Computing Fair and Efficient Indivisible Chore Allocations with Bounded Surplus

Nicholas Teh

University of Oxford, UK
nicholas.teh@cs.ox.ac.uk

Abstract. We study the problem of allocating indivisible chores with additive costs, seeking allocations that are weighted envy-free up to one chore (WEF1) and fractionally Pareto optimal (fPO). Mahara (SODA 2026) recently proved that such allocations always exist, but finding one in polynomial time for general instances remains open. For every fixed constant s, we give a polynomial-time algorithm when there are n agents and $m \leq n + s$ chores. This guarantee allows for arbitrarily many agents and arbitrarily many distinct cost functions. Our approach proceeds through the prescribed quota problem. For a quota vector q, let $S(q)$ denote the number of chores assigned beyond the first chore in each nonempty bundle. We decide whether there is a WEF1 and fPO allocation respecting q, and compute one if it exists, in time $m^{S(q)}\text{poly}(n, m)$. The algorithm relies on two structural observations: with positive costs and fixed bundle sizes, fPO is equivalent to minimizing the product of assigned costs; and agents who receive at most one chore impose no nontrivial WEF1 constraints. We complement the algorithm with a hardness result showing that the prescribed quota problem becomes strongly NP-hard when $S(q)$ is unbounded, even with two agent types and quotas restricted to $\{3, 5\}$.

1 Introduction

Fair division of indivisible items is a central problem in algorithmic game theory and multi-agent systems; see, e.g., [2,4] for recent surveys. It captures settings in which tasks or resources must be divided among agents with heterogeneous preferences. Two desiderata have been especially central in this literature: *fairness* and *efficiency*. A classical fairness notion is *envy-freeness* (EF), which asks that no agent prefer another agent's bundle to her own. For indivisible items, exact EF may be impossible, and much of the literature therefore studies relaxations such as envy-freeness up to one item (EF1) [27]. An orthogonal desideratum is *Pareto optimality* (PO), which requires that there be no alternative allocation that makes some agent better off without making anyone worse off.

The combination of approximate envy-freeness and Pareto efficiency has become a standard benchmark in indivisible fair division. For indivisible goods (i.e., items that agents value positively), this benchmark is well understood under additive valuations: maximum Nash welfare allocations are EF1 and PO [10], and subsequent work developed algorithmic and fractional-efficiency refinements [6]. Chores (i.e., items that impose costs rather than create value) are more delicate. Moving a chore away from one agent helps that agent but burdens another, and the structural arguments used for goods do not transfer directly.

In this paper, we study the allocation of indivisible chores under additive costs. The chores may represent undesirable shifts, maintenance tasks, grading duties, service obligations, or administrative work, and agents may incur different costs for performing them. We allow agents to have different entitlements, and use the corresponding weighted version of EF1, called *weighted envy-freeness up to one chore* (WEF1). Informally, after removing her most costly chore from her own bundle, an agent's entitlement-normalized cost should be no larger than her entitlement-normalized cost for any other agent's bundle. For efficiency, we use *fractional Pareto optimality* (fPO), which rules out Pareto improvements even by fractional reallocations of chores and is therefore stronger than ordinary PO.

A recent breakthrough of Mahara [29] shows that every additive chore instance admits an EF1 and fPO allocation and, more generally, a WEF1 and fPO allocation in the weighted-entitlement setting. This settles the existential question, but leaves open the computational one: can such an allocation be found in polynomial time when both the number of agents and the number of chores are part of the input?

We study this computational question in the few-extra-chores setting, where the number of chores exceeds the number of agents by at most a constant. Similar "few extra items" settings have been studied extensively for goods and mixed manna [8,17,23,25,26,31]. For chores, this setting has a particularly useful WEF1 structure. If an agent receives at most one chore, then removing her most costly chore leaves zero cost. Thus, such an agent creates no nontrivial WEF1 constraint. The difficulty is therefore concentrated on agents who receive at least two chores.

For the prescribed quota problem underlying our algorithm, let $q = (q_1, \ldots, q_n)$ be a quota vector, where q_i is the number of chores assigned to agent i. We define the *quota surplus* $S(q) := \sum_{i=1}^{n} \max\{q_i - 1, 0\}$. This quantity counts the number of chores assigned beyond the first chore in each nonempty bundle. When $S(q)$ is small, only a small part of the allocation can generate nontrivial WEF1 constraints, even if the number of agents and the number of distinct cost functions are large. This parameter is the basis of our algorithms and hardness results.

1.1 Our Contributions

Our main contribution is a polynomial-time algorithm for computing WEF1 and fPO allocations whenever the number of chores exceeds the number of agents by at most a constant. The result is obtained through a more general prescribed quota algorithm whose running time is controlled by the quota surplus $S(q)$. Table 1 summarizes the algorithmic landscape.

Fractional efficiency with fixed bundle sizes. Our first result gives a simple way to identify fractional Pareto optimality once the number of chores assigned

Table 1. Algorithmic landscape. Results labeled by theorem or corollary number are proved in this work; the final row details the remaining general case. FPT means fixed-parameter tractable.

Setting	**Status**	**Remarks**
$m \le n$	Polynomial time	Base case: all nonempty bundles can be singletons, so WEF1 is immediate.
$m \le n + s$, fixed s	Polynomial time	Theorem 2 for positive costs; Corollary 1 extends this to nonnegative costs.
Prescribed q, fixed $S(q)$	$m^{S(q)}\mathrm{poly}(n, m)$ time	Theorem 1.
Prescribed q, p chore types	Polynomial/FPT for the stated parameters	Theorem 3; polynomial for fixed p and a fixed number of agents with quota at least two, and FPT in $p + S(q)$.
Prescribed q, unbounded $S(q)$	Strongly NP-hard	Theorem 4, even with two agent types and quotas in $\{3, 5\}$.
General m, n	Computationally open	Existence follows from Mahara [29]; polynomial-time computation remains open.

to each agent is fixed. With strictly positive additive costs, an allocation is fPO exactly when the product of the individual chore costs, evaluated for the agents who receive those chores, is as small as possible among all allocations with the same bundle sizes. We state and prove this result in Section 3.

Prescribed quota vectors. We next study the problem in which the bundle sizes are fixed in advance. Given a quota vector q, positive costs, and positive entitlements, we decide whether there is a WEF1 and fPO allocation in which agent i receives exactly q_i chores for every i, and compute one if it exists, in time $m^{S(q)}\mathrm{poly}(n, m)$. For each agent who must receive at least two chores, the algorithm guesses all but one chore in that agent's bundle. These guessed chores determine the left-hand side of that agent's WEF1 inequalities. The one remaining chore for each nonempty agent is selected by a minimum-cost perfect matching whose permitted edges ensure WEF1. A final product comparison checks fPO.

