Fair and Efficient Allocation of Chores

Ben Wortman, Sitao Zhang, Neil Ashtekar

May 4th, 2022

Abstract

We consider the problem of finding a fair and efficient allocation of chores when agents have supermodular valuations with binary marginal costs. This problem setting naturally models situations in which chores may be more convenient to complete when grouped together. Previous work has studied fair division of chores under additive valuations as well as fair division of goods under submodular valuations. We show that existing methods do not extend to our setting due to chore-specific constraints such as non-positive valuations and completeness.

1 Introduction

Consider the problem of allocating household chores between roommates. Each roommate has preferences over the chores – perhaps some roommates don't mind getting groceries but hate mowing the lawn, while other feel differently. We would like this allocation to be efficient and fair. Informally, we should assign chores such that all roommates are relatively happy, and we want to avoid envy between roommates.

Some chores may be related and are therefore convenient to complete together. For example, consider two chores: getting groceries, and returning library books. If the library is on the way to the grocery store, then completing both chores together will be more convenient than completing them separately, and it may be efficient to assign both chores to a single roommate. Other chores may be unrelated and have no benefit when assigned together – for example, getting groceries and mowing the lawn.

This problem setting can be modeled using supermodular functions with binary marginal costs to express agent valuations of chores (see section 3 for precise definitions). To the best of our knowledge, there has been no prior work on finding fair and efficient allocations of chores in this specific problem setting. In addition, several interesting properties of chores prevent the straightforward application of existing methods for goods in the chores setting.

2 Related Work

2.1 Fair Division of Chores

In comparison to goods, the problem of fair division of chores is relatively under-developed. In general, the allocation of chores is much harder since many of the results established on goods may not have a direct extension on chores. For example, an elegant rule called Maximum Nash Welfare (MNW) satisfies EF1 and PO simultaneously [8]. Unfortunately, MNW has no natural equivalent for chores and the existence of fair and efficient allocation for chores is still a major open question. [15] studied the trade-off between fairness and efficiencies in allocating indivisible chores to agents with additive cost functions. Specifically, it proved that the price of fairness, defined as the supremum ratio of the minimum social cost of a fair allocation to the minimum social cost of any allocation, can be intractable under several fairness notion with more than two agents.

Previous works addressed these problem by either focusing on approximations or by adding domain restrictions. [1] showed that an Max-Min Share (MMS) allocation does not need to exist for chores and the computation of an MMS solution - if it exists - is strongly NP-hard. Upon this, a polynomial-time 2-approximation algorithm for MMS allocation was proposed for chores. [10] studied the fair division for chores under several fairness notions related to equitability. It proved the existence of equitable up to one chore (EQ1) and PO allocation for indivisible chores and provided a pseupolynomial-time algorithm for finding such an allocation. [11] proposed a strongly polynomial-time algorithm for computing an EF1+PO allocation for bivalued utility function. [9] further extend the result to the personalized bivalued utilities where chore valuations are agent-specific. It showed that an MMS allocation always exists and can be computed in polynomial time given a group of bivalued utilities $\{-1, -p_i\}$ where p_i is an integer for each agent i. [13] focused on lexicographic preferences as a domain restriction and showed the existence and the computability of EFX and PO allocation for chores. This result was later on extended to weakly lexicographic utilities [9].

2.2 Fair Division for Non-additive Valuations

Another line of research extends additive valuations to submodular valuations, as submodular valuations display a diminishing returns property which is well-suited for a number of real-world applications [4]. Most of the existing work focused on the fair division of goods only. Among them, the matroid-rank function is the most favored valuation model as it constitutes a well-studied class of submodular functions with binary marginals. [12] showed existence and computability of a 1/3-approximate MMS allocation for goods. [2] further improved the result to a 1/2-approximate MMS allocation. Recently, [3] proved that a MMS allocation is guaranteed to exist and can be found in polynomial time. [4] and [2] studied the fair division of goods under the envy-freeness notion. [2] proposed a deterministic truthful allocation mechanism such that the allocation output is Lorenz dominating, and is consequently EFX and maximizes Nash Social Welfare (MNW). [4] proposed a simpler mechanism such that the allocation output is EF1 and utilitarian optimal. Though the allocation could be incomplete, it is guaranteed to be clean meaning that each bundle contains no item with zero marginal gain. Both mechanisms reduce the allocation problem to the matroid intersection problem [5] to ensure the efficiency of the solution, and achieve the fairness by proper post-processing.

