.1145/2492002.2482607`. 62

[120] Baharak Rastegari, Anne Condon, Nicole Immorlica, Robert W. Irving, and Kevin Leyton-Brown. Reasoning about optimal stable matchings under partial information. In Moshe Babaioff, Vincent Conitzer, and David A. Easley, editors, *ACM Conference on Economics and Computation, EC '14, Stanford , CA, USA, June 8-12, 2014*, pages 431–448. ACM, 2014. doi: 10.1145/2600057.2602884. URL `https://doi.org/10.1145/2600057.2602884`. 62

[121] Charles Reiss, John Wilkes, and Joseph L Hellerstein. Google cluster-usage traces: format + schema. *Google Inc., White Paper*, pages 1–14, 2011. 118

[122] Jack Robertson and William Webb. *Cake-Cutting Algorithm: Be Fair If You Can*. A K Peters/CRC Press, 1998. 8, 42

[123] Erel Segal-Halevi and Balázs R Sziklai. Resource-monotonicity and population-monotonicity in connected cake-cutting. *Mathematical Social Sciences*, 95:19–30, 2018. 87

[124] Erel Segal-Halevi and Balázs R. Sziklai. Monotonicity and competitive equilibrium in cake-cutting. *Economic Theory*, 68(2):363–401, 2019. 42, 86

[125] Erel Segal-Halevi, Avinatan Hassidim, and Haris Aziz. Fair allocation with diminishing differences. *Journal of Artificial Intelligence Research*, 67:471–507, 2020. 64

[126] Hugo Steinhaus. Sur la division pragmatique. *Econometrica*, 17:315–319, 1949. 10, 39, 42

[127] Warut Suksompong. Constraints in fair division. *ACM SIGecom Exchanges*, 19(2):46–61, 2021. 42

[128] Yad Tahir, Shusen Yang, Alexandros Koliousis, and Julie McCann. Udrf: Multi-resource fairness for complex jobs with placement constraints. In *2015 IEEE Global Communications Conference (GLOBECOM)*, pages 1–7, 2015. 109

[129] Shanjiang Tang, BingSheng He, Shuhao Zhang, and Zhaojie Niu. Elastic multi-resource fairness: balancing fairness and efficiency in coupled cpu-gpu architectures. In *SC'16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 875–886, 2016. 106, 108

[130] Shanjiang Tang, Ce Yu, and Yusen Li. Fairness-efficiency scheduling for cloud computing with soft fairness guarantees. *IEEE Transactions on Cloud Computing*, pages 1–1, 2020. 106

[131] Hal R Varian. Equity, envy, and efficiency. *Journal of Economic Theory*, 9 (1):63–91, 1974. 39, 42

[132] Hao Wang and Xiaohui Bei. Real-time driver-request assignment in ridesourcing. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36 (4):3840–3849, Jun. 2022. doi: 10.1609/aaai.v36i4.20299. URL `https://ojs.aaai.org/index.php/AAAI/article/view/20299`. 18

[133] Wei Wang, Baochun Li, and Ben Liang. Dominant resource fairness in cloud computing systems with heterogeneous servers. In *IEEE INFOCOM 2014-IEEE Conference on Computer Communications*, pages 583–591, 2014. 109

[134] Wei Wang, Baochun Li, Ben Liang, and Jun Li. Multi-resource fair sharing for datacenter jobs with placement constraints. In *SC'16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1003–1014, 2016. 109

[135] Xiaowei Wu, Bo Li, and Jiarui Gan. Budget-feasible maximum Nash social welfare is almost envy-free. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 465–471, 2021. 42

[136] Evan Yifan Xu and Pan Xu. Exploring the Tradeoff Between System Profit and Income Equality Among Ride-hailing Drivers. *Journal of Artificial Intelligence Research*, 79:569–597, February 2024. ISSN 1076-9757. doi: 10.1613/jair.1.15170. URL `https://www.jair.org/index.php/jair/article/view/15170`. 2