Few extra chores. The prescribed quota algorithm gives us our main result. For every fixed constant s, a WEF1 and fPO allocation can be computed in polynomial time whenever $m \le n + s$. This guarantee allows arbitrarily many agents and imposes no restriction on the number of distinct cost functions. The proof uses the fact that, under positive costs, any WEF1 allocation with $m > n$ must assign at least one chore to every agent. Thus, the quota surplus of such an allocation is exactly $m - n$, which is at most s. We enumerate the possible positive quota vectors and apply the prescribed quota algorithm. A zero-cost preprocessing extends the result from positive costs to nonnegative costs.

Few chore types. We also give a typed-chores extension. If chores have p cost types, then the prescribed quota algorithm can guess type counts rather than individual chores. This gives a polynomial-time algorithm for fixed p and a fixed number of agents with quota at least two, and an algorithm fixed-parameter tractable in $p + S(q)$.

Hardness for prescribed quotas. Finally, we show that the dependence on $S(q)$ is meaningful. When $S(q)$ is unbounded, deciding whether a quota-respecting EF1+PO allocation exists is strongly NP-hard. The hardness holds even with positive integer costs, only two agent types, and quotas restricted to $\{3, 5\}$. The same construction gives strong NP-hardness for EF1+fPO.

1.2 Related Work

Fair and efficient allocation of indivisible items. The fair allocation of indivisible items is a central topic in algorithmic game theory and multi-agent systems; see, e.g., [2,4] for recent surveys. Since exact envy-freeness can fail for indivisible items, much of the literature focuses on relaxations such as envy-freeness up to one item (EF1). The notion of EF1 was implicit in the work of Lipton et al. [27] and became a standard and well-studied relaxation in many subsequent works. For indivisible goods with additive valuations, the maximum Nash welfare allocation is EF1 and PO [10], and Barman, Krishnamurthy, and Vaish [6] gave a pseudo-polynomial-time algorithm for finding EF1+PO allocations and established the existence of EF1+fPO allocations. Several recent works have also established the non-existence of EF1+PO allocations for more general valuation classes [14,28,37]. Brandl, Suksompong, and Teh [9] gave a characterization of MNW, which makes use of both EF1 and PO properties (the latter, implicitly) in its proof. However, these results for goods do not transfer directly to chores, where the direction of envy and the direction of Pareto improvements interact differently.

Fair and efficient chore allocation. Chores are significantly more delicate than goods. Aziz et al. [3] studied fair allocation of indivisible goods and chores and highlighted the difficulty of combining approximate envy-freeness with efficiency for chores. Several positive results were known for restricted chore instances. Ebadian, Peters, and Shah [15], and independently Garg, Murhekar, and Qin [19] gave polynomial algorithms for EF1+PO in bivalued chore instances. Later, Garg, Murhekar, and Qin [20] obtained EF1+fPO allocations for three agents and for instances with at most two disutility functions, as well as EFX+fPO allocations for three-agent bivalued instances. For instances with two chore types, Aziz et al. [5] gave a polynomial-time algorithm for computing EF1+fPO allocations, as well as a polynomial-time algorithm for EFX allocations. Mahara [29] recently resolved the existential question for general additive chores by proving that an EF1+fPO allocation always exists, with an extension to the weighted setting (i.e., agents with possibly unequal entitlements). This leaves open the computational problem of finding such allocations in polynomial time when both n and m are part of the input. Our work addresses this computational question in a setting not covered by constant-agent or few-type algorithms: both the number of agents and the number of distinct cost functions may be unbounded; after zero-cost preprocessing, the quota vectors considered by the algorithm have bounded quota surplus.

Weighted setting (entitlements) and few-type restrictions. Weighted fairness notions (and, in particular, weighted envy-freeness) capture asymmetric obligations or entitlements and have been extensively studied in the goods setting [11,12,13,30,34,35,36]. For chores, Wu, Zhang, and Zhou [38] gave algorithms for WEF1 allocations and studied when weighted picking sequences guarantee stronger properties. Springer, Hajiaghayi, and Yami [32] studied almost envy-free allocations with entitlements for goods and chores. Closest to our weighted setting, Garg, Murhekar, and Qin [21] proved polynomial-time WEF1+PO results for structured instances, including instances with few agent types or few chore types. Our results are orthogonal: we allow arbitrarily many agent types and chore types, but parameterize by the quota surplus $S(q)$. Moreover, our positive results target fPO, which is stronger than PO.

Surplus, balancedness, and cardinality constraints. Our use of surplus differs from the k-surplus model of Akrami et al. [1], where extra copies of chores may be introduced. Here no chores are added and feasibility is not relaxed. The quota surplus $S(q)$ is only a measure of how far the quota vector is from assigning at most one chore to each nonempty agent. Balanced chore allocation is another related direction. Garg, Murhekar, and Qin [22] give a polynomial-time algorithm for computing a balanced fPO allocation; this implies EF1+fPO when $m \leq n$ and EF2+fPO when $m \leq 2n$. Our few extra chores result strengthens the scarce chores setting in a complementary way: for every fixed s, it computes WEF1+fPO allocations whenever $m \leq n + s$, even though some agents may receive more than two chores. More broadly, fair division under constraints has been studied for goods under category, matroid, and balancedness constraints [7,33,24], and for chores under size constraints [16]. These works impose feasibility constraints on the bundles, often for goods, and typically seek fairness or approximate fairness guarantees under those constraints. Our prescribed quota model instead prescribes the bundle cardinalities for chores. Our algorithms ask whether WEF1+fPO can be achieved for given entitlements, while our hardness result already holds for EF1+PO and EF1+fPO under equal entitlements. The strong NP-hardness result for exact quotas shows that even very simple cardinality prescriptions can make the fair and efficient chore problem computationally intractable when the quota surplus is unbounded.

2 Preliminaries

Let $N = \{1, \ldots, n\}$ be the set of *agents* and $M = \{1, \ldots, m\}$ the set of *chores*. Agent i has an *additive cost function* $c_i : 2^M \to \mathbb{R}_{\geq 0}$, with $c_i(S) = \sum_{j \in S} c_{ij}$.[1] An *allocation* is a tuple $A = (A_1, \ldots, A_n)$ that partitions M. For an allocation

[1] All numerical inputs in algorithmic statements are rational.