3 Model and Definitions

3.1 Model

In the following, we formally describe our problem setting. We consider a set of $N = \{1, .., n\}$ agents, and a set of $M = \{c_1, ..., c_m\}$ chores. These chores are allocated into an n-set partition of M denoted as $A = (A_1, ..., A_n)$ corresponding to a chore assignment for each agent. Note that the allocation must be *complete*, meaning that all chores are allocated. Subsets of M are referred to as bundles, and each agent has a valuation over all possible bundles represented by a valuation function $v_i : 2^m \to \mathbb{Z}$. We assume that value queries (i.e. returning $v_i(S)$ for any given $S \subseteq M$) can be completed in polynomial time. We consider valuation functions with the following properties:

- Valuations are non-positive[1]: $v_i(S) \leq 0$ for all $S \subseteq M$
- Empty bundles have 0 valuation: $v_i(\emptyset) = 0$
- Valuations are *monotonically non-increasing*: $v_i(S) \geq v_i(T)$ for all $S \subseteq T \subseteq M$
- Valuations have *binary marginal costs* $\Delta_i(S; c) \in \{0, -1\}$. Marginal costs are defined as the disutility of adding an additional chore c to a given bundle $S \in M$:

$$\Delta_i(S; c) \triangleq v_i(S \cup \{c\}) - v_i(S)$$

- Valuations are *supermodular*. Informally, this means that a single chore may add more disutility to a smaller set (S) compared to a larger set (T):

$$\forall\ S \subseteq T \subseteq M,\ \ c \in M \backslash T,\ \ \Delta_i(S; c) \leq \Delta_i(T; c)$$

To motivate why these properties are useful, consider the example from section 1. The marginal cost for an agent to complete an additional chore given that they are assigned a related chore is 0 (i.e. returning library books on the way to the grocery store), while the marginal cost of completing an additional chore given that they are assigned an unrelated chore is -1 (i.e. getting groceries and mowing the lawn). With supermodularity we assume diminishing costs, i.e. if a roommate is already assigned many chores, they may feel that an additional chore is not much of a hassle.

3.2 Fairness and Efficiency

To define fairness, we consider ***envy-freeness (EF)*** and its relaxations [6]. Envy-freeness requires that no agent prefers the chore assignment of another agent to their own $\forall i, j : v_i(A_i) \geq v_i(A_j)$. Relaxations of envy freeness include ***envy-freeness up to any chore (EFX)*** $\forall i, j : \forall c \in A_i,\ v_i(A_i \setminus \{c\}) \geq v_i(A_j)$ and ***envy-freeness up to one chore (EF1)*** $\forall i, j : \exists c \in A_i,\ v_i(A_i \setminus \{c\}) \geq v_i(A_j)$. EF implies EFX, and EFX implies EF1.

To define efficiency, we first consider ***Pareto optimality (PO)***. An allocation is PO if it is not Pareto dominated by another allocation. Formally, allocation A is PO if there is no allocation B such that $v_i(B_i) \geq v_i(A_i)\ \forall i \in N$ with at least one strict inequality. In addition, we consider two measures of social welfare: ***utilitarian social welfare (USW)*** and ***Nash social welfare (NSW)*** [14]. USW is defined as the sum of all agents' valuations of their respective bundles USW$(A) \triangleq \sum_{i=1}^{n} v_i(A_i)$ while NSW is defined as similarly using the product NSW$(A) \triangleq \prod_{i=1}^{n} v_i(A_i)$. An allocation which maximizes USW is referred to as utilitarian optimal, while an allocation which maximizes NSW is referred to as MNW. In the goods setting, both utilitarian optimal and MNW allocations are PO. In the chores setting, a utilitarian optimal allocation is PO, however there is no known equivalent to MNW for chores.

4 Extensions

In this section, we describe the closest related work in detail – allocation of *goods* with submodular valuation functions with binary marginal gains [2, 4] – and discuss why these approaches do not extend to our setting with chores.

[1] Formally, chores have non-positive valuations, while goods have non-negative valuations. We focus on the chores setting and do not consider mixed items.

Both [4] and [2] propose polynomial-time algorithms to fairly and efficiently allocate goods under submodular valuation functions with binary marginal gains. [4] describes a simple algorithm which guarantees an EF1 and utilitarian optimal allocation. [2] focuses on strategyproofness, and provides an algorithm which guarantees EFX and MNW. Both methods model the problem setting using matroids and make use of matroid intersection algorithms.

