Cost Sharing over Combinatorial Domains

GEORGIOS BIRMPAS, Sapienza University of Rome
EVANGELOS MARKAKIS, Athens University of Economics and Business
GUIDO SCHÄFER, Centrum Wiskunde & Informatica (CWI) and University of Amsterdam

We study the problem of designing cost-sharing mechanisms for combinatorial domains. Suppose that multiple items or services are available to be shared among a set of interested agents. The outcome of a mechanism in this setting consists of an assignment, determining for each item the set of players who are granted service, together with respective payments. Although there are several works studying specialized versions of such problems, there has been almost no progress for general combinatorial cost-sharing domains until recently [9]. Still, many questions about the interplay between strategyproofness, cost recovery, and economic efficiency remain unanswered.

The main goal of our work is to further understand this interplay in terms of budget balance and social cost approximation. Towards this, we provide a refinement of cross-monotonicity (which we term *trace-monotonicity*) that is applicable to iterative mechanisms. The trace here refers to the order in which players become finalized. On top of this, we also provide two parameterizations (complementary to a certain extent) of cost functions, which capture the behavior of their average cost-shares.

Based on our trace-monotonicity property, we design an Iterative Ascending Cost-Sharing Mechanism, which is applicable to the combinatorial cost-sharing setting with symmetric submodular valuations. Using our first cost function parameterization, we identify conditions under which our mechanism is weakly group-strategyproof, $O(1)$-budget-balanced, and $O(H_n)$-approximate with respect to the social cost. Furthermore, we show that our mechanism is budget-balanced and H_n-approximate if both the valuations and the cost functions are symmetric submodular; given existing impossibility results, this is best possible.

Finally, we consider general valuation functions and exploit our second parameterization to derive a more fine-grained analysis of the Sequential Mechanism introduced by Moulin. This mechanism is budget balanced by construction, but in general, only guarantees a poor social cost approximation of n. We identify conditions under which the mechanism achieves improved social cost approximation guarantees. In particular, we derive improved mechanisms for fundamental cost-sharing problems, including Vertex Cover and Set Cover.

A preliminary version of this work appeared in [2].

Part of this work was done while Georgios Birmpas was an intern of the Networks and Optimization group at Centrum Wiskunde & Informatica. Georgios Birmpas was supported by the ERC Advanced Grant 321171 (ALGAME) “Algorithms, Games, Mechanisms, and the Price of Anarchy”, the ERC Advanced Grant 788893 (AMDROMA) “Algorithmic and Mechanism Design Research in Online Markets”, and the MIUR PRIN project (ALGADIMAR) “Algorithms, Games, and Digital Markets”. The research was also supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the “First Call for H.F.R.I. Research Projects to support Faculty members and Researchers and the procurement of high-cost research equipment grant” (Project Number: HFRI-FM17-3512).

Authors’ addresses: G. Birmpas, Sapienza University of Rome, Address: Via Ariosto 25, Postal Code: 00185, Rome, Lazio, Italy; email: gebirbas@gmail.com; E. Markakis, Athens University of Economics and Business, Address: 28 Octovriou 76, Postal Code: 104 34, Athens, Attica, Greece; email: markakis@aueb.gr; G. Schäfer, Centrum Wiskunde & Informatica (CWI) and University of Amsterdam, Address: Science Park 123, Postal Code: 1098 XG, Amsterdam, North Holland, The Netherlands; email: g.schaefer@cwi.nl.

2167-8375/2022/04-ART4 $15.00
https://doi.org/10.1145/3505586

CCS Concepts: • **Theory of computation → Algorithmic mechanism design;**

Additional Key Words and Phrases: Cost-sharing, strategy-proofness, social cost approximation, symmetric functions, subadditive functions

ACM Reference format:
Georgios Birmpas, Evangelos Markakis, and Guido Schäfer. 2022. Cost Sharing over Combinatorial Domains. *ACM Trans. Econ. Comput.* 10, 1, Article 4 (April 2022), 26 pages.
https://doi.org/10.1145/3505586

1 INTRODUCTION

How to share the cost of a common service (or public good) among a set of interested agents N constitutes a fundamental problem in mechanism design that has been studied intensively for at least two decades. Several deep and significant advancements have been achieved throughout this period, notably also combining classical mechanism design objectives (such as incentive compatibility, economic efficiency, and so on) with theoretical computer science objectives (such as approximability and computational efficiency).

However, in the vast majority of the cost-sharing models that have been proposed and analyzed in the literature, it is assumed that the mechanism designer is offering a single service and that each agent has a private value describing the willingness to pay for the service. At the same time, there is also a publicly known cost function c, which describes the total cost $c(S)$ for offering the service to each possible subset of agents $S \subseteq N$. Said differently, this results in a single-parameter mechanism design problem, where the goal is to select a subset of the players that will be granted service, subject to covering the cost and achieving an economically efficient outcome. This is a very fundamental setting, with several motivating examples that range from sharing water resource and flood control management costs [24, 26], to sharing the cost of public investments [18], as well as multicast messages over a computer network [16] (e.g., sharing multicast transmission costs of video or other filetypes). We refer to [21] for further references on application areas.

Although significant progress has been made for such single-parameter domains, moving towards more general *combinatorial domains* has been almost elusive so far. Imagine that there are multiple goods or services to be shared among the agents who have more complex valuation functions, expressing their willingness to pay for different subsets (or bundles) of goods. The cost function now depends on the subsets of agents sharing each of the items. In this setting, an outcome of a mechanism consists of an allocation, which specifies for each agent the goods for which she is granted service, together with a payment scheme. The motivating examples mentioned above for the single-parameter domain naturally extend to the combinatorial setting, as users are often interested in sharing more than a single good or service. Another example, also mentioned in [9], is the sharing of maintenance (or construction) costs of runways of (possibly) different airports among airlines. Furthermore, this setting becomes more and more relevant for participatory sensing applications, where multiple users can require access to sensor information from various sensor types at the same time; see, e.g., the models considered in [1].

The desired properties in designing a cost-sharing mechanism (be it combinatorial or not) are three-fold: (i) *group-strategyproofness*: we would like resistance to misreporting preferences by individual agents or coalitions, (ii) *budget-balance*: the payments of the players should cover the incurred cost, (iii) *economic efficiency*: the allocation should maximize a measure of social efficiency. The fundamental results in [12, 22] rule out the possibility that all three properties can be achieved. As a result, if we insist on any variant of strategyproofness, we are forced to settle with approximate notions of at least one of the other two criteria. In this context, approximate budget balance means that the mechanism may overcharge the agents, but not by too much. In terms of efficiency, considering a social cost objective instead of the classical social welfare objective (definitions are given in Section 2) seems more amenable for multiplicative approximation guarantees.

These adapted objectives have been investigated thoroughly for single-parameter problems, especially for cost-sharing variants of well-known optimization problems. In the context of more general combinatorial cost-sharing mechanisms, a restricted model with multiple levels of service was first studied in [19]. Ever since, for almost a decade, there was no additional progress along these lines. It was only recently that a step forward was made by Dobzinski and Ovadia [9]. In their work, they introduce a combinatorial cost-sharing model and derive the first mechanisms guaranteeing good budget balance and social cost approximation guarantees for different classes of valuation and cost functions. As already pointed out in [9]; however, several important questions concerning our understanding of the approximability of these objectives remain open and deserve further study. This constitutes the starting point of our investigations reported in this work.

1.1 Our Contributions

We make further advancements on the design and analysis of mechanisms for combinatorial cost-sharing models. To begin with, the analysis of the mechanisms, we study asks for new conceptual ideas (which might be interesting on their own):

- We first provide a refinement of the well-known notion of *cross-monotonic* cost-sharing functions, which is key in the intensively studied class of Moulin–Shenker mechanisms [21] for the single-parameter domain. We introduce the notion of *trace-monotonic* cost-sharing functions, which is applicable for mechanisms that proceed iteratively and evict agents one-by-one. Trace-monotonicity formalizes the fact that the cost-shares observed by a player for an item do not decrease throughout the *course* of the mechanism. That is, these cost shares may depend on the specific order (or *trace* as we will call it) in which the mechanism considers the agents.
- We identify two different and, to some extent, complementary (see the discussion after Definition 4.2) parameterizations of the cost functions. Intuitively, these parameters measure the "variance" of the average cost-share $c(S)/|S|$ overall agent sets $S \subseteq N$. We introduce two such notions, which we term *α-average decreasing* and *α-average min-bounded* (see Definitions 3.3 and 4.2, respectively). We note that for every cost function, there exist respective values of α (possibly different for each definition) for which these properties are satisfied. These definitions provide an alternative way to classify cost functions and their respective approximation guarantees in terms of budget balance and social cost.

In Section 3, we derive an **Iterative Ascending Cost-Sharing Mechanism** (**IACSM**), which can be seen as a (non-trivial) adaptation of the Moulin–Shenker mechanisms from the binary accept/reject setting [21] to the combinatorial cost-sharing domain. Our notion of trace-monotonic cost shares play a crucial role here. We show that our proposed mechanism is applicable for any non-decreasing cost function and for symmetric submodular valuations (i.e., submodular functions whose value depends only on the cardinality of the set). By exploiting the first parameterization of α-average decreasing cost functions, our main result of Section 3 is that for $\alpha = O(1)$ our mechanism is polynomial-time, weakly group-strategyproof, $O(1)$-budget-balanced and $O(H_n)$-approximate with respect to social cost, where n is the number of agents.[1] As a consequence, if both the valuation and cost functions are symmetric submodular ($\alpha = 1$), the mechanism is budget-balanced and H_n-approximate. This is best possible even for a single item, as there exist

[1] We use H_n to denote the *nth Harmonic number* defined as $H_n = 1 + \frac{1}{2} + \cdots + \frac{1}{n}$.

corresponding inapproximability results by Dobzinski et al. [8]. Prior to our work, the best-known mechanism for symmetric submodular valuation and cost functions is H_n-budget balanced and H_n-approximate [9]. We anticipate that further extensions and generalizations might be feasible through our framework and this type of ascending mechanisms.

In Section 4, we exploit our second parameterization of α-average min-bounded cost functions, and provide results for general valuation functions. As it turns out, our parameterization enables us to obtain a more fine-grained analysis of the Sequential Mechanism introduced by Moulin [20]. This mechanism is budget-balanced by construction, but in general, only guarantees a poor social cost approximation of factor n. We show that for α-average min-bounded cost functions with $\alpha = O(1)$, the Sequential Mechanism is budget balanced and H_n-approximate with respect to social cost. Interestingly, this result does not even require monotonicity of the valuation functions. In addition, we can push our results even a bit further by introducing a refinement of this class of cost functions (see Definition 4.7) for which we show that the Sequential Mechanism is $O(1)$-approximate. The refinement allows us to obtain improved mechanisms for several cost functions originating from combinatorial optimization problems. For example, our result implies that the Sequential Mechanism is d-approximate for certain cost-sharing variants of Vertex Cover and Set Cover, where d is the maximum degree of a node or the maximum size of a set, respectively; this improves upon existing results, even in the well-studied single-item case, when d is constant.