A, let $q(A) = (|A_1|, \dots, |A_n|)$ denote its *quota vector*. For a vector $q \in \mathbb{Z}^n_{\geq 0}$ with $\sum_i q_i = m$, let $\mathcal{A}(q) = \{A : |A_i| = q_i$ for every $i \in N\}$ be the set of allocations respecting q. We call an allocation in $\mathcal{A}(q)$ *quota-respecting*.

Definition 1 (Weighted EF1 for chores). *Let $\alpha_i > 0$ be the entitlement of agent i. For a set $S \subseteq M$, define $d_i^-(S) := c_i(S) - \max_{j \in S} c_{ij}$ if $S \neq \varnothing$; and $d_i^-(S) := 0$ otherwise. An allocation A is* weighted envy-free up to one chore (WEF1) *if for every ordered pair of agents i, k, $\frac{d_i^-(A_i)}{\alpha_i} \leq \frac{c_i(A_k)}{\alpha_k}$.*

When all entitlements are equal, this is the usual EF1 notion for chores. The definition uses $d_i^-(A_i)$ because, for chores, agent i eliminates as much envy as possible by removing her most costly chore from her own bundle. A fractional allocation is a matrix $x = (x_{ij})_{i \in N, j \in M}$ such that $x_{ij} \geq 0$ for every $i \in N$ and $j \in M$, and $\sum_{i \in N} x_{ij} = 1$ for every $j \in M$. Here, x_{ij} is the fraction of chore j assigned to agent i. For each agent i, denote $x_i = (x_{ij})_{j \in M}$ and define $c_i(x_i) := \sum_{j \in M} c_{ij} x_{ij}$.

Definition 2 (PO and fPO). *An allocation A is* Pareto optimal (PO) *if there is no allocation B such that $c_i(B_i) \leq c_i(A_i)$ for every $i \in N$, with strict inequality for at least one agent. It is* fractionally Pareto optimal (fPO) *if there is no fractional allocation x such that $c_i(x_i) \leq c_i(A_i)$ for every $i \in N$, with strict inequality for at least one agent.*

Clearly fPO implies PO. We use the following standard characterization of fPO for additive costs. It is the chore analogue of the usual weighted social cost characterization of fractional Pareto efficiency.

Proposition 1 (fPO characterization). *Assume $c_{ij} > 0$ for all i, j. An allocation A is fPO if and only if there are weights $\lambda_1, \dots, \lambda_n > 0$ such that, for every $j \in A_i$ and every agent k, $\lambda_i c_{ij} \leq \lambda_k c_{kj}$.*

Equivalently, the same condition can be written as follows: there exist weights $\lambda_1, \dots, \lambda_n > 0$ such that, for every fractional allocation x, $\sum_{i \in N} \lambda_i c_i(A_i) \leq \sum_{i \in N} \lambda_i c_i(x_i)$.

Proof. The "if" direction is immediate: if A minimizes $\sum_i \lambda_i c_i(x_i)$ for some $\lambda_i > 0$, then any fractional allocation that weakly decreases every agent's cost and strictly decreases one agent's cost would have strictly smaller weighted cost, a contradiction. For the converse, let $a_i = c_i(A_i)$, and consider the linear program

$$
\begin{aligned}
\max \; & \sum_{i \in N} \delta_i \\
\text{s.t.} \; & \sum_{j \in M} c_{ij} x_{ij} + \delta_i \leq a_i \quad \forall i \in N, \\
& \sum_{i \in N} x_{ij} = 1 \quad \forall j \in M, \\
& x_{ij} \geq 0 \quad \forall i, j, \\
& \delta_i \geq 0 \quad \forall i.
\end{aligned}
$$

A positive objective value is exactly a fractional Pareto improvement over A. Since A is fPO, the optimum is 0; feasibility follows from the fractional allocation corresponding to A, with $\delta_i = 0$ for all i.

The dual, with variables $\lambda_i \geq 0$ for the first set of constraints and free variables for the chore-assignment equalities, can be written as

$$\begin{aligned} \min \ & \sum_{i \in N} \lambda_i a_i - \sum_{j \in M} p_j \\ \text{s.t.} \ & p_j \leq \lambda_i c_{ij} \quad \forall i \in N,\ j \in M, \\ & \lambda_i \geq 1 \quad \forall i \in N. \end{aligned}$$

By strong duality, there is an optimal dual solution (λ, p) of value 0. In particular, $\lambda_i > 0$ for every i. Let $\sigma(j)$ be the agent who receives chore j in A. Since $p_j \leq \lambda_{\sigma(j)} c_{\sigma(j)j}$ for every j,

$$\sum_{j \in M} p_j \leq \sum_{j \in M} \lambda_{\sigma(j)} c_{\sigma(j)j} = \sum_{i \in N} \lambda_i c_i(A_i) = \sum_{i \in N} \lambda_i a_i.$$

The dual objective value is 0, so equality must hold throughout. Therefore, for every $j \in A_i$, we have $p_j = \lambda_i c_{ij}$. Since also $p_j \leq \lambda_k c_{kj}$ for every agent k, it follows that $\lambda_i c_{ij} \leq \lambda_k c_{kj}$ for every $j \in A_i$ and every k. The equivalent weighted-cost minimization statement follows because the objective is separable over chores. □

3 Product Minimality at Fixed Quotas

Although fPO is defined against all fractional allocations, it has a simple characterization once the quota vector is fixed. For strictly positive costs, define $\Phi(A) = \sum_{i=1}^{n} \sum_{j \in A_i} \log c_{ij}$. Then, minimizing Φ is equivalent to minimizing the product $\prod_{i=1}^{n} \prod_{j \in A_i} c_{ij}$.

Lemma 1. *Assume $c_{ij} > 0$ for all i, j. An allocation A is fPO if and only if*

$$\Phi(A) = \min\{\Phi(B) : B \in \mathcal{A}(q(A))\}.$$

Equivalently, A minimizes the product of assigned costs among all allocations with quota vector $q(A)$.