4.1 Matroids

A matroid is a mathematical object which generalizes the notion of linear independence beyond vectors. Formally, a matroid is a tuple (E, I) where E is a finite set and I is a collection of its independent sets. Matroids satisfy the following axioms [16]:

1. The empty set is independent: $\emptyset \in I$
2. Independent sets are downward closed: if $Y \in I$ and $X \subseteq Y$, then $X \in I$
3. Independent sets satisfy the exchange property: if $X, Y \in I$ and $|X| > |Y|$, then $\exists\, x \in X \setminus Y$ s.t. $Y \cup \{x\} \in I$

The rank of a matroid is the size of its maximum independent set. A matroid rank function (MRF) returns the rank of each subset of E. If an agent has submodular valuations with binary marginal gains, then their valuation function can be modeled using an MRF. Specifically, an agent's set of *clean* bundles forms the set of independent sets I of a matroid. A bundle is clean if it contains no goods for which a given agent has 0 marginal gain (i.e. $\nexists\, g \in S$ s.t. $\Delta_i(S\setminus\{g\}; g) = 0$) and an allocation is clean if all agents receive clean bundles. Note that a clean allocation may not be complete.

4.2 Methods for Goods

The methods in [2, 4] follow the same general approach. First, they compute a utilitarian optimal allocation, then they modify the allocation to satisfy fairness criteria. To compute a utilitarian optimal allocation, two matroids are defined: the first matroid models MRF valuations for each agent, and the second matroid models the constraint that each good can be assigned to at most one agent. Next, the methods make use of existing polynomial-time matroid intersection algorithms to find the largest independent set in the intersection of the two matroids, corresponding to a utilitarian optimal allocation.

Afterwards, [4] swaps goods between agents to satisfy EF1, while [2] performs a series of checks and modifications to ensure that the resulting allocation meets other desirable properties including EFX and MNW.

4.3 Challenges for Chores

There are a number of challenges which cause existing methods for goods to fail when applied to chores. Further, several distinct properties of chores suggest that it may not be possible to model our problem setting using matroids.

First, we describe why finding a fair and efficient allocation for chores in our problem setting is nontrivial. To achieve efficiency, we wish to maximize the number of chores assigned with 0 marginal cost. This is not straightforward because of supermodularity – when added to a small bundle, a chore may have a marginal cost of -1, but when added to a large bundle, its marginal cost may become 0. A priori, it is not obvious how to perform assignments such that the end result is optimal. This fact also makes it challenging to find a fair allocation, although efficiency appears to the be the main bottleneck.

Next, we consider the naive application of existing methods for goods to the chores setting. To do so, we must map the non-positive, supermodular chore valuations to non-negative, submodular good valuations. This mapping will allow us to model the problem using MRFs. A straightforward way to perform this mapping is to take the absolute value[2] of agents' chore bundle valuations (i.e. $V_i^{new}(S) = |V_i(S)|$ for all $S \subseteq M, i \in N$). While absolute value maps to the non-negative submodular setting, it fails all efficiency goals as the resulting allocation will be pessimal with respect to the original chore valuations [3]. To attempt to remedy this failure, we could modify existing methods to find the minimum USW after performing the absolute value mapping. However, this also fails as the empty set (allocating nothing) trivially satisfies minimum USW.

This highlights a key difference between the goods and chores setting – a fair and efficient allocation of goods may be incomplete, while a valid chore allocation must be complete. Recall that existing methods for goods result in clean but potentially incomplete allocations. Incorporating this completeness constraint is challenging within the matroid framework. Existing methods model the constraint that a good can be assigned to at most one agent using a matroid and define independent sets as those which satisfy this constraint. However, the completeness constraint cannot be modeled in a similar way using independent sets of a matroid since the empty set must be independent and independent sets must be downward closed.

5 Conclusion

We have formally described and analyzed the problem of finding fair and efficient allocations of chores when agents have supermodular valuations and binary marginal costs. Although solutions exist for similar problem settings with goods, these solutions cannot be easily extended to chores. This is an unsolved problem setting which presents interesting challenges due to the additional constraints imposed by chores.

References

[1] Haris Aziz, Gerhard Rauchecker, Guido Schryen, and Toby Walsh. Algorithms for max-min share fair allocation of indivisible chores. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, pages 335–341, San Francisco, California, USA, February 2017. AAAI Press.

[2] Moshe Babaioff, Tomer Ezra, and Uriel Feige. Fair and Truthful Mechanisms for Dichotomous Valuations. *arXiv:2002.10704 [cs]*, December 2020. URL `http://arxiv.org/abs/2002.10704`. arXiv: 2002.10704.