In general, the two parameterizations of the cost functions introduced in this article seem to be suitable means to accurately capture the approximation guarantees of both the Iterative Ascending Cost-Sharing Mechanism of Section 3 and the Sequential Mechanism of Section 4. In fact, we have not managed to identify *natural* examples of cost functions for which neither of our mechanisms admits an $O(H_n)$-approximation. Even though we can show that cost functions exist that, for every constant α, are neither α-average decreasing nor α-average min-bounded, they seem rather artificial (see Proposition 5.1 and the discussion in Section 5 for more details). As such, these parameterizations help us to narrow down the class of cost functions, which are not yet known to admit a good social cost approximation and enhance our understanding towards further progress in combinatorial cost-sharing.

1.2 Related Work

For the single-item setting and with submodular cost functions, the best-known group-strategyproof and budget-balanced cost-sharing mechanism is arguably the Shapley value mechanism, introduced by Moulin and Shenker [20, 21]. This was also the first work that tried to quantify the efficiency loss of budget balanced cost-sharing mechanisms. Later, Feigenbaum et al. [11] showed that if one insists on truthfulness, there is no mechanism that achieves a finite approximation of the social welfare objective, even if one relaxes the budget balance property to cost recovery. To overcome this impossibility result, Roughgarden et al. [23] introduced the notion of *social cost* as an alternative means to quantify the efficiency of a mechanism. In the same work, they showed that the Shapley value mechanism is H_n-approximate with respect to this objective. Dobzinski et al. [8] established another impossibility result for the social cost objective, and showed that every mechanism satisfying truthfulness and cost recovery cannot achieve a social cost approximation guarantee better than $\Omega(\log n)$. The problem of deriving mechanisms with the best possible budget balance and social cost approximation guarantees for different cost functions arising from combinatorial optimization problems has been extensively studied in various works, see e.g., [3–5, 14].

Moving beyond the single-item case, Mehta et al. [19] introduced a new family of truthful mechanisms (called *acyclic mechanisms*), which apply to general demand settings of multiple identical items when players have symmetric submodular valuations. For additional works that consider the general demand setting, the reader is referred to [3, 4, 7, 20]. Birmpas et al. [1] also studied families of valuation and cost functions for the multiple item setting, under cost-sharing models that are motivated by applications in participatory sensing environments.

Most related to our work is the recent work by Dobzinski and Ovadia [9]. To the best of our knowledge, this is the only prior work that considers a more general approach for combinatorial cost-sharing. They studied a multi-parameter setting and proposed a new VCG-based mechanism. Basically, the idea is to run a VCG mechanism [6, 13, 25] with respect to a modified objective function, which is defined as the sum of the player valuations minus a potential. Intuitively, the latter ensures that the payments computed by the mechanism cover the actual cost. They showed that this mechanism is strategyproof and H_n-approximate with respect to social cost. They also identified several classes of valuation and cost functions for which the mechanism is H_n-budget balanced. In particular, this is the case if the valuation and cost functions are symmetric.[2] Additionally, they established that their mechanism is optimal with respect to the social cost approximation among all symmetric VCG-based mechanisms that always cover the cost.

2 DEFINITIONS AND NOTATION

We assume there is a set $N = \{1, 2, \dots, n\}$ of players and a set $M = \{1, 2, \dots, m\}$ of items. Each item can be viewed as a public good or some service that can be shared by the players. Each player i has a *private valuation function* $v_i : 2^M \rightarrow \mathbb{R}_{\geq 0}$ specifying the value that she derives from each subset of items.

A *cost-sharing mechanism* takes as input the declared (possibly false) valuation functions $b = (b_i)_{i \in N}$ of the players and outputs: (i) an allocation that determines which players share each item, and (ii) a payment p_i for each player i. An allocation is denoted by a tuple $A = (A_1, \dots, A_n)$, where $A_i \subseteq M$ is the set of items provided to player i. For notational convenience, we also represent an allocation $A = (A_i)_{i \in N}$ as a tuple over the items space $(T_1, \dots, T_m)$ such that for every item $j \in M$, $T_j \subseteq N$ is the subset of players sharing item j, i.e., $T_j = \{i \in N \ : \ j \in A_i\}$.

In this article, we consider mostly *separable* cost functions. In the *separable setting*, we assume that the overall cost of an allocation decomposes into the cost for providing each item separately. Hence, every item j is associated to a known cost function $c_j : 2^N \rightarrow \mathbb{R}_{\geq 0}$, which specifies for each set of players $T \subseteq N$, the cost $c_j(T)$ of providing item j to the players in T. Thus, the total cost of an allocation A is defined as

$$C(A) = \sum_{j \in M} c_j(T_j). \tag{1}$$

We note that separability is applicable in any scenario where there is no correlation between the costs of the items (the agents may still however exhibit dependencies among the items, in their valuation functions, separability involves only the costs). In Section 4.3, we also consider the *non-separable setting*, where we are given a more general cost function $C : (2^M)^n \rightarrow \mathbb{R}_{\geq 0}$, specifying for every allocation $A = (A_i)_{i \in N}$ the corresponding cost $C(A)$. Non-separable functions can capture dependencies among different items.

We assume that the utility functions of the players are *quasilinear*, i.e., given an allocation $A = (A_i)_{i \in N}$ and payments $p = (p_i)_{i \in N}$ determined by the mechanism for valuation functions $v = (v_i)_{i \in N}$, the utility of player i is defined as $u_i(v) = v_i(A_i) - p_i$. All our mechanisms have ***no positive transfers*** (**NPT**), i.e., $p_i \geq 0$, and satisfy ***individual rationality*** (**IR**), i.e., $p_i \leq v_i(A_i)$.

[2] We note that their definition of symmetry for the cost function differs from the one we use here.

In addition to the above, we are also interested in the following properties:

- **Weak Group-Strategyproofness (WGSP):** We insist on a stronger notion of resistance to manipulation than truthfulness: A mechanism is *weakly group-strategyproof* if there is no deviation by a coalition of players that makes all its members strictly better off. More formally, we require that for every coalition $Q \subseteq N$ of players, every profile v_{-Q} of the other players, there is no deviation b_Q of the players in Q such that
$$u_i(b_Q, v_{-Q}) > u_i(v_Q, v_{-Q}) \quad \text{for every } i \in Q, \tag{2}$$
where v_Q is the profile of the actual valuation functions of Q.
- **Budget Balance:** We are interested in mechanisms whose payments cover the allocation cost, ideally exactly. However, the latter is not always possible as it may be incompatible with the other objectives. We, therefore, consider an approximate budget balance notion: A mechanism is *β-budget-balanced* ($\beta \geq 1$) if for every valuation profile $v = (v_i)_{i \in N}$, the outcome (A, p) computed by the mechanism satisfies
$$C(A) \leq \sum_{i \in N} p_i \leq \beta \cdot C(A). \tag{3}$$
Clearly, we want β to be as small as possible to not overcharge players too much for covering the cost. We say that the mechanism is *budget balanced* if $\beta = 1$.
- **Economic Efficiency:** Our goal is to compute outcomes that are (approximately) efficient. To this aim, we use the *social cost objective*, originally defined in [23]. Adapted to our combinatorial setting, the *social cost* of an allocation $A = (A_i)_{i \in N}$ is defined as the actual cost of the outcome plus the value missed by not serving all items to all players, i.e.,
$$\pi(A) = \sum_{j \in M} c_j(T_j) + \sum_{i \in N} [v_i(M) - v_i(A_i)].^{3} \tag{4}$$
A mechanism is said to be *α-approximate* with respect to the social cost objective if for every valuation profile $v = (v_i)_{i \in N}$, the allocation A output by the mechanism satisfies
$$\pi(A) \leq \alpha \cdot \pi(A^*), \tag{5}$$
where A^* is an allocation of minimum social cost.

We assume that both the valuation functions $(v_i)_{i \in N}$ and the cost functions $(c_j)_{j \in M}$ are non-decreasing (see below for formal definitions). Furthermore, we focus on certain classes of valuation and cost functions: More specifically, we consider *submodular* and *subadditive* cost functions, both naturally modeling economies of scale. As to the valuation functions, we consider *submodular* valuation functions in Section 3 and general valuation functions in Section 4. Furthermore, the class of *symmetric XOS* functions play a prominent role in Section 3.[4] Below we summarize all relevant definitions (see also Lehmann et al. [17]).

Definition 2.1. Let $f : 2^U \rightarrow \mathbb{R}_{\geq 0}$ be a function defined over subsets of a universe U.

(1) f is *non-decreasing* if $f(S) \leq f(T)$ for every $S \subseteq T \subseteq U$.
(2) f is *symmetric* if $f(S) = f(T)$ for every $S, T \subseteq U$ with $|S| = |T|$.
(3) f is *submodular* if $f(S \cup \{i\}) - f(S) \geq f(T \cup \{i\}) - f(T)$ for every $S \subseteq T \subseteq U$ and $i \notin S$.
(4) f is *XOS* if there are additive functions $f^1, \dots, f^k$ such that $f(S) = \max_{i \in [k]} f^i(S)$ for all $S \subseteq U$.
(5) f is *subadditive* if $f(S \cup T) \leq f(S) + f(T)$ for every $S, T \subseteq U$.

[3] Note that this adaptation was proposed in [9].
[4] It is not hard to verify that these functions can equivalently be defined as stated in Definition 2.1 (see also [10]).

ALGORITHM 1: Iterative Ascending Cost Sharing Mechanism (IACSM)

```
Input: Declared valuation functions $(b_i)_{i \in N}$.
Output: Allocation $A = (A_i)_{i \in N}$ and payments $p = (p_i)_{i \in N}$.
Initialization: Let $X = N$ be the set of active players and define $T_j = N$ for every item $j \in M$.
while $X \neq \emptyset$ do
    Compute an optimal bundle $A_i$ for every player $i \in X$:
        $A_i \in \arg\max_{S \subseteq M} \{b_i(S) - p_i(S)\}, \quad \text{where} \quad p_i(S) = \sum_{j \in S} \chi_j(T_j)$    (6)
    (If there are several optimal bundles, resolve ties as described in Section 3.1.)
    Let $i^* \in X$ be a player such that $|A_{i^*}| \leq |A_i|$ for every $i \in X$.
    Assign the items in $A_{i^*}$ to player $i^*$ and remove player $i^*$ from $X$.
    For every item $j \in M \setminus A_{i^*}$, set $T_j = T_j \setminus \{i^*\}$, and update the cost shares $\chi_j(T_j)$.
return $A = (A_i)_{i \in N}$ and $p = (p_i)_{i \in N}$, where $p_i = \sum_{j \in A_i} \chi_j(T_j)$.
```

(6) f is *symmetric XOS* if it is symmetric and $f(S)/|S| \geq f(T)/|T|$ for every $S \subseteq T \subseteq U$.

Some of our mechanisms make use of cross-monotonic cost-sharing functions. Intuitively, cross-monotonicity is a quite natural property, which requires that the cost share of a player increases as the set of players who share the cost becomes smaller. We give a formal definition below.