Proof. Suppose first that A is fPO. By Proposition 1, there are $\lambda_i > 0$ such that, whenever $j \in A_i$, $\lambda_i c_{ij} \leq \lambda_k c_{kj}$ for every k. Let $B \in \mathcal{A}(q(A))$. If $j \in A_i$ and $j \in B_k$, then $\log \lambda_i + \log c_{ij} \leq \log \lambda_k + \log c_{kj}$. Summing over all chores gives us

$$\sum_i |A_i| \log \lambda_i + \Phi(A) \leq \sum_i |B_i| \log \lambda_i + \Phi(B).$$

Since B has the same quota vector as A, the λ-terms cancel. Thus $\Phi(A) \leq \Phi(B)$. Conversely, suppose A minimizes Φ over $\mathcal{A}(q)$, where $q = q(A)$. Let $\ell_{ij} = \log c_{ij}$

and consider

$$\begin{aligned}\min \ & \sum_{i,j} \ell_{ij} x_{ij} \\ \text{s.t.} \ & \sum_i x_{ij} = 1 \quad \forall j \in M, \\ & \sum_j x_{ij} = q_i \quad \forall i \in N, \\ & x_{ij} \geq 0 \quad \forall i, j.\end{aligned}$$

The feasible region is a transportation polytope and hence has integral extreme points. Therefore the LP optimum is attained by an integral allocation in $\mathcal{A}(q)$. Since A minimizes Φ over $\mathcal{A}(q)$, A is an optimal solution of this LP. Its dual is

$$\begin{aligned}\max \ & \sum_j u_j + \sum_i q_i v_i \\ \text{s.t.} \ & u_j + v_i \leq \ell_{ij} \quad \forall i, j.\end{aligned}$$

Let (u, v) be an optimal dual solution. For $j \in A_i$, complementary slackness gives $u_j + v_i = \ell_{ij}$, while dual feasibility gives $u_j + v_k \leq \ell_{kj}$ for every k. Hence, $\ell_{ij} - v_i \leq \ell_{kj} - v_k$. Setting $\lambda_i = e^{-v_i}$ gives $\lambda_i c_{ij} \leq \lambda_k c_{kj}$ for every $j \in A_i$ and every k. Proposition 1 implies that A is fPO. □

Remark 1. The min-cost matching and flow computations with edge costs $\log c_{ij}$ can be carried out without evaluating logarithms. Work instead in the ordered multiplicative group of positive rationals: addition and subtraction of logarithmic costs correspond to multiplication and division of rational costs, and comparisons are performed exactly by cross-multiplication. In particular, equality of two Φ-values is equivalent to equality of the corresponding rational products.

4 Prescribed Quotas and Quota Surplus

We now consider the problem in which the quota vector q is *prescribed* as part of the input. The goal is to find an allocation in $\mathcal{A}(q)$ that is both WEF1 and fPO, or decide that no such allocation exists.

Definition 3 (Quota surplus). *The quota surplus of q is $S(q) = \sum_{i=1}^{n} \max\{q_i - 1, 0\}$. Let $H(q) = \{i : q_i \geq 2\}$, $R(q) = \{i : q_i = 1\}$, and $E(q) = \{i : q_i = 0\}$. Agents in $H(q)$ are called* heavy agents.

Write $H = H(q)$, $R = R(q)$, $E = E(q)$, and $S = S(q)$. Since each agent in H contributes at least one to S, we have $|H| \leq S$. Moreover,

$$\sum_{i \in H} q_i = \sum_{i \in H} (q_i - 1) + |H| = S + |H| \leq 2S.$$

Thus, at most $2S$ chores are assigned to heavy agents.

Algorithm 1 PrescribedQuota

Require: Positive rational costs c_{ij}, positive entitlements α_i, and a quota vector q with $\sum_i q_i = m$.

Ensure: An allocation $A \in \mathcal{A}(q)$ that is WEF1 and fPO, if one exists.

Compute an allocation $B^q \in \arg\min\{\Phi(B) : B \in \mathcal{A}(q)\}$ by min-cost flow with edge costs $\log c_{ij}$.
Let $H = H(q)$, $R = R(q)$, $E = E(q)$, and $N^+ = H \cup R$.
for all partial allocations $(T_h)_{h \in H}$ of core chores to the agents in H **do**
  for each $h \in H$ **do**
    $\tau_h \leftarrow c_h(T_h)/\alpha_h$ and $\beta_h \leftarrow \max_{j \in T_h} c_{hj}$
  end for
  if $E \neq \varnothing$ and $\tau_h > 0$ for some $h \in H$ **then**
    continue
  end if
  Let $M' = M \setminus \bigcup_{h \in H} T_h$.
  Build a bipartite graph between N^+ and M'.
  for each $j \in M'$ **do**
    for each $r \in R$ **do**
      Allow (r, j) iff $\tau_a \leq \frac{c_{aj}}{\alpha_r}$ for every $a \in H$.
    end for
    for each $k \in H$ **do**
      Allow (k, j) iff $c_{kj} \geq \beta_k$ and $\tau_a \leq \frac{c_a(T_k)+c_{aj}}{\alpha_k}$ for every $a \in H$.
    end for
  end for
  Find a minimum-cost perfect matching μ, using cost $\log c_{ij}$ for edge (i, j).
  if no such matching exists **then**
    continue
  end if
  Define $A_h = T_h \cup \{\mu(h)\}$ $(h \in H)$, $A_r = \{\mu(r)\}$ $(r \in R)$, $A_e = \varnothing$ $(e \in E)$.
  if $\Phi(A) = \Phi(B^q)$ **then**
    return A
  end if
end for
return no quota-respecting WEF1+fPO allocation exists

4.1 The prescribed quota algorithm

For a quota vector q, define $\Gamma(q) = \min\{\Phi(A) : A \in \mathcal{A}(q)\}$. By Lemma 1, an allocation respecting q is fPO if and only if it has Φ-value $\Gamma(q)$. The key point is that, for an agent $h \in H(q)$, the algorithm does not need to guess the entire bundle A_h. It guesses only the set $T_h = A_h \setminus \{g_h\}$, where g_h is a chore removed from h's bundle in the WEF1 comparison. The chore g_h is chosen later by a matching.

In Algorithm 1, a tuple $(T_h)_{h \in H}$ is a partial allocation of core chores to the heavy agents: the sets $T_h \subseteq M$ are pairwise disjoint and satisfy $|T_h| = q_h - 1$ for every $h \in H$.

The prescribed quota problem asks for an allocation $A \in \mathcal{A}(q)$ that is both WEF1 and fPO.

Theorem 1. *For positive rational costs and positive entitlements, the problem of finding a quota-respecting WEF1 and fPO allocation for a prescribed quota vector q, or deciding that none exists, can be solved in time $m^{S(q)}\mathrm{poly}(n,m)$.*

Proof. For any candidate allocation $A \in \mathcal{A}(q)$ and any heavy agent $h \in H(q)$, let

$$g_h \in \arg\max_{j \in A_h} c_{hj} \quad \text{and} \quad T_h = A_h \setminus \{g_h\}.$$

Then $d_h^-(A_h) = c_h(T_h)$. Thus the algorithm enumerates all possible choices of these core sets T_h. Once the sets T_h are fixed for all $h \in H(q)$, it remains to choose one guard chore g_h for each heavy agent and one singleton chore for each agent in $R(q)$. Algorithm 1 does exactly this by a minimum-cost perfect matching.