[2] We considered mappings other than absolute value such as adding a constant to chore valuations and mapping based on marginal costs, however these approaches failed because they were either infeasible or did not map to the submodular setting.

[3] Another potential approach is to use matroid partitioning, specifically matroid-constrained number partitioning. This involves partitioning a set into a fixed number of subsets such that each subset is independent with respect to its corresponding matroid over the original set [7]. However, we were not able to formally establish the connection between matroid partitioning and our problem setting.

[3] Siddharth Barman and Paritosh Verma. Existence and Computation of Maximin Fair Allocations Under Matroid-Rank Valuations. *arXiv:2012.12710 [cs]*, February 2021. URL `http://arxiv.org/abs/2012.12710`. arXiv: 2012.12710.

[4] Nawal Benabbou, Mithun Chakraborty, Ayumi Igarashi, and Yair Zick. Finding fair and efficient allocations when valuations don't add up. In Tobias Harks and Max Klimm, editors, *Algorithmic Game Theory*, pages 32–46, Cham, 2020. Springer International Publishing. ISBN 978-3-030-57980-7.

[5] Carl Brezovec, Gérard Cornuéjols, and Fred Glover. Two algorithms for weighted matroid intersection. *Math. Program.*, 36(1):39–53, oct 1986. ISSN 0025-5610.

[6] Eric Budish. The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119:1061–1061, 12 2011. doi: 10.1086/664613.

[7] Rainer E Burkard and En-Yu Yao. Constrained partitioning problems. *Discrete applied mathematics*, 28(1):21–34, 1990.

[8] Ioannis Caragiannis, David Kurokawa, Hervé Moulin, Ariel D. Procaccia, Nisarg Shah, and Junxing Wang. The unreasonable fairness of maximum nash welfare. *ACM Trans. Econ. Comput.*, 7(3), sep 2019. ISSN 2167-8375. doi: 10.1145/3355902. URL `https://doi.org/10.1145/3355902`.

[9] Soroush Ebadian, Dominik Peters, and Nisarg Shah. How to Fairly Allocate Easy and Difficult Chores. *arXiv:2110.11285 [cs]*, February 2022. URL `http://arxiv.org/abs/2110.11285`. arXiv: 2110.11285.

[10] Rupert Freeman, Sujoy Sikdar, Rohit Vaish, and Lirong Xia. Equitable Allocations of Indivisible Chores. *arXiv:2002.11504 [cs]*, February 2020. URL `http://arxiv.org/abs/2002.11504`. arXiv: 2002.11504.

[11] Jugal Garg, Aniket Murhekar, and John Qin. Fair and Efficient Allocations of Chores under Bivalued Preferences. *arXiv:2110.09601 [cs]*, October 2021. URL `http://arxiv.org/abs/2110.09601`. arXiv: 2110.09601.

[12] Mohammad Ghodsi, Mohammadtaghi Hajiaghayi, Masoud Seddighin, Saeed Seddighin, and Hadi Yami. Fair allocation of indivisible goods: Improvements and generalizations. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, EC '18, page 539–556, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450358293. doi: 10.1145/3219166.3219238. URL `https://doi.org/10.1145/3219166.3219238`.

[13] Hadi Hosseini, Sujoy Sikdar, Rohit Vaish, and Lirong Xia. Fairly Dividing Mixtures of Goods and Chores under Lexicographic Preferences. *arXiv:2203.07279 [cs, econ]*, March 2022. URL `http://arxiv.org/abs/2203.07279`. arXiv: 2203.07279.

[14] Amartya Sen. *Collective Choice and Social Welfare*. Holden Day, San Francisco, 1970. URL `http://www.amazon.com/Collective-Choice-Social-Welfare-K/dp/0444851275`. Edinburgh: Oliver and Boyd, 197l; Amsterdam: North-Holland, 1979. Swedish translation: Bokforlaget Thales, 1988.

[15] Ankang Sun, Bo Chen, and Xuan Vinh Doan. Fairness Criteria for Allocating Indivisible Chores: Connections and Efficiencies. *arXiv:2101.07435 [cs]*, September 2021. URL `http://arxiv.org/abs/2101.07435`. arXiv: 2101.07435.

[16] William T. Tutte. Lectures on matroids. *Journal of Research of the National Bureau of Standard-Section B Mathematics and Mathematical Physics*, page 1, 1965.