Definition 2.2. Let $c : 2^N \to \mathbb{R}_{\geq 0}$ be a cost function. A *cost-sharing function*[5] $\chi : N \times 2^N \to \mathbb{R}_{\geq 0}$ with respect to c specifies for each subset $S \subseteq N$ and every player $i \in S$ a non-negative cost share $\chi(i, S)$ such that $\sum_{i \in S} \chi(i, S) = c(S)$.[6] χ is *cross-monotonic* if for all $S \subseteq T \subseteq N$ and every $i \in S$, we have $\chi(i, S) \geq \chi(i, T)$.

3 AN ITERATIVE ASCENDING COST-SHARING MECHANISM

In this section, we present our *IACSM* for the combinatorial cost-sharing setting with symmetric submodular valuations and general cost functions. In particular, we begin with Sections 3.1 and 3.2, where we provide a generic description of our mechanism, and identify two properties which are sufficient for our main result. We then proceed with Section 3.3, where we show that these properties are satisfied if the valuations are symmetric submodular. In Section 3.4, we define the class of cost functions under which we will study IACSM, and we state our main result (Theorem 3.4). Finally, Sections 3.5–3.7 are devoted to the proof of Theorem 3.4 and the properties of IACSM.

3.1 Definition of IACSM

Mechanism IACSM can be viewed as a generalization of the *Moulin–Shenker* mechanism [21] to the combinatorial setting in the sense that it simulates in parallel an ascending iterative auction for each item. To our knowledge this is the first ascending price mechanism for the combinatorial setting which is not VCG-based and as we will describe below, this adaptation is not straightforward since there are several obstacles we need to overcome. A description of our mechanism IACSM is given in Algorithm 1.

The mechanism maintains a set of *active* players X and for each item $j \in M$ a set of players T_j who are *tentatively* assigned to j. Initially, each player is active and tentatively assigned to all the items, i.e., $X = N$ and $T_j = N$ for all $j \in M$. The mechanism then proceeds in iterations. In each iteration, each item j is offered to each active player $i \in X$ at a price that only depends on the

[5] We stress here that it is also possible for cost-sharing methods to overcharge, something that leads to approximate budget balanced mechanisms.

[6] For notational convenience, we define $\chi(i, S) = \infty$ for $i \notin S$.

set of tentatively assigned players T_j. For this, we use a player-independent *cost-sharing function* $\chi_j(\cdot, T_j)$ for every item j, and since we require that $\chi_j(i, T_j) = \chi_j(k, T_j)$ for every $i, k \in T_j$, we will simply denote by $\chi_j(T_j)$ the cost share that each player $i \in T_j$ tentatively assigned to j has to pay. Based on these cost shares, every active player $i \in X$ computes an *optimal bundle* A_i with respect to the payments $p_i(\cdot)$, as defined in Equation (3). If there are ties, we resolve them according to the following *tie-breaking rule*: If there are several optimal bundles, then player i chooses one of maximum size. If there are multiple optimal bundles of maximum size k, then she chooses the bundle consisting of the k cheapest items (where ties between equal cost-share items are resolved consistently, say by an index of the items).

After determining the optimal bundle for each active player, the mechanism then chooses an active player i^* whose optimal bundle has a minimum size. Again, we break ties consistently, say by an index of the players. The items in A_{i^*} are assigned to player i^* and i^* becomes inactive. Finally, for every item j, which is not part of the optimal bundle A_{i^*}, i^* is removed from the tentative set T_j. The mechanism terminates when all players are inactive.

We briefly review the classic example of *excludable public good* to provide some intuition for our mechanism.

Example 1. Suppose the n players are interested in a single item (*i.e.,* $M = \{1\}$) and let $C(S) = 1$ for every $S \subseteq N$ with $S \neq \emptyset$ and $C(\emptyset) = 0$.[7] The valuation function v_i of each player i can be expressed by a single non-negative value w_i such that $v_i(\{1\}) = w_i$ (i gets the item) and $v_i(\emptyset) = 0$ (i does not get the item). Suppose the cost-sharing function χ is defined as the average cost, *i.e.,* $\chi(S) = C(S)/|S|$ for every $S \subseteq N$; note that this cost-sharing function is indeed player-independent.

Our mechanism IACSM then starts by tentatively assigning all players in N to the item, *i.e.,* $T = N$. The cost $C(T)$ for serving the players in T is shared equally among the players in T. Each active player i will thus choose the optimal (singleton) bundle $A_i = \{1\}$ if $w_i \geq 1/|T|$ and $A_i = \emptyset$ otherwise. As a consequence, the mechanism first iteratively evicts all players that cannot afford the respective average cost share from the set of active players, until $w_i \geq 1/|T|$ for all $i \in T$. It then iteratively assigns the item to the remaining players in T, each paying a share of $1/|T|$.

Our mechanism reduces to the well-known Moulin–Shenker mechanism (also known as the *Shapley value mechanism*) in this case. This mechanism is known to be group-strategyproof, budget balanced, and H_n-approximate [21, 23]. Note that the cost-share that each active player observes throughout the course of the mechanism is non-decreasing; this is a property that we will crucially exploit below. Another useful property here is that the cost function satisfies $\alpha \cdot C(S)/|S| \geq C(T)/|T|$ for every $S \subseteq T \subseteq N$ with $\alpha = 1$; we also say that the cost function is α-average decreasing (formalized in Definition 3.3). These properties will enable us to show that our mechanism (even for more general combinatorial domains) is weakly group-strategyproof, budget balanced, and H_n-approximate (see Corollary 3.5).

3.2 Two Crucial Properties

In this section, we identify two crucial properties that our mechanism has to satisfy for our main result to go through. To formalize these properties, we introduce first some more notation.

Trace of IACSM. Note that the execution of our mechanism IACSM on an instance of the problem induces an order $\tau = (\tau_1, \ldots, \tau_n)$ on the players. Without loss of generality, we may assume that the players are renamed such that $\tau = (1, \ldots, n)$, i.e., player i becomes inactive in iteration i; however, we emphasize that this order is determined by the run of our mechanism.

[7] Note that we are considering a single-parameter setting here. For notational convenience, we simply omit all subscripts referring to the single item.

The order $\tau = (1, \ldots, n)$ together with the final bundle A_i assigned to each player i at the end of iteration i induces an order of player withdrawals for each item j. More precisely, for every $j \in M$, we let τ_j be the subsequence of τ consisting only of the players who withdrew from item j (at the end of the iteration when they became inactive). We refer to τ_j as the *trace of item j*. Recall that initially $T_j = N$, and hence all players are tentatively assigned to j. The length of τ_j can vary from 0, when nobody withdraws from item j and τ_j is the null sequence, all the way to n, when everybody withdraws from j and $\tau_j = \tau$. Given a trace τ_j in the form $\tau_j = (i_1, i_2, \ldots, i_\ell)$ and $k \in \{0, 1, \ldots, |\tau_j|\}$, let $R_j^k = N \setminus \{i_1, i_2, \ldots, i_k\}$; define $R_j^0 = N$. Note that the set R_j^k is precisely the set of players tentatively assigned to j after k players have withdrawn from j during the execution of the mechanism. We note that the notion of trace is valid also for any other iterative mechanism where the assignment of one player becomes finalized at each iteration, e.g., [19].

Trace-monotonic cost-sharing functions. We introduce a new property of cost-sharing functions which will turn out to be crucial below. Intuitively, it is a refinement of the standard cross-monotonicity property, which has to hold only for *certain* subsets of players encountered by the mechanism, namely, for the sets $\{R_j^k\}$. More precisely, given a trace τ_j for an item $j \in M$, we say that the cost-sharing function χ_j is *cross-monotonic with respect to* τ_j (or, *trace-monotonic* for short), if

$$\forall k \in \{0, \ldots, |\tau_j| - 1\} : \qquad \chi_j\left(R_j^k\right) \le \chi_j\left(R_j^{k+1}\right).$$

Note that this ensures that the cost share of item j (weakly) increases during the execution of the mechanism, as we consider the sequence of sets

$$R_j^0 \supset R_j^1 \supset \cdots \supset R_j^{|\tau_j|}.$$

A subtle point here is that the definition of the cost-share $\chi_j(R_j^k)$ may not only depend on the set of players R_j^k, but also on the trace τ_j specifying how the set R_j^k has been reached by the mechanism.[8] It will become clear below that this additional flexibility enables us to implement our mechanism for *arbitrary* cost functions.

Properties (P1) and (P2). Our first property is rather intuitive: An item j needs to be offered to all active players at the same price, and this price can only increase in subsequent iterations. In particular, this ensures that if at the end of iteration i, player i withdraws from an item $j \in M \setminus A_i$, then the price of j for the remaining players in $T_j \setminus \{i\}$ does not decrease. This is crucial to achieve strategyproofness, and it is captured precisely by trace-monotonic cost-sharing functions.[9]

(P1) For each item $j \in M$ the cost-sharing function χ_j is trace-monotonic for every trace τ_j.

The first property alone is not sufficient to ensure that our mechanism IACSM is weakly group-strategyproof (or even strategyproof). Additionally, we need to enforce the following refinement property on the final bundles assigned to the players. We prove below that Property (P2) is satisfied for symmetric submodular valuation functions.

(P2) The final bundles $(A_i)_{i \in N}$ assigned to the players satisfy the following *refinement property*: $A_i \subseteq A_{i+1}$ for every $i \in \{1, \ldots, n-1\}$.

[8] Notationally, we would have to write here $\chi_j^{\tau_j}$ to indicate the dependency on τ_j. However, in the analysis, we focus on a fixed trace produced by an execution of the mechanism and omit the explicit reference to it for notational convenience.

[9] Note that we have to require trace-monotonicity with respect to an *arbitrary* trace of item j; here, because we cannot control the trace τ_j that will be realized by IACSM.

3.3 Feasibility of (P1) and (P2)

We next define the cost-sharing function that we use. The intuition is as follows: Suppose that $S = T_j$ is the set of players who are tentatively allocated to item j at the beginning of iteration i for some $i \in [n]$. Ideally, we would like to charge the average cost $c_j(S)/|S|$ to each player in S, but we cannot simply do this because the average cost might decrease with respect to the previous iteration, and this will destroy Property (P1). Given our new notion of trace-monotonicity, we can resolve this by defining the cost share of item j as the maximum average cost overall player sets, which were tentatively allocated to j so far.

More formally, let τ_j be the trace of item j induced by IACSM when executed on a given instance. Let S be the set of players tentatively assigned to item j at the beginning of iteration i, and fix k such that $R_j^k = S$ (by the definition of our mechanism, such a k must exist and $k \leq i-1$). We define

$$\chi_j(S) = \max_{\ell \in \{0,\ldots,k\}} \frac{c_j\left(R_j^\ell\right)}{|n-\ell|}. \tag{7}$$

Note that by using this definition, we may end up overcharging the actual cost $c_j(S)$ of item j in the sense that $|S| \cdot \chi_j(S) > c_j(S)$. As we show in Section 3.4, the budget balance and social cost approximation guarantees depend on the magnitude by which we might overcharge.