We first prove soundness. Suppose Algorithm 1 outputs an allocation A. Agents in $R \cup E$ receive at most one chore, so $d_i^-(A_i) = 0$ for every $i \in R \cup E$. Hence all WEF1 constraints with such an agent on the left-hand side hold automatically.

Now consider a heavy agent $h \in H$. The matching assigns h the chore $\mu(h)$. Since the edge $(h, \mu(h))$ is allowed, we have $c_{h\mu(h)} \geq \beta_h = \max_{j \in T_h} c_{hj}$. Therefore $\mu(h)$ is a valid guard chore for h, and $d_h^-(A_h) = c_h(T_h)$. Consequently, $\frac{d_h^-(A_h)}{\alpha_h} = \tau_h$.

It remains to check the right-hand side of the WEF1 inequalities. If $e \in E$, then $A_e = \varnothing$, so the empty-agent test in Algorithm 1 verifies the constraints from heavy agents to empty agents. If $r \in R$, then $A_r = \{\mu(r)\}$, and the allowed-edge condition for $(r, \mu(r))$ gives

$$\tau_h \leq \frac{c_{h\mu(r)}}{\alpha_r} = \frac{c_h(A_r)}{\alpha_r} \quad \text{for every } h \in H.$$

If $k \in H$, then $A_k = T_k \cup \{\mu(k)\}$, and the allowed-edge condition for $(k, \mu(k))$ gives

$$\tau_h \leq \frac{c_h(T_k) + c_{h\mu(k)}}{\alpha_k} = \frac{c_h(A_k)}{\alpha_k} \quad \text{for every } h \in H.$$

Thus A is WEF1. The product comparison in the algorithm is equivalent to $\Phi(A) = \Gamma(q)$, because all costs are positive and B^q has minimum Φ-value in $\mathcal{A}(q)$.

We next prove completeness. Let $A^* \in \mathcal{A}(q)$ be any allocation that is WEF1 and fPO. By Lemma 1, $\Phi(A^*) = \Gamma(q)$. For every heavy agent $h \in H$, choose $g_h \in \arg\max_{j \in A_h^*} c_{hj}$ and set $T_h = A_h^* \setminus \{g_h\}$. Then $|T_h| = q_h - 1$, and the algorithm considers the tuple $(T_h)_{h \in H}$.

For this tuple, $\tau_h = \frac{c_h(T_h)}{\alpha_h} = \frac{d_h^-(A_h^*)}{\alpha_h}$. If $E \neq \varnothing$, the WEF1 constraints of A^* toward empty agents imply that the empty-agent test passes. The residual assignment in A^* gives a perfect matching on the graph constructed by the algorithm: each agent $h \in H$ is matched to g_h, and each singleton agent $r \in R$ is matched to its unique chore in A_r^*. The guard condition holds by definition of g_h, and all allowed-edge inequalities are exactly the WEF1 inequalities of A^* from heavy agents to the corresponding target agents.

Therefore a feasible perfect matching exists in this iteration. Let μ be the minimum-cost perfect matching chosen by the algorithm, and let A be the allocation induced by μ and the guessed cores. The matching induced by A^* is feasible, so $\Phi(A) \leq \Phi(A^*) = \Gamma(q)$. On the other hand, $A \in \mathcal{A}(q)$, so by the definition of $\Gamma(q)$, $\Phi(A) \geq \Gamma(q)$. Therefore, $\Phi(A) = \Gamma(q)$, and the algorithm outputs an allocation.

It remains to bound the running time. The algorithm enumerates exactly $\sum_{h \in H}(q_h - 1) = S(q)$ chores across the guessed sets $(T_h)_{h \in H}$. Hence the number of tuples is at most $m^{S(q)}$. For each tuple, the graph construction and the perfect matching computation take polynomial time. The computation of $\Gamma(q)$ is also polynomial by a minimum-cost flow formulation, with logarithmic costs handled as in Remark 1. Thus the total running time is $m^{S(q)}\text{poly}(n, m)$. □

5 Unconstrained Instances with Few Extra Chores

We next use the prescribed quota algorithm to handle unconstrained instances when the number of chores exceeds the number of agents by at most a constant. The key point is that, under strictly positive costs, any WEF1 allocation with more chores than agents must give every agent at least one chore.

Theorem 2. *Assume $c_{ij} > 0$ and $\alpha_i > 0$. For every fixed $s \geq 0$, a WEF1 and fPO allocation can be computed in polynomial time whenever $m \leq n + s$.*

Proof. First suppose $m \leq n$. Compute a minimum-cost matching that assigns each chore to a distinct agent, using edge cost $\log c_{ij}$ for assigning chore j to agent i, implemented as in Remark 1, and let A be the induced allocation. Every agent receives at most one chore, so A is WEF1. Let $q = q(A)$. Since this matching minimizes Φ over all allocations in which every agent receives at most one chore, it in particular minimizes Φ over $\mathcal{A}(q)$. Lemma 1 therefore implies that A is fPO.

Now suppose $m > n$, and write $m = n + s'$ with $1 \leq s' \leq s$. In any WEF1 allocation, every agent must receive at least one chore. Otherwise, if some agent k is empty, then another agent i receives at least two chores, and strict positivity gives us $\frac{d_i^-(A_i)}{\alpha_i} > 0 = \frac{c_i(A_k)}{\alpha_k}$, contradicting WEF1.

By Mahara's weighted-entitlement existence theorem [29], there exists a WEF1 and fPO allocation for the given entitlements. Its quota vector q has all entries positive, and thus, we have that

$$S(q) = \sum_i (q_i - 1) = m - n = s'.$$

There are $\binom{n+s'-1}{s'} = n^{O(s')}$ positive quota vectors summing to m. We enumerate them and run Theorem 1 for each. The quota vector of some WEF1+fPO allocation is among the enumerated vectors, so the algorithm outputs an allocation for at least one vector. Since $s' \leq s$, the running time is polynomial for every fixed s. □

The strict positivity assumption can be removed for the unconstrained corollary by a standard zero-cost preprocessing.

Corollary 1 (Nonnegative costs). *For every fixed $s \geq 0$, given nonnegative rational costs c_{ij} and positive rational entitlements α_i, a WEF1 and fPO allocation can be computed in polynomial time whenever $m \leq n + s$.*

Proof. While there is a chore j and an agent i with $c_{ij} = 0$, assign j permanently to i and remove j. If no chore remains, use the empty allocation on the reduced instance. Otherwise, all remaining costs are strictly positive, and the number of remaining chores is still at most $n+s$. Apply Theorem 2 to the reduced instance.

We now add back the removed chores. Consider a removed chore j that was assigned permanently to an agent i with $c_{ij} = 0$. Adding j to i's bundle does not change $c_i(A_i)$, and it also does not change $d_i^-(A_i)$: if A_i already contains a positive-cost chore for i, the maximum i-cost chore is unchanged, while if all chores in $A_i \cup \{j\}$ have zero i-cost, then $d_i^-(A_i \cup \{j\}) = 0$. For any other agent h, the cost $c_h(A_i)$ of i's bundle can only increase, which can only make the WEF1 constraint toward i easier. Thus WEF1 is preserved.

For fPO, let $\bar{A}$ be the allocation computed for the reduced instance, and let A be the final allocation after adding back the removed chores. Suppose for contradiction that a fractional allocation x of the original instance Pareto-dominates A. Let x' be the restriction of x to the chores that remain after the zero-cost removals. Then x' is a feasible fractional allocation of the reduced instance. For every agent a,

$$c_a(x'_a) \leq c_a(x_a) \leq c_a(A_a) = c_a(\bar{A}_a).$$

The first inequality uses nonnegativity of costs, and the equality uses the fact that every removed chore was added back to an agent who has zero cost for it. If x strictly decreases the cost of some agent a relative to A, then the displayed inequality is strict for that same agent. Therefore, x' Pareto-dominates $\bar{A}$, contradicting fPO of $\bar{A}$. □

6 Extensions: Typed and Identical Chores

The prescribed quota algorithm guesses actual chores in the sets T_h. When chores have few cost types, it is enough to guess how many chores of each type appear in each such set. We say that chores have p cost types if M is partitioned into $M_1, \ldots, M_p$ such that, for every type t and every two chores $j, j' \in M_t$, we have $c_{ij} = c_{ij'}$ for all agents i. Let $m_t = |M_t|$, and write $c_{i,t}$ for the cost of a type-t chore to agent i.

Theorem 3. *Assume positive rational costs and positive entitlements. If chores have p cost types, then the prescribed quota WEF1+fPO problem can be solved in time*

$$\left(\prod_{a \in H(q)} \binom{q_a + p - 2}{p - 1} \right) \mathrm{poly}(n, m).$$

In particular, the problem is polynomial-time solvable for fixed p *and fixed* $|H(q)|$*, and is fixed-parameter tractable in* $p + S(q)$*.*

Proof. For each agent $a \in H(q)$, guess the type-count vector

$$z_a = (z_{a,1}, \dots, z_{a,p}) \in \mathbb{Z}^p_{\geq 0}, \quad \sum_{t=1}^{p} z_{a,t} = q_a - 1.$$

We only keep guesses satisfying $\sum_{a \in H(q)} z_{a,t} \leq m_t$ for every type t. For a kept guess, we realize the guessed cores by choosing, for every type t, arbitrary disjoint subsets of M_t of sizes $z_{a,t}$ for the heavy agents $a \in H(q)$. This is possible precisely because $\sum_{a \in H(q)} z_{a,t} \leq m_t$. Since chores of the same type have the same cost vector, only these type counts matter for WEF1 and for the product objective. The number of such guesses is at most

$$\prod_{a \in H(q)} \binom{q_a + p - 2}{p - 1}.$$

Fix one feasible guess $z = (z_a)_{a \in H(q)}$. For any agent i, define $C_i(z_a) = \sum_{t=1}^{p} z_{a,t} c_{i,t}$. For each $a \in H(q)$, define

$$\tau_a = \frac{C_a(z_a)}{\alpha_a} \quad \text{and} \quad \beta_a = \max\{c_{a,t} : z_{a,t} > 0\},$$

where the maximum is well-defined because $a \in H(q)$ implies $\sum_t z_{a,t} = q_a - 1 \geq 1$. If $E(q) \neq \varnothing$ and some $\tau_a > 0$, then this guess cannot lead to a WEF1 allocation, since a heavy agent would WEF1-envy an empty agent.

It remains to choose one residual chore for each agent in $H(q) \cup R(q)$. Let $\rho_t = m_t - \sum_{a \in H(q)} z_{a,t}$ be the number of remaining chores of type t. We build a min-cost flow instance with supply ρ_t at each type t, and demand one at each agent-slot in $H(q) \cup R(q)$.

A type t may be sent to a singleton agent $r \in R(q)$ iff

$$\tau_a \leq \frac{c_{a,t}}{\alpha_r} \quad \text{for every } a \in H(q).$$

A type t may be sent to a heavy agent $b \in H(q)$ iff $c_{b,t} \geq \beta_b$ and

$$\tau_a \leq \frac{C_a(z_b) + c_{a,t}}{\alpha_b} \quad \text{for every } a \in H(q).$$

The first condition says that the chosen residual chore can serve as the guard chore of b. The second condition is exactly the WEF1 constraint from every heavy agent a toward b.

The edge cost of assigning a type-t chore to agent i is $\log c_{i,t}$, implemented as in Remark 1. Since the flow network has integral supplies and demands, an optimal flow is integral and therefore corresponds to an actual residual assignment of chores. If this flow instance is infeasible, we discard the current guess z.

For the fixed guess z, add the core cost $\sum_{a\in H(q)}\sum_{t=1}^{p} z_{a,t}\log c_{a,t}$ to the minimum residual-flow cost. We output the resulting allocation only if this total is $\Gamma(q)=\min\{\Phi(A) : A\in\mathcal{A}(q)\}$, where $\Gamma(q)$ is computed, again as in Remark 1, by the typed transportation problem with type supplies m_t, agent demands q_i, and edge costs $\log c_{i,t}$.

Soundness follows because the allowed edges encode exactly the guard condition and all WEF1 constraints whose left-hand side is nonzero. The equality with $\Gamma(q)$ implies fPO by Lemma 1. Completeness follows by taking any quota-respecting WEF1+fPO allocation, recording the type counts z_a of its WEF1 cores, and observing that its residual guard/singleton assignment is a feasible integral flow for the corresponding guess.