It is now trivial to show that Property (P1) holds.

Lemma 3.1. *Consider some item $j \in M$ and let $c_j : 2^N \rightarrow \mathbb{R}_{\geq 0}$ be an arbitrary cost function. Let τ_j be an arbitrary trace of j. The cost-sharing function χ_j defined in Equation (7) is trace-monotonic.*

Proof. By definition (7), we have for every $k \in \{0, \ldots, |\tau_j| - 1\}$

$$\chi_j(R_j^k) = \max_{\ell \in \{0,\ldots,k\}} \frac{c_j\left(R_j^\ell\right)}{|n-\ell|} \leq \max_{\ell \in \{0,\ldots,k+1\}} \frac{c_j\left(R_j^\ell\right)}{|n-\ell|} = \chi_j\left(R_j^{k+1}\right).$$

□

We turn to Property (P2). In general, it seems difficult to guarantee (P2), but it is not hard to see that it holds if the valuation functions are symmetric submodular.

Lemma 3.2. *Suppose the valuation functions are symmetric submodular. Then $A_i \subseteq A_{i+1}$ for every $i \in \{1, \ldots, n-1\}$.*

Proof. Fix some $i \in \{1, \ldots, n-1\}$ and consider players i and $i+1$. Note that both i and $i+1$ are active at the beginning of iteration i. Let A_i and A'_{i+1} be the optimal bundles chosen by i and $i+1$ in iteration i, respectively. Because i is chosen, we have $|A_i| \leq |A'_{i+1}|$. Furthermore, because the valuation functions are symmetric, A_i consists of the $|A_i|$ smallest cost-share items (by our tie-breaking rule). Similarly, A'_{i+1} consists of the $|A'_{i+1}|$ smallest cost-share items. We conclude that $A_i \subseteq A'_{i+1}$. (Note that here we exploit that if there are several optimal bundles for player $i+1$, then the one of maximum size is chosen.)

Note that at the end of iteration i, player i becomes inactive and withdraws from the items in $A'_{i+1} \setminus A_i$. By trace-monotonicity (Property (P1)), the cost shares of these items do not decrease in iteration $i+1$. Also, the cost shares of all items in A_i remain the same.

Consider now player $i+1$ in iteration $i+1$. By using similar arguments, it follows that the optimal bundle A_{i+1} consists of the $|A_{i+1}|$ lowest cost share items. But note that in iteration i, the optimal bundle A'_{i+1} of player $i+1$ contained all the items of A_i and possibly a few more. Since the items of A_i continue to be the ones of the smallest cost share, the optimal bundle of $i+1$ in iteration $i+1$ must still contain all the items in A_i. (Note that here we again exploit that the optimal bundle of maximum size is chosen if there are ties.) Thus, it must hold that $A_{i+1} \supseteq A_i$. □

3.4 Main Result for IACSM

In order to state our main result of this section, we need to introduce a crucial parameter that determines the budget balance and social cost approximation guarantees of our mechanism.

Definition 3.3. A cost function $c : 2^N \to \mathbb{R}_{\geq 0}$ is *α-average decreasing* if there exists some $\alpha \geq 1$ such that for every $S \subseteq T \subseteq N$, $\alpha \cdot \frac{c(S)}{|S|} \geq \frac{c(T)}{|T|}$.

Note that for every cost function c there exists some $\alpha \geq 1$ such that c is α-average decreasing. However, here, we are particularly interested in α-average decreasing cost functions for which the parameter α is small, as can be seen by Theorem 3.4 below. Average decreasing functions with small values of α arise naturally in the domains of digital goods and public goods. For digital goods, the cost of serving a non-empty set of customers is typically assumed to be constant because there is a cost for producing the good and then it can be shared with no additional cost (hence, the definition is satisfied with $\alpha = 1$). The same is applicable for some public good models. Note also that *symmetric XOS cost functions* (see Definition 2.1) are average decreasing (i.e., $\alpha = 1$).

The following is the main result of this section.

Theorem 3.4. *Suppose the valuation functions are symmetric submodular and the cost functions are α-average decreasing. Then the mechanism IACSM runs in polynomial time, satisfies IR, NPT, WGSP, and is α-budget balanced and $2\alpha^3 H_n$-approximate.*

Symmetric submodular cost functions are average decreasing (i.e., $\alpha = 1$) since they are a subclass of symmetric XOS functions. As a consequence, we obtain the following corollary from Theorem 3.4 (with an additional improvement on the social cost approximation).

Corollary 3.5. *Suppose the valuation functions and the cost functions are symmetric submodular. Then the mechanism IACSM runs in polynomial time, satisfies IR, NPT, WGSP and is budget balanced, and H_n-approximate.*

Proof. Applying Theorem 3.4 with $\alpha = 1$, we get immediately all the claimed properties except for the social cost approximation which is $2H_n$. However, we note that for symmetric submodular cost functions the term 2α in Lemma 3.14 (see Section 3.7) can simply be omitted, because we have $c(S) \leq c(T)$ for any two sets S, T with $|S| \leq |T|$. By exploiting this in the remaining proof of the social cost approximation guarantee, we obtain an H_n-approximation. □

Note that the approximation factor of H_n for symmetric submodular functions is tight: The impossibility result of Dobzinski et al. [8] for a single public good implies that achieving a better approximation ratio is impossible, even in the single-item case ($m = 1$).

Finally, we point out that *α-average decreasing* functions are subadditive when $\alpha = 1$, while this is not necessarily true for $\alpha > 1$.

Proposition 3.6. *Let $c(\cdot)$ be an α-average decreasing cost function where $\alpha = 1$. Then $c(\cdot)$ is subadditive and in addition, not necessarily symmetric, or submodular. In case $c(\cdot)$ is α-average decreasing with $\alpha > 1$, then $c(\cdot)$ is not necessarily subadditive.*

Proof. Let $S, T \subseteq N$ and assume without loss of generality that $\frac{c(S)}{|S|} \leq \frac{c(T)}{|T|}$. Using that the average cost of c is non-increasing, we obtain

$$c(S \cup T) \leq |S \cup T| \cdot \frac{c(S)}{|S|} \leq c(S) + |T| \frac{c(S)}{|S|} \leq c(S) + c(T).$$

Hence, c is subadditive.

We construct now an example of a non-symmetric and non-submodular cost function that is average decreasing. Let $N = \{1, 2, 3\}$ and consider the function c defined as follows: $c(\{1\}) = 5$,

$c(\{2\}) = 7$, $c(\{3\}) = 8$, $c(\{1,2\}) = 10$, $c(\{2,3\}) = c(\{1,3\}) = 9$, and $c(\{1,2,3\}) = 11$. It is easy to verify that c has decreasing average cost, but obviously, it is not a symmetric function; in fact, it is not even submodular because $c(\{1,2,3\}) - c(\{1,3\}) = 11 - 9 = 2 > 1 = 9 - 8 = c(\{2,3\}) - c(\{3\})$.

Finally consider the following example:

$$c(S) = \begin{cases} 0 & \text{if } S = \emptyset \\ 1 & \text{if } |S| = 1, 2 \\ 3 & \text{if } |S| \geq 3. \end{cases}$$

It is easy to see that this function is *2-average decreasing* (just consider sets of cardinality 2 and supersets of cardinality 3 and notice that this case gives the maximum possible α). Regarding non-subadditivity, consider sets $S \subseteq T \subseteq N$ where $|S| = 1, |T| = 2$, and $S \cap T = \emptyset$. We have that $c(S \cup T) = 3 > 2 = 1 + 1 = c(S) + c(T)$. □

The remainder of this section is devoted to the proof of Theorem 3.4. Unless stated otherwise, we assume below that the valuation functions are symmetric submodular, and the cost functions are α-average decreasing.

3.5 Computational Efficiency

We argue that IACSM can be implemented in polynomial time. Clearly, the mechanism terminates after n iterations. In each iteration, the only non-trivial operations are (i) to compute the optimal bundles for all active players (Line 3) and (ii) to update the cost shares of the items (Line 6). All other operations can be implemented to run in time $O(n+m)$. We prove below that (i) and (ii) can be done in polynomial time.

We first show that optimal bundles can be computed efficiently if the valuation functions are symmetric submodular.

Lemma 3.7. *If the valuation functions are symmetric submodular, then an optimal bundle for player i can be computed in polynomial time.*

Proof. Fix an arbitrary iteration and let $i \in X$ be an active player. We need to show that we can efficiently compute an optimal bundle as defined in Equation (3). Recall that the value that player i derives from a bundle of items only depends on its cardinality (because the valuation functions are symmetric submodular). Thus, to build an optimal bundle, we can start with the empty set and iteratively add an item of the lowest price as long as this price is at most the added marginal value. As a result, the value of the constructed bundle (weakly) increases whenever we add a new item. At the same time, the marginal value of the added items (weakly) decreases because of submodularity. Thus, the first time we consider an item whose marginal value is (strictly) less than its price, we have to stop as the utility can only decrease if we add any of the remaining items. Note that the optimal bundle constructed in this way satisfies the tie-breaking rule described above. Clearly, this procedure stops after at most m iterations. □

We note that for non-symmetric valuation functions, the complexity of IACSM depends on the time needed to compute an optimal bundle. For example, if one has access to demand queries, we can still have an efficient implementation with polynomially many queries.

We next turn to the computation of the cost shares. Note that in general, it is not clear whether the cost shares as defined in Equation (7) can be computed efficiently (as there might be exponentially many supersets that need to be considered). However, as we show below, the cost shares of the items which are constructed throughout the execution of our mechanism IACSM can be computed efficiently.

Lemma 3.8. *The cost shares as defined in Equation (7) can be computed efficiently throughout the execution of mechanism IACSM.*

Proof. Let $\tau = (1, 2, \ldots, n)$ be the player order induced by IACSM. Fix an arbitrary item $j \in M$. After the initialization, we have $T_j = N$ and the cost share of j is thus $\chi_j(R_j^0) = c_j(N)/|N|$. Clearly, as long as the set of players T_j who are tentatively assigned to item j does not change, the cost share of j remains the same. Suppose that T_j changes at the end of iteration i because player i withdraws from item j, resulting in a new set $T_j' = T_j \setminus \{i\}$. The new cost-share $\chi_j(T_j')$ of item j can then be determined simply by taking the maximum of the current cost-share $\chi_j(T_j)$ and $c_j(T_j')/|T_j'|$. Note that this update ensures that the cost-share definition in Equation (7) is met. □

3.6 IR, NPT, and WGSP

The individual rationality and the no positive transfers properties follow directly from the definition of the mechanism. The WGSP property is established by the following lemma.

Lemma 3.9. *Mechanism IACSM is weakly group-strategyproof.*

Proof. Fix a coalition $Q \subseteq N$. Let I be the instance in which all players in Q report their valuations truthfully and let $\hat{I}$ be an instance in which all players in Q misreport their valuations. We need to prove that not every player $i \in Q$ strictly improves her utility by misreporting.

Consider the runs of IACSM on instances I and $\hat{I}$, respectively. Let τ and $\hat{\tau}$ be the player orders induced by IACSM for I and $\hat{I}$, respectively. We assume without loss of generality that in the run on I, player i is considered in iteration i, i.e., $\tau = (1, \ldots, n)$.[10] Let $\hat{\tau} = (\hat{\tau}_1, \ldots, \hat{\tau}_n)$ be the order in which the players are considered in the run on $\hat{I}$. Let i be the first iteration in which either (a) the considered players i and $\hat{\tau}_i$ differ, i.e., $i \neq \hat{\tau}_i$, or (b) the same player $i = \hat{\tau}_i$ is considered, but the bundles allocated in I and $\hat{I}$ differ. Note that such an i must exist as otherwise both runs return the same allocation and we are done. By the choice of i, at the beginning of iteration i the cost share of each item is the same in I and $\hat{I}$, and can only increase subsequently. We distinguish two cases:

Case 1: $i \in Q$. The bundle A_i allocated to player i in I is chosen such that

$$v_i(A_i) - p_i(A_i) \geq v_i(S) - p_i(S) \qquad \forall S \subseteq M, \tag{8}$$

where $p_i(S)$ is the sum of the cost shares of the items in S in iteration i in I (which are the same as in $\hat{I}$). Suppose player i is considered in iteration k in the run on $\hat{I}$, i.e., $\hat{\tau}_k = i$. By the choice of i, we have $k \geq i$. Thus, in iteration k in $\hat{I}$ the price $\hat{p}_i(S)$ of i for each bundle $S \subseteq M$ must satisfy $\hat{p}_i(S) \geq p_i(S)$. In particular, for the bundle $\hat{A}_i$ allocated to player i in iteration k in $\hat{I}$ we have

$$v_i(\hat{A}_i) - \hat{p}_i(\hat{A}_i) \leq v_i(\hat{A}_i) - p_i(\hat{A}_i) \leq v_i(A_i) - p_i(A_i),$$

where the last inequality follows from Equation (8). The claim follows because i is part of the deviating coalition Q.

Case 2: $i \notin Q$. We first argue that player $k = \hat{\tau}_i \in Q$. Assume that $k \notin Q$. Then both i and k bid truthfully in iteration i in I and $\hat{I}$. Because in iteration i the cost shares are the same in both runs, the bundles chosen by i in I and $\hat{I}$ are the same. The same holds for player k. If $i \neq k$ then this is a contradiction to the assumption that IACSM uses a consistent tie-breaking rule (as i is chosen in I but k in $\hat{I}$). If $i = k$ then this is a contradiction to our choice of iteration i (as i chooses the same bundle in I and $\hat{I}$). We conclude that $k \in Q$.

[10] Note that the mechanism might terminate before iteration n, but for the analysis it will be convenient to assume that it uses exactly n iterations. Conceptually, simply assume that the players who become inactive all at once in the final iteration are removed one-by-one (using an arbitrary but consistent tie breaking rule).

We now compare the utility obtained by player $k = \hat{\tau}_i \in Q$ in I and $\hat{I}$. Note that $k \neq i$ because $k \in Q$ and $i \notin Q$. Observe that in iteration i in instance $\hat{I}$, player i reports truthfully and thus opts for the same bundle A_i as in iteration i in I. Given that player k is chosen in iteration i in $\hat{I}$ (and not player i), the bundle $\hat{A}_k$ allocated to k satisfies $\hat{A}_k \subseteq A_i$.

Now, consider the run on I and let $k > i$ be the iteration in which player $k = \hat{\tau}_i$ is considered. By Property (P2), we have $A_i \subseteq A_k$. We conclude that $\hat{A}_k \subseteq A_k$. Because k reports truthfully in I, the choice of A_k implies that

$$v_k(A_k) - p_k(A_k) \geq v_k(S) - p_k(S) \qquad \forall S \subseteq M.$$

In particular, for the bundle $\hat{A}_k$ this implies that

$$v_k(A_k) - v_k(\hat{A}_k) - p_k(A_k \setminus \hat{A}_k) \geq 0. \tag{9}$$

Furthermore, note that the cost shares of all items in A_i and $\hat{A}_k$ remain the same as in iteration i in the runs on I and $\hat{I}$, respectively. Exploiting that $\hat{A}_k \subseteq A_i$ and that the cost shares in iteration i are the same in both runs, we conclude that $p_k(\hat{A}_k) = \hat{p}_k(\hat{A}_k)$. We obtain

$$\begin{aligned} v_k(A_k) - p_k(A_k) &= v_k(\hat{A}_k) - p_k(\hat{A}_k) + [v_k(A_k) - v_k(\hat{A}_k) - p_k(A_k \setminus \hat{A}_k)] \\ &\geq v_k(\hat{A}_k) - p_k(\hat{A}_k) = v_k(\hat{A}_k) - \hat{p}_k(\hat{A}_k), \end{aligned}$$

where the first equality holds because p_k is additive and the inequality follows from Equation (9). The claim now follows because of $k \in Q$. □

3.7 Budget Balance and Social Cost Approximation

We start by providing the budget balance performance of the mechanism.

Lemma 3.10. *Mechanism IACSM is α-budget balanced.*

Proof. Let the player order induced by IACSM be $\tau = (1, \dots, n)$. Fix an arbitrary item j and let τ_j be the trace of item j. Let T_j be the final set of players allocated to item j and recall that $R_j^k = T_j$ for $k = |\tau_j|$. By the definition of the cost-sharing function χ_j in Equation (7), there is some set $S = R_j^\ell$ with $\ell \in \{0, \dots, k\}$ such that the cost share of item j is

$$\chi_j(T_j) = \frac{c_j\left(R_j^\ell\right)}{n-\ell} \geq \frac{c_j\left(R_j^k\right)}{n-k} = \frac{c_j(T_j)}{|T_j|}. \tag{10}$$

Summing over all players in T_j, we obtain

$$c_j(T_j) \leq \sum_{i \in T_j} \chi_j(T_j) = |T_j| \cdot \frac{c_j(S)}{|S|} \leq |T_j| \cdot \alpha \frac{c_j(T_j)}{|T_j|} = \alpha \cdot c_j(T_j), \tag{11}$$

where the second inequality holds because c_j is α-average decreasing and $T_j \subseteq S$.

Finally, summing inequality Equation (11) over all items $j \in M$ we obtain

$$\sum_{j \in M} c_j(T_j) \leq \sum_{j \in M} \sum_{i \in T_j} \chi_j(T_j) \leq \alpha \sum_{j \in M} c_j(T_j), \tag{12}$$

which proves the claim. □

We now show that our mechanism IACSM is $2\alpha^3 H_n$-approximate with respect to the social cost objective for symmetric submodular valuation functions.

Let $A = (A_i)_{i \in N}$ be the allocation computed by the mechanism, where $A_i \subseteq M$ is the subset of items that player i receives. As before, without loss of generality, we assume that the player order induced by IACSM is $\tau = (1, \dots, n)$. Recall that for every item $j \in M$, $T_j = \{i \in N : j \in A_i\}$ is the final set of players that receive item j. We also use T_j^i to refer to the subset of players who are allocated to item j at the beginning of iteration i. Clearly, $T_j^i \supseteq T_j$ for every player i and item j.

We first prove some lemmas, which will be helpful later on.

Lemma 3.11. *Fix an item $j \in M$ and let i be the first player in τ such that $j \in A_i$. Then $T_j = \{i, \dots, n\}$.*

Proof. By the choice of i, we have that $j \in A_i$ and $j \notin A_k$ for every player $k < i$. From Property (P2), it follows that for every k with $i \le k < n$, $A_k \subseteq A_{k+1}$. Thus, $j \in A_k$ for every $i \le k \le n$, which concludes the proof. □

Lemma 3.12. *Consider player i who becomes inactive in iteration i. We have*

$$v_i(A_i) - \sum_{j \in A_i} \chi_j(T_j) \ge v_i(S) - \sum_{j \in S} \chi_j(T_j) \quad \forall S \subseteq M.$$

Proof. In iteration i, the final bundle A_i is chosen as the set of items maximizing the utility of player i with respect to the current cost shares, i.e.,

$$v_i(A_i) - \sum_{j \in A_i} \chi_j\left(T_j^i\right) \ge v_i(S) - \sum_{j \in S} \chi_j\left(T_j^i\right) \quad \forall S \subseteq M. \tag{13}$$

Recall that T_j^i is the set of players that are allocated to item j in iteration i. Note that by Lemma 3.11, $T_j^i = T_j$ for every $j \in A_i$. Furthermore, $T_j^i \supseteq T_j$ for every $j \in M \setminus A_i$ as additional players might withdraw from j in subsequent iterations. Note that the final set T_j is reached from T_j^i by following the trace τ_j of item j. The claim now follows from the trace-monotonicity of χ_j (Property (P1)). □

Lemma 3.13. *Consider player i who becomes inactive in iteration i. For every item $j \in M$,*

$$\chi_j\left(T_j^i\right) \le \alpha \frac{c_j(\{i, \dots, n\})}{n - i + 1}.$$

Proof. Fix some $j \in M$. In iteration i, we have $T_j^i \supseteq \{i, \dots, n\}$. Let k be such that $T_j^i = R_j^k$ and recall that $R_j^\ell \supseteq R_j^k$ for every $\ell \in \{0, \dots, k\}$. We obtain

$$\chi_j\left(T_j^i\right) = \max_{\ell \in \{0, \dots, k\}} \frac{c_j\left(R_j^\ell\right)}{n - \ell} \le \alpha \frac{c_j(\{i, \dots, n\})}{n - i + 1}, \tag{14}$$

where the inequality holds because c_j is α-average decreasing. □

Lemma 3.14. *Let c be an α-average decreasing cost function. Let $S, T \subseteq N$ be arbitrary subsets with $|S| \le |T|$. Then $c(S) \le 2\alpha c(T)$.*

Proof. Assume for the sake of a contradiction that $c(S) > 2\alpha c(T)$. Consider the set $S \cup T$ and note that $|S \cup T| \le 2|T|$. Because c is non-decreasing, we have $c(S \cup T) \ge c(S) > \frac{2}{\alpha} c(T)$. Using that c is α-average decreasing, we obtain $\frac{c(S \cup T)}{|S \cup T|} \le \alpha \frac{c(T)}{|T|}$, which implies that $c(S \cup T) \le \alpha \frac{|S \cup T|}{|T|} c(T) \le 2\alpha c(T)$, a contradiction. □

We are now ready to prove the approximation guarantee.