The fixed-parameter claim follows since

$$\prod_{a\in H(q)}\binom{q_a+p-2}{p-1}\le\prod_{a\in H(q)}p^{q_a-1}=p^{S(q)}\le(p+S(q))^{p+S(q)}.$$

Thus the non-polynomial part of the running time is bounded by a function only of $p+S(q)$. □

Corollary 2 (Identical chores). *Assume all chores have the same positive cost vector: for every agent i, $c_{ij}=c_i$ for all chores j. For a prescribed quota vector q, every allocation in $\mathcal{A}(q)$ is fPO. Moreover, a quota-respecting allocation is WEF1 if and only if*

$$\frac{(q_i-1)^+}{\alpha_i}\le\frac{q_k}{\alpha_k}\quad \textit{for every pair of agents } i,k.$$

Thus prescribed quota WEF1+fPO is decidable in $O(n^2)$ time for identical chores.

Proof. Since all chores are identical, every allocation with quota vector q has the same product $\prod_{i\in N}c_i^{q_i}$. Hence every allocation in $\mathcal{A}(q)$ is fPO by Lemma 1. For WEF1, $d_i^-(A_i)=(q_i-1)^+c_i$, while $c_i(A_k)=q_kc_i$. Since $c_i>0$, the WEF1 inequalities are exactly

$$\frac{(q_i-1)^+}{\alpha_i}\le\frac{q_k}{\alpha_k}\quad\forall i,k,$$

as desired. □

7 Hardness for Prescribed Quotas

The algorithm of Theorem 1 is exponential in $S(q)$. We now show that prescribed quotas become hard when this quantity is unbounded. The hardness holds even without entitlements, i.e., for ordinary EF1. Here, two agent types means that there are only two distinct cost functions, although agents of the same type may have separate quotas.

Theorem 4. *It is strongly NP-hard to decide whether there exists an allocation $A\in\mathcal{A}(q)$ that is EF1 and PO, even with positive integer costs, two agent types, and $q_i\in\{3,5\}$ for every agent i. The same hardness holds for EF1+fPO.*

Proof. We reduce from 3-Partition. The input consists of integers $a_1, \ldots, a_{3r}$ with

$$\sum_{t=1}^{3r} a_t = rB \quad \text{and} \quad B/4 < a_t < B/2$$

for every t. It is strongly NP-complete to decide whether these numbers can be partitioned into r triples, each summing to B [18].

Create one distinguished agent d and bin agents $b_1, \ldots, b_r$. Create five special chores $h_1, \ldots, h_5$ and one number chore e_t for each a_t. Set $q_d = 5$, $q_{b_s} = 3$ for every s. For the distinguished agent, set $c_d(h_\ell) = B$ and $c_d(e_t) = 4a_t$. For every bin agent b_s, set $c_{b_s}(h_\ell) = K$ and $c_{b_s}(e_t) = 4a_t$, where $K = 3B$. Since $B/4 < a_t < B/2$, we have $B < 4a_t < 2B < K$. All costs are positive integers, and all bin agents have the same cost function.

We first show that every quota-respecting PO allocation assigns all special chores to d. Suppose some special chore h_ℓ is assigned to a bin agent. Since d has quota five and there are exactly five special chores, d must receive some number chore e_t. Swapping h_ℓ and e_t changes d's cost from $4a_t$ to B, and changes the bin agent's cost from K to $4a_t$. Both agents strictly improve, and all others are unchanged. Thus the original allocation was not PO.

Now suppose a quota-respecting EF1+PO allocation exists. By the previous paragraph, d receives $\{h_1, \ldots, h_5\}$, and every bin receives three number chores. The value of d_d^- on d's bundle is

$$d_d^-(\{h_1, \ldots, h_5\}) = 5B - B = 4B.$$

For d to be EF1 toward bin b_s, we need

$$4B \le c_d(A_{b_s}) = \sum_{e_t \in A_{b_s}} 4a_t.$$

Thus the three numbers assigned to each bin have sum at least B. Since the total sum is rB, every bin must receive a triple summing exactly to B. Hence the 3-Partition instance is a yes-instance.

Conversely, suppose the 3-Partition instance has triples $T_1, \ldots, T_r$ such that $\sum_{t \in T_s} a_t = B$ for every $s \in [r]$. Assign all five special chores to d, and assign the number chores in T_s to bin agent b_s.

This allocation is EF1. The self-constraints are immediate from nonnegativity, so consider two distinct agents. Agent d's threshold is $4B$, and d's cost for every bin bundle is $4B$. A bin agent's own bundle has cost $4B$; after removing any one chore, its remaining own cost is at most $4B$. Since every other bin bundle also has cost $4B$ to this agent, the EF1 constraints between bin agents hold. It is also EF1 toward d, because its EF1 threshold is at most $4B$, while its cost for d's bundle is $5K = 15B$.

It remains to prove fPO. By Lemma 1, it is enough to prove quota-wise product optimality. In any quota-respecting allocation where a special chore is assigned to a bin, d must receive some number chore. Swapping such a pair preserves all quotas and changes the product contribution from $c_d(e_t)c_{b_s}(h_\ell) =$

$4a_tK$ to $c_d(h_\ell)c_{b_s}(e_t) = B \cdot 4a_t$, which is smaller because $B < K$. Repeating this swap strictly decreases the product until all special chores are assigned to d. Once this holds, the product is $B^5 \prod_{t=1}^{3r} 4a_t$, independent of how the number chores are assigned to the bin agents. Therefore the constructed allocation is quota-wise product-minimizing, and hence fPO. Since fPO implies PO, it is also EF1+PO.

Thus the prescribed quota EF1+PO and EF1+fPO instances are yes-instances exactly when the original 3-PARTITION instance is a yes-instance. The reduction uses only integers bounded by $3B$, so the hardness is strong. □

8 Conclusion

We give a polynomial-time algorithm for computing WEF1 and fPO allocations of indivisible chores whenever $m \leq n + s$ for fixed s. The main structural reason is specific to chores: if an agent receives at most one chore, then $d_i^-(A_i) = 0$, so only agents with at least two chores can create nontrivial WEF1 constraints. Combined with product minimality at fixed quotas, this gives us a prescribed quota algorithm running in time $m^{S(q)}\text{poly}(n, m)$.

The hardness result shows that prescribed quotas become strongly NP-hard when $S(q)$ is unbounded, even under severe restrictions on costs and quotas. This leaves two natural open questions. First, can the dependence on $S(q)$ be improved to $f(S(q))\text{poly}(n, m)$? Second, already for equal entitlements, can EF1+fPO allocations be computed in polynomial time for general unconstrained chore instances without enumerating quota vectors?