Lemma 3.15. *Mechanisms IACSM is $2\alpha^3 H_n$-approximate.*

Proof. Let $A^* = (A_1^*, \dots, A_n^*)$ be an optimal allocation and let T_j^* be the respective set of players that receive item j in A^*. We have

$$\begin{aligned}\pi(A) &= \sum_{i \in N} \Big(v_i(M) - v_i(A_i)\Big) + \sum_{j \in M} c_j(T_j) \\ &\leq \sum_{i \in N} v_i(M) - \sum_{i \in N} \Bigg(v_i(A_i) - \sum_{j \in A_i} \chi_j(T_j)\Bigg) \\ &\leq \sum_{i \in N} v_i(M) - \sum_{i \in N} \Bigg(v_i(A_i^*) - \sum_{j \in A_i^*} \chi_j\left(T_j^i\right)\Bigg) \\ &= \sum_{i \in N} \Big(v_i(M) - v_i\left(A_i^*\right)\Big) + \sum_{i \in N} \sum_{j \in A_i^*} \chi_j\left(T_j^i\right),\end{aligned}$$

where the first inequality holds because χ_j is α-budget balanced and the second inequality follows from Equation (13) in the proof of Lemma 3.12.

The proof follows if we can show that

$$\sum_{i \in N} \sum_{j \in A_i^*} \chi_j\left(T_j^i\right) \leq 2\alpha^3 H_n \sum_{j \in M} c_j\left(T_j^*\right). \tag{15}$$

We use a charging argument to prove Equation (15). Fix some item $j \in M$ and order the players in T_j^* according to the player order $\tau = (1, \dots, n)$ induced by IACSM; let $T_j^* = \{i_1, \dots, i_{k_j^*}\}$ be the ordered set with $k_j^* := |T_j^*|$. We now “tag” each player i in T_j^* with a fraction of the cost $c_j(T_j^*)$ for item j as follows: for the lth player $i = i_l$ in T_j^* with $1 \leq l \leq k_j^*$, define

$$\text{tag}_i(j) := \frac{c_j\left(T_j^*\right)}{k_j^* - l + 1}. \tag{16}$$

That is, the first player i_1 in T_j^* is tagged with $c_j(T_j^*)/k_j^*$, the second player i_2 with $c_j(T_j^*)/(k_j^* - 1)$ and so forth, and the last player $i_{k_j^*}$ is tagged with $c_j(T_j^*)$.

We first derive two lower bounds on the tagged cost:

Claim 3.16. *For every player $i \in N$ and for every item $j \in A_i^*$:*

$$\text{tag}_i(j) \geq \frac{c_j\left(T_j^*\right)}{n - i + 1} \qquad \text{and} \qquad \text{tag}_i(j) \geq \frac{c_j\left(T_j^*\right)}{\left|T_j^*\right|}.$$

Proof. The latter bound holds by definition Equation (16). To see that the former bound holds, observe that the kth last player ($1 \leq k \leq k_j^*$) in the ordered set T_j^* is tagged by $c_j(T_j^*)/k$. The claim now follows because there are at most $n - i$ players succeeding i in T_j^* according to the order. □

Note that the total tagged cost of item j satisfies

$$\sum_{i \in T_j^*} \text{tag}_i(j) = \sum_{l=1}^{k_j^*} \frac{c_j\left(T_j^*\right)}{k_j^* - l + 1} \leq H_n c_j\left(T_j^*\right). \tag{17}$$

Thus, to prove Equation (15) it suffices to show that the total cost-share sum is upper bounded by the total tagged cost, i.e.,

$$\sum_{i \in N} \sum_{j \in A_i^*} \chi_j\left(T_j^i\right) \leq 2\alpha^3 \sum_{j \in M} \sum_{i \in T_j^*} \text{tag}_i(j). \tag{18}$$

ALGORITHM 2: Sequential Mechanism (SM)

```
Input: Declared valuation functions $(b_i)_{i \in N}$.
Output: Allocation $A = (A_i)_{i \in N}$ and payments $p = (p_i)_{i \in N}$.
Initialization: Fix an order on the set of players $N = \{1, \dots, n\}$.
for $i = 1, \dots, n$ do
    Compute an optimal bundle $A_i$ for player $i$:
        $A_i \in \arg\max_{S \subseteq M} \{b_i(S) - p_i(S)\}$,    where
            $p_i(S) = C(A_1, \dots, A_{i-1}, S, \emptyset, \dots, \emptyset) - C(A_1, \dots, A_{i-1}, \emptyset, \dots, \emptyset)$.
    (If there are multiple optimal bundles, choose the lexicographically smallest one.)
return $A = (A_i)_{i \in N}$ and $p = (p_i)_{i \in N}$, where $p_i = p_i(A_i)$.
```

We show that for every i and every $j \in A_i^*$, $\chi_j(T_j^i) \leq \text{tag}_i(j)$. Summing over all $i \in N$ and $j \in A_i^*$ then proves Equation (15).

We distinguish two cases:

Case 1: $|T_j^*| \geq n - i + 1$: Let $S \subseteq T_j^*$ be a set such that $|S| = n - i + 1$. We have

$$\chi_j\left(T_j^i\right) \leq \alpha \frac{c_j(\{i, \dots, n\})}{n - i + 1} \leq 2\alpha^2 \frac{c_j(S)}{|S|} \leq 2\alpha^2 \frac{c_j\left(T_j^*\right)}{n - i + 1} \leq 2\alpha^2 \text{tag}_i(j), \tag{19}$$

where the first inequality follows from Lemma 3.13, the second inequality follows from Lemma 3.14, the third inequality holds because c_j is non-decreasing and the last inequality follows from Claim 3.16.

Case 2: $|T_j^*| < n - i + 1$: Let $S \supset T_j^*$ be a set such that $|S| = n - i + 1$. We have

$$\chi_j\left(T_j^i\right) \leq \alpha \frac{c_j(\{i, \dots, n\})}{n - i + 1} \leq 2\alpha^2 \frac{c_j(S)}{|S|} \leq 2\alpha^3 \frac{c_j\left(T_j^*\right)}{\left|T_j^*\right|} \leq 2\alpha^3 \text{tag}_i(j), \tag{20}$$

where the first inequality follows from Lemma 3.13, the second inequality follows from Lemma 3.14, the third inequality holds because c_j is α-average decreasing and the last inequality follows from Claim 3.16. This concludes the proof. □

4 MECHANISMS FOR GENERAL VALUATIONS AND SUBADDITIVE COST FUNCTIONS

In this section, we move away from symmetric submodular valuation functions and derive results for more general functions. In particular, we investigate the performance of the *Sequential Mechanism* [20] for general valuations and subadditive cost functions. Although for arbitrary subadditive cost functions this mechanism does not provide favorable approximation guarantees, we identify conditions on the cost functions under which it achieves significantly better approximation factors. This is based on considering a different parameterization of cost functions with regard to their average cost shares.

4.1 The Sequential Mechanism

The ***Sequential Mechanism (SM)*** was introduced by Moulin [20] and was also studied in [9]. A description of the mechanism SM is given in Algorithm 2. We note that this mechanism is applicable both to separable and non-separable cost functions. Here, we first focus on separable cost functions. In Section 4.3, we consider generalizations to the non-separable setting.

It is trivial to see that SM is budget-balanced, and it is also known that it is WGSP [9]. However, for arbitrary monotone subadditive cost functions, the mechanism achieves a (poor) social cost approximation guarantee of n only (see [9]).

Theorem 4.1 ([9]). *Suppose we have general valuation functions and non-decreasing subadditive cost functions. Then, the Sequential Mechanism satisfies IR, NPT, WGSP and is budget balanced, and n-approximate.*

Despite this, we show that SM has better guarantees under certain conditions. Namely, we identify a crucial parameter of each cost function c_j with $j \in M$ that allows us to quantify this improvement. The parameterization introduced here is different from the one used in Section 3 and it compares the average cost of a set $T \subseteq N$ with the minimum standalone cost of a player in T. More specifically, we define the following property:

Definition 4.2. A cost function $c : 2^N \to \mathbb{R}_{\geq 0}$ is *α-average min-bounded*, if there exists some $\alpha \geq 1$ such that for every set $T \subseteq N$, we have $\alpha \cdot \frac{c(T)}{|T|} \geq c_{\min}$, where $c_{\min} = \min_{j \in T} c(\{j\})$.

Definition 4.2 may look somewhat contrived at first glance and we thus provide some more intuition on how we arrived at this parameterization. Given that IACSM performs well for α-average decreasing functions and small values of α (see Section 3), it is natural to focus on the complement of this class. For example, fix $\alpha = 1$ for now. Then the exact complement is not easy to characterize because it involves two existential quantifiers. We therefore consider a subset of this complement (with only one existential quantifier) by demanding that for every T, there exists $S \subseteq T$ such that $c(S)/|S| < c(T)/|T|$. It is not hard to verify that this definition is equivalent to the class of 1-average min-bounded functions. For larger values of α, we can again see that α-average-min-bounded functions capture a chunk of the complement of α-average decreasing functions. Thus, a positive result for α-average-min-bounded functions narrows down on the cost functions that are not yet known to admit good approximation guarantees.

Note that for every cost function we can find an $\alpha \geq 1$ such that it is α-average min-bounded. As the next theorem reveals, the Sequential Mechanism attains a favorable performance for small values of α.

Theorem 4.3. *Suppose we have general valuation functions and for each item $j \in M$, the cost function $c_j : 2^N \to \mathbb{R}_{\geq 0}$ is non-decreasing, subadditive, and α-average min-bounded for some $\alpha \geq 1$. Then, the Sequential Mechanism satisfies IR, NPT, WGSP and is budget balanced, and $\alpha \cdot H_n$-approximate.*

For the proof of Theorem 4.3, we use the following proposition:

Proposition 4.4. *If $c : 2^N \to \mathbb{R}_{\geq 0}$ is non-decreasing and α-average min-bounded, then $\sum_{i \in T} c(\{i\}) \leq \alpha H_{|T|} \cdot c(T)$ for every $T \subseteq N$.*

Proof. Fix $T \subseteq N$ and rename the players of T so that they are in decreasing order with respect to the standalone cost, i.e., for any $i, j \in T$ with $i < j$, it holds $c(\{i\}) \geq c(\{j\})$. For convenience, we may assume that $T = \{1, 2, \ldots, |T|\}$. Fix some $i \in T$ and consider the set $\{1, 2, \ldots, i\} \subseteq T$. Note that in this set player i has the minimum standalone cost $c(\{i\})$. Hence, by Definition 4.2, the cost function satisfies:

$$c(\{i\}) \leq \alpha \cdot \frac{c(\{1, \ldots, i\})}{i} \leq \alpha \cdot \frac{c(T)}{i},$$

where the last inequality holds because c is non-decreasing. The above inequality holds for any $i \in T$. Summing overall players $i \in T$ proves the claim. □

Proof of Theorem 4.3. We only need to prove that SM is αH_n-approximate. All the other properties have been established in [9, 20]. Let $A = (A_i)_{i \in N}$ be the allocation output by the mechanism and let $A^* = (A_i^*)_{i \in N}$ be an optimal allocation. Furthermore, let T_j^* be the respective set of players that receive item j in A^*. To simplify notation in the analysis, we also let $A_{<i}$ denote the tuple $(A_1, \dots, A_{i-1}, \emptyset, \dots, \emptyset)$. Define now the incremental cost of a player i for a bundle $S \subseteq M$, with respect to the allocation constructed by the Sequential Mechanism before i's turn as $\Delta_i(A_{<i}, S) = C(A_1, \dots, A_{i-1}, S, \emptyset, \dots, \emptyset) - C(A_1, \dots, A_{i-1}, \emptyset, \dots, \emptyset)$.

We have

$$\begin{aligned}\pi(A) &= \sum_{i \in N} \Big[v_i(M) - v_i(A_i) \Big] + C(A) \\ &= \sum_{i \in N} v_i(M) - \sum_{i \in N} \Big[v_i(A_i) - \Delta_i(A_{<i}, A_i) \Big] \\ &\leq \sum_{i \in N} v_i(M) - \sum_{i \in N} \Big[v_i(A_i^*) - \Delta_i(A_{<i}, A_i^*) \Big] \\ &= \sum_{i \in N} \Big[v_i(M) - v_i(A_i^*) \Big] + \sum_{i \in N} \Delta_i(A_{<i}, A_i^*).\end{aligned}$$

Note that the inequality holds because A_i was chosen as the optimal bundle for i. The next step is to prove a bound on the incremental costs in the form

$$\sum_{i \in N} \Delta_i(A_{<i}, A_i^*) \leq \beta \cdot C(A^*). \tag{21}$$

The proof follows if we can show that Equation (21) holds for $\beta = \alpha H_n$ because we then have

$$\pi(A) \leq \sum_{i \in N} \Big[v_i(M) - v_i(A_i^*) \Big] + \alpha \cdot H_n C(A^*) \leq \alpha H_n \cdot \pi(A^*).$$

By exploiting the subadditivity of the cost functions c_j, we obtain

$$\Delta_i(A_{<i}, A_i^*) = C(A_{<i}, A_i^*) - C(A_{<i}) \leq C(A_{<i}) + C(A_i^*, \emptyset_{-i}) - C(A_{<i}) = \sum_{j \in A_i^*} c_j(\{i\}).$$

Summing over all $i \in N$, and using Proposition 4.4, we get:

$$\sum_{i \in N} \Delta_i(A_{<i}, A_i^*) \leq \sum_{i \in N} \sum_{j \in A_i^*} c_j(\{i\}) = \sum_{j \in M} \sum_{i \in T_j^*} c_j(\{i\}) \leq \sum_{j \in M} \alpha H_{|T_j^*|} c_j(T_j^*) \leq \alpha H_n C(A^*).$$

□

By going through the proof of Theorem 4.3 more carefully, we realize the following:

Remark 4.5. For any subclass of non-decreasing, subadditive cost functions, it suffices to establish inequality Equation (21) to prove that the Sequential Mechanism has a social cost approximation guarantee of β.

We next prove that for $\alpha = 1$ the approximation factor is tight.

PROPOSITION 4.6. *Even for the single item setting, there exists a* 1*-average min-bounded cost function, under which the Sequential Mechanism provides an* H_n*-approximation.*

PROOF. Consider a set $N = \{p_1, p_2, \dots, p_n\}$ of players and the following function $c : 2^n \to \mathbb{R}_{\geq 0}$:

$$c(S) = \begin{cases} 0 & \text{if } S = \emptyset \\ \frac{k}{j} & \text{if } S = \{p_j\} \\ \min\{k, \sum_{p_j \in S} c(\{p_j\})\} & \text{if } |S| \geq 2, \end{cases} \tag{22}$$

where $k \geq 0$ and $S \subseteq N$. We have to show that the above function is subadditive, non-decreasing, it has bounded average costs, and that the *Sequential Mechanism* is H_n-approximate.

Subadditivity: Initially notice that if $p_1 \in S \subseteq N$, then $c(S) = k$. Now consider the non-empty sets $A, B \subseteq N$.

- If $p_1 \in A \cup B$, then p_1 is in at least one of A, B. Thus we have that $c(A \cup B) \leq c(A) + c(B)$.
- If $p_1 \notin A \cup B$, then

$$\begin{aligned} c(A \cup B) &= \min\left\{k, \sum_{p_j \in A \cup B} c(\{p_j\})\right\} \\ &\leq \min\left\{k, \sum_{p_j \in A} c(\{p_j\}) + \sum_{p_j \in B} c(\{p_j\})\right\} \\ &\leq \min\left\{k, \sum_{p_j \in A} c(\{p_j\})\right\} + \min\left\{k, \sum_{p_j \in B} c(\{p_j\})\right\} \\ &= c(A) + c(B). \end{aligned}$$

Non-decreasingness: Consider the sets $A \subseteq B \subseteq N$.

- If $p_1 \in A$, then $p_1 \in B$ and thus $c(A) = k = c(B)$.
- If $p_1 \notin A$ and $p_1 \in B$, then $c(A) = \min\{k, \sum_{p_j \in A} c(\{p_j\})\} \leq k = c(B)$.
- If $p_1 \notin A$ and $p_1 \notin B$, then $c(A) = \min\{k, \sum_{p_j \in A} c(\{p_j\})\} \leq \min\{k, \sum_{p_j \in B} c(\{p_j\})\} = c(B)$.

Min-bounded average costs: Consider the non-empty set of indices $A \subseteq [n]$ and the set $B = \{p_1, p_2, \ldots, p_{|A|}\}$.

- If $p_1 \in A$, then $c(A) = k$. Now notice that $\frac{c(A)}{|A|} = \frac{k}{|A|} = \min_{j \in B} c(\{i_j\}) \geq \min_{j \in A} c(\{i_j\})$.
- If $p_1 \notin A$, then we have that either $c(A) = k$ and thus $\frac{c(A)}{|A|} \geq \min_{j \in A} c(\{i_j\})$ as before, or $c(A) = \sum_{p_j \in A} c(\{p_j\})$ and thus

$$\frac{c(A)}{|A|} = \frac{\sum_{p_j \in A} c(\{p_j\})}{|A|} \geq \min_{j \in A} c(\{i_j\}).$$

Approximation of the Sequential Mechanism: Consider an instance where the value each player p_j has for the item is $v_{p_j} = \frac{k}{j} - \epsilon$, for arbitrary small $\epsilon > 0$. Suppose now that A^* is the optimal allocation, A' is the allocation where every player is served, and A is the output of the Sequential Mechanism, in which it is easy to see that no player is served. We have that $\pi(A^*) \leq \pi(A') = k$, while

$$\pi(A) = 0 + \sum_{p_j \in N} \frac{k}{j} - n\epsilon = kH_n - n\epsilon.$$

So since ϵ is arbitrary small, the approximation cannot be better than H_n. □

4.2 Improved Approximation Guarantees and Applications

We continue with a natural refinement of Definition 4.2, which turns out to provide even better approximation factors of the Sequential Mechanism.

Definition 4.7. A cost function $c : 2^N \rightarrow \mathbb{R}_{\geq 0}$ is *α-average max-bounded*, if there exists some $\alpha \geq 1$ such that for every set $T \subseteq N$, we have $\alpha \cdot \frac{c(T)}{|T|} \geq c_{\max}$, where $c_{\max} = \max_{j \in T} c(\{j\})$.

Clearly, any function that is α-average max-bounded is also α-average min-bounded. Thus, we already have an αH_n-approximation for non-decreasing, subadditive and α-average max-bounded cost functions. Below we show that we can achieve a much better guarantee.

Theorem 4.8. *Suppose we have general valuation functions and for each item $j \in M$, the cost function $c_j : 2^N \to \mathbb{R}_{\geq 0}$ is non-decreasing, subadditive, and α-average max-bounded for some $\alpha \geq 1$. Then the Sequential Mechanism satisfies IR, NPT, WGSP,and is budget-balanced, and α-approximate.*

Proof. We only need to prove that the mechanism is α-approximate. By following exactly the same reasoning as in the proof of Theorem 4.3 and using the observation made in Remark 4.5 (note that c_j is subadditive by assumption), we only need to prove that Equation (21) holds for $\beta = \alpha$. Exploiting the properties of the cost functions, we obtain

$$\sum_{i \in N} \Delta_i(A_{<i}, A_i^*) \leq \sum_{i \in N} \sum_{j \in A_i^*} c_j(\{i\}) \leq \sum_{j \in M} \sum_{i \in T_j^*} \alpha \frac{c_j(T_j^*)}{|T_j^*|} = \alpha \sum_{j \in M} c_j(T_j^*) \leq \alpha C(A^*).$$

□

Example applications of combinatorial cost functions. We give some examples of combinatorial cost functions below and show that they are α-average max-bounded (possibly depending on some parameters of the combinatorial problem). In particular, by applying Theorem 4.8 we obtain attractive social cost approximation guarantees for these problems. For simplicity, all examples consider a single item only; but clearly, we can consider more general multiple item settings (e.g., when for each item $j \in M$, c_j captures one of the problems below).

(1) **Set Cover.** We are given a universe of elements U and a family $\mathcal{F} \subseteq 2^U$ of subsets of U. The players correspond to the elements of U and the cost $c(S)$ for serving a set of players $S \subseteq U$ is defined as the size of a minimum cardinality set cover for S.
(2) **Vertex Cover.** This is a special case of Set Cover. We are given an undirected and unweighted graph $G = (V, E)$, and the players are the edges of the graph. The cost $c(S)$ for serving a set $S \subseteq E$ of players is defined as the size of a minimum vertex cover in the subgraph induced by S.
(3) **Matching.** We are given an undirected and unweighted graph $G = (V, E)$ and the players correspond to the edges. The cost $c(S)$ for serving a set S of players is defined as the size of a maximum cardinality matching in the subgraph induced by S.

Using our α-average max-bounded notion, it is now easy to prove that these problems admit constant social cost approximation guarantees (under certain restrictions).

Theorem 4.9. *The Sequential Mechanism is α-approximate for the above problems, where*

(1) *$\alpha = d$ for the Set Cover problem, where d is the maximum cardinality of the sets in $\mathcal{F}$;*
(2) *$\alpha = k$ for the Vertex Cover problem in graphs of maximum degree k;*
(3) *$\alpha = k$ for the Matching problem in bipartite graphs of maximum degree k;*
(4) *$\alpha = (5k + 3)/4$ for the Matching problem in general graphs of maximum degree k.*

Proof. We have the following:

(1) For Set Cover, it is trivial to check that the cost function is subadditive. Consider a subset T of the players. Note that for a single player $i \in T$, $c(\{i\}) = 1$, since each element can be covered by a single subset. Given that the maximum cardinality of a subset is d, the minimum set cover for covering the set of players T is at least $|T|/d$. We conclude that c is d-average max-bounded.
(2) Vertex Cover is a special case of Set Cover. Consider a subset $T \subseteq E$ of the edges. Given that the maximum degree is k, the minimum vertex cover for covering a set of edges T is at least $|T|/k$. We conclude that this cost function is k-average max-bounded.

(3) The cost function in the Maximum Matching problem is also subadditive. In bipartite graphs, we also know that the cardinality of a maximum matching equals the cardinality of a minimum vertex cover. This immediately implies a k-approximation.