References

1. Hannaneh Akrami, Bhaskar Ray Chaudhury, Jugal Garg, Kurt Mehlhorn, and Ruta Mehta. Fair and efficient allocation of indivisible chores with surplus. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2494–2502, 2023.
2. Georgios Amanatidis, Haris Aziz, Georgios Birmpas, Aris Filos-Ratsikas, Bo Li, Hervé Moulin, Alexandros A. Voudouris, and Xiaowei Wu. Fair division of indivisible goods: Recent progress and open questions. *Artificial Intelligence*, 322(C), 2023.
3. Haris Aziz, Ioannis Caragiannis, Ayumi Igarashi, and Toby Walsh. Fair allocation of indivisible goods and chores. *Autonomous Agents and Multi-Agent Systems*, 36(3), 2022.
4. Haris Aziz, Bo Li, Hervé Moulin, and Xiaowei Wu. Algorithmic fair allocation of indivisible items: A survey and new questions. *ACM SIGecom Exchanges*, 20(1):24–40, 2022.
5. Haris Aziz, Jeremy Lindsay, Angus Ritossa, and Mashbat Suzuki. Fair allocation of two types of chores. In *Proceedings of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 143–151, 2023.
6. Siddharth Barman, Sanath Kumar Krishnamurthy, and Rohit Vaish. Finding fair and efficient allocations. In *Proceedings of the 19th ACM Conference on Economics and Computation (EC)*, pages 557–574, 2018.

7. Arpita Biswas and Siddharth Barman. Fair division under cardinality constraints. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 91–97, 2018.
8. Sylvain Bouveret and Michel Lemaître. Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *Autonomous Agents and Multi-Agent Systems*, 30(2):259–290, 2016.
9. Florian Brandl, Warut Suksompong, and Nicholas Teh. Fair division with binary valuations: Characterizations. In *Proceedings of the 19th International Symposium on Algorithmic Game Theory (SAGT)*, 2026. Extended version available at arXiv:2607.10064.
10. 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.
11. Mithun Chakraborty, Ayumi Igarashi, Warut Suksompong, and Yair Zick. Weighted envy-freeness in indivisible item allocation. *ACM Transactions on Economics and Computation*, 9(3):18:1–18:39, 2021.
12. Mithun Chakraborty, Ulrike Schmidt-Kraepelin, and Warut Suksompong. Picking sequences and monotonicity in weighted fair division. *Artificial Intelligence*, 301:103578, 2021.
13. Mithun Chakraborty, Erel Segal-Halevi, and Warut Suksompong. Weighted fairness notions for indivisible items revisited. *ACM Transactions on Economics and Computation*, 12(3):9:1–9:45, 2024.
14. Harish Chandramouleeswaran and Prajakta Nimbhorkar. Nonexistence of simultaneously EF1 and pareto optimal allocations for submodular valuations. *arXiv preprint arXiv:2607.18220*, 2026.
15. Soroush Ebadian, Dominik Peters, and Nisarg Shah. How to fairly allocate easy and difficult chores. In *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 372–380, 2022.
16. Edith Elkind, Ayumi Igarashi, and Nicholas Teh. Fair division of chores with budget constraints. In *Proceedings of the 17th International Symposium on Algorithmic Game Theory (SAGT)*, pages 55–71, 2024.
17. Uriel Feige, Ariel Sapir, and Laliv Tauber. A tight negative example for MMS fair allocations. In *International Conference on Web and Internet Economics*, pages 355–372. Springer, 2021.
18. Michael R. Garey and David S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., 1979.
19. Jugal Garg, Aniket Murhekar, and John Qin. Fair and efficient allocations of chores under bivalued preferences. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI)*, pages 5043–5050, 2022.
20. Jugal Garg, Aniket Murhekar, and John Qin. New algorithms for the fair and efficient allocation of indivisible chores. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2710–2718, 2023.
21. Jugal Garg, Aniket Murhekar, and John Qin. Weighted EF1 and PO allocations with few types of agents or chores. In *Proceedings of the 33rd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2799–2806, 2024.
22. Jugal Garg, Aniket Murhekar, and John Qin. Constant-factor EFX exists for chores. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing (STOC)*, pages 1580–1589, 2025.
23. Kevin Hsu. Existence of MMS allocations with mixed manna. *arXiv preprint arXiv:2401.07490*, 2024.

24. Yasushi Kawase and Ryoga Mahara. Fair and efficient balanced allocation for indivisible goods. In *Proceedings of the 40th AAAI Conference on Artificial Intelligence (AAAI)*, pages 17067–17075, 2026.
25. David Kurokawa, Ariel Procaccia, and Junxing Wang. When can the maximin share guarantee be guaranteed? In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI)*, pages 523–529, 2016.
26. Eugene Lim, Tzeh Yuan Neoh, and Nicholas Teh. The cost of EFX: Generalized-mean welfare and complexity dichotomies with few surplus items. *arXiv preprint arXiv:2601.12849*, 2026.
27. 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.
28. Simon Mackenzie and Mashbat Suzuki. When one good is not enough: EF1 and pareto optimality are not compatible for submodular valuations. *arXiv preprint arXiv:2607.17811*, 2026.
29. Ryoga Mahara. Existence of fair and efficient allocation of indivisible chores. In *Proceedings of the 37th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 6742–6766, 2026.
30. Luisa Montanari, Ulrike Schmidt-Kraepelin, Warut Suksompong, and Nicholas Teh. Weighted envy-freeness for submodular valuations. *Social Choice and Welfare*, 65:983–1016, 2025.
31. Tzeh Yuan Neoh and Nicholas Teh. Understanding EFX allocations: Counting and variants. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence (AAAI)*, pages 14036–14044, 2025.
32. Max Springer, MohammadTaghi Hajiaghayi, and Hadi Yami. Almost envy-free allocations of indivisible goods or chores with entitlements. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence (AAAI)*, pages 9901–9908, 2024.
33. Warut Suksompong. Constraints in fair division. *ACM SIGecom Exchanges*, 19(2):46–61, 2021.
34. Warut Suksompong. Weighted fair division of indivisible items: a review. *Information Processing Letters*, 187:106519, 2025.
35. Warut Suksompong and Nicholas Teh. On maximum weighted Nash welfare for binary valuations. *Mathematical Social Sciences*, 117:101–108, 2022.
36. Warut Suksompong and Nicholas Teh. Weighted fair division with matroid-rank valuations: Monotonicity and strategyproofness. *Mathematical Social Sciences*, 126:48–59, 2023.
37. Nicholas Teh. Fair division with strictly increasing valuations: A tight threshold for two-agent EF1 and PO. *arXiv preprint arXiv:2607.23367*, 2026.
38. Xiaowei Wu, Cong Zhang, and Shengwei Zhou. Weighted EF1 allocations for indivisible chores. *Artificial Intelligence*, 347:104386, 2025.