(4) For general graphs, note again that $c(\{i\}) = 1$ for a single player $i \in E$. By the work of [15], we know that in a graph of m edges and with degree k, there always exists a matching of size at least $4m/(5k+3)$. Hence for a set of players $T \subseteq E$, we have that $c(T)/|T| \geq 4/(5k+3)$. We conclude that the cost function c is $(5k+3)/4$-average max-bounded. □

We compare these bounds with the existing results in the literature: For Vertex Cover, there is a mechanism that is 2-budget-balanced and $O(\log n)$-approximate [19]. Thus, for graphs with maximum degree less than $\log n$, we obtain a better guarantee. For Set Cover, there is a mechanisms that is $O(\log n)$-budget-balanced and $O(\log n)$-approximate [19]. Hence, we obtain an improvement if the sets in $\mathcal{F}$ have small size. Finally, we note that our results do not apply to the weighted versions of these problems.

4.3 Guarantees of the Sequential Mechanism for Non-Separable Cost Functions

We extend our results to non-separable cost functions. Recall that in this setting, the cost $C(A)$ of an allocation $A = (A_i)_{i \in N}$ is given by some general (not necessarily separable) cost function $C : (2^M)^n \rightarrow \mathbb{R}_{\geq 0}$. In particular, C may encode dependencies among different items.

We introduce some more notation. Given two allocations $S = (S_i)_{i \in N}$ and $T = (T_i)_{i \in N}$, we define $S \cup T$ as the componentwise union of S and T, i.e., $S \cup T = (S_1 \cup T_1, \ldots, S_n \cup T_n)$. Similarly, we write $S \subseteq T$ if this relation holds componentwise, i.e., $S_i \subseteq T_i$ for every $i \in N$. Given an allocation $A = (A_i)_{i \in N}$ and a set of players $S \subseteq N$, we define $A|_S = (A_S, \emptyset_{-S})$ as the allocation in which each player $i \in S$ receives the items in A_i, and all other players receive nothing. If $S = \{i\}$ is a singleton set, we also write $A|_i$ instead of $A|_{\{i\}}$. Throughout this section, we remain in the domain of non-decreasing and subadditive cost functions. In the non-separable case, a cost function $C : (2^M)^n \rightarrow \mathbb{R}_{\geq 0}$ is non-decreasing if $C(S) \leq C(T)$ for every pair of allocations S, T, with $S \subseteq T$. Also, it is subadditive if for every two allocations $S = (S_i)_{i \in N}$ and $T = (T_i)_{i \in N}$, we have $C(S \cup T) \leq C(S) + C(T)$.

We now adapt Definitions 4.2 and 4.7 to non-separable cost functions.

Definition 4.10. Let $C : (2^M)^n \rightarrow \mathbb{R}_{\geq 0}$ be a non-separable cost function.

- C is *α-average min-bounded*, if there exists some $\alpha \geq 1$ such that for every allocation A and every subset $T \subseteq N$ with $|T| \geq 2$, it holds $\alpha \frac{C(A|_T)}{|T|} \geq C_{\min}$, where $C_{\min} = \min_{j \in T} C(A|_j)$.
- C is *α-average max-bounded*, if there exists some $\alpha \geq 1$ such that for every allocation A and every subset $T \subseteq N$ with $|T| \geq 2$, it holds $\alpha \frac{C(A|_T)}{|T|} \geq C_{\max}$, where $C_{\max} = \max_{j \in T} C(A|_j)$.

As before, if a non-separable function is α-average max-bounded, then it is also α-average min-bounded.

We remark that it has been shown in [9, 20] that the Sequential Mechanism is weakly group-strategyproof and budget balanced for the non-separable setting. We proceed by adapting Proposition 4.4 for the non-separable setting.

Proposition 4.11. *If $C : (2^M)^n \rightarrow \mathbb{R}_{\geq 0}$ is a non-separable cost function, which is non-decreasing and α-average min-bounded, then for every allocation A, $\sum_{i \in T} C(A|_i) \leq H_{|T|} \cdot C(A|_T)$ for every $T \subseteq N$.*

Proof. Let A be an allocation and fix $T \subseteq N$. Rename the players in T such that for any $i, j \in T$ with $i < j$, it holds that $C(A|_i) \geq C(A|_j)$. For convenience, we may assume that $T = \{1, 2, \ldots, |T|\}$.

Fix some $i \in T$ and consider the set $\{1, 2, \ldots, i\} = [i] \subseteq T$. By exploiting that C is α-average min-bounded and non-decreasing, we have

$$C(A|_i) \leq \alpha \frac{C(A|_{[i]})}{i} \leq \frac{C(A|_T)}{i}.$$

Note that the above inequality holds for any $i \in T$. Summing over all players $i \in T$ proves the claim. □

Now using the same reasoning as in the proof of Theorem 4.3, we obtain the same approximation guarantee of αH_n as in the separable setting. Further, the improvement we obtained in Theorem 4.8 also goes through in this setting. We summarize these observations in the following corollary.

Corollary 4.12. *Suppose we have general valuation functions and a non-decreasing, subadditive, and α-average min-bounded cost function $C : (2^M)^n \to \mathbb{R}_{\geq 0}$. Then, the Sequential Mechanism satisfies IR, NPT, WGSP and is budget balanced, and $\alpha \cdot H_n$-approximate. Furthermore, if C is also α-average max-bounded, then the Sequential Mechanism is α-approximate.*

5 DISCUSSION

In Section 3, we proposed the mechanism IACSM, which is weakly group-strategyproof under general cost functions and symmetric submodular valuations. Moreover, it is α-budget balanced and $2\alpha^3 H_n$-approximate when we restrict the cost functions to the α-average decreasing class. The social cost approximation guarantee further improves to H_n if the cost functions are symmetric submodular and this is the best possible (due to the known lower bound for public-excludable goods [8]).

We would like to point out that in order for our mechanism to provide these guarantees, properties (P1) and (P2) are necessary. In particular, Property (P1) dictates that the price of the items increases through the iterations of the mechanism, something that is crucial for ensuring strategyproofness. Property (P2) is also very important. When a player i is offered her optimal bundle A_i at iteration i, her cost share is computed under the assumption that every player $j > i$, will also get a bundle of items that contains A_i, i.e., $A_i \subseteq A_{i+1}$. Notice that if we cannot guarantee this, the price of the items in A_i may change in the iterations that follow iteration i. As a result, we might have to reconsider player i in the future, as otherwise we lose the budget balance properties of IACSM. The latter, however, may affect the strategyproofness of the mechanism as now a player has more options to strategize. By assuming symmetric submodular valuation functions, we showed that we can guarantee Property (P2). Going beyond this class of valuation functions; however, seems highly non trivial, and it is a very interesting problem, where we expect that entirely new ideas are needed.

It would also be interesting to extend our mechanism to non-separable cost functions. We note that separability of the costs in Section 3 is needed for IACSM only to argue that the cost share per item increases as players withdraw (with respect to the trace). One would need to investigate how to adapt the mechanism and enforce this property in the non-separable setting. Technically, this seems far from obvious and we leave a proper treatment of this issue for future work.

In Section 4, we studied the (partially) complementary class of α-average min-bounded cost functions. We showed that the well-known Sequential Mechanism is budget balanced and αH_n-approximate even for general valuation functions. These results also extend to non-separable cost functions. A very natural question is whether SM is optimal in this setting, and we note that the answer is not yet clear: The impossibility result of [8] holds for the public-excludable good cost function, which is symmetric submodular and thus 1-average decreasing. However, it is not hard to see that this does not fall within the α-average min-bounded class for any constant α. This leads to the question of whether there exists a WGSP mechanism that breaks the $\Omega(\log(n))$-approximation in terms of social cost for α-average min-bounded functions with small values of α.

Finally, what we also find very interesting is to identify the class of cost functions for which neither of the two mechanisms studied here perform well. Recall that, for any constant value of α, if a cost function is either α-average decreasing or α-average min-bounded, then a good performance is guaranteed. Thus, we need look at the complement of the set of α-average decreasing functions and the set of α-average min-bounded functions for small value of α and examine whether these complements have a non-empty intersection. The following proposition shows that this intersection is indeed non-empty.

Proposition 5.1. *Given $\alpha \geq 1$, the intersection of the complements of α-average decreasing and α-average min-bounded functions is non-empty.*

Proof. We begin by defining the complements of the two sets:

- *Complement of α-average decreasing*: For every constant $\alpha \geq 1$ that we choose, there exist sets $S \subseteq T \subseteq N$ such that: $\alpha \frac{C(S)}{|S|} < \frac{C(T)}{|T|}$.
- *Complement of α-average min-bounded*: For every constant $\alpha \geq 1$ that we choose, there exists a set $T \subseteq N$ such that: $\alpha \frac{C(T)}{|T|} < C(\{i\})$, where $i = \arg\min_{j \in T} C(\{j\})$.

We will prove this for the case where there exists only a single item. Consider set $N = \{p_1, p_2, \ldots, p_n\}$ of players and the following function $c : 2^n \rightarrow \mathbb{R}_{\geq 0}$:

$$c(S) = \begin{cases} 0 & \text{if } S = \emptyset \\ \sqrt{j} & \text{if } S = \{p_j\} \\ \max_{j \in S} c(\{p_j\}) & \text{if } |S| \geq 2, \end{cases} \tag{23}$$

where $S \subseteq N$. Before we proceed with the proof, notice that this function is also subadditive and by definition non-decreasing.

- *Complement of α-average decreasing*: Set $S = \{p_2\}$ and $T = \{p_2, p_n\}$. Now notice that $\alpha < \frac{\sqrt{n}}{2} \Rightarrow 2\alpha < \sqrt{n}$, which holds since for any $\alpha \geq 1$ that we choose, we can always find a large enough n for the inequality to be true.
- *Complement of α-average min-bounded*: $S = \{p_2\}$ and $T = N$. Now notice that $\alpha \frac{\sqrt{n}}{n} < 1 \Rightarrow \sqrt{n} < \frac{n}{\alpha}$, which holds since for any $\alpha \geq 1$ that we choose, we can always find a large enough n for the inequality to be true.

Therefore, the intersection of the complements is non-empty and the proof is complete. □

Notice that the proof of this proposition follows by constructing a cost function that requires non-constant values of α to be captured by either of our parameterizations. Although the intersection turns out to be non-empty, the constructed cost function is rather artificial and more natural examples are elusive so far. In fact, for most of the known cost functions that have been studied in the literature, at least one of our mechanisms achieves an $O(H_n)$-approximation. To make further progress, we believe it is important to understand better the class of functions defined by the intersection of the two complements, as it would help us to identify the missing elements for deriving mechanisms for a wider class of cost functions.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their valuable feedback on the paper.

REFERENCES

[1] Georgios Birmpas, Costas Courcoubetis, Ioannis Giotis, and Evangelos Markakis. 2015. Cost-sharing models in participatory sensing. In *Proceedings of the 8th International Symposium on Algorithmic Game Theory*. Martin Hoefer (Ed.), Lecture Notes in Computer Science, Vol. 9347, Springer, 43–56. DOI: https://doi.org/10.1007/978-3-662-48433-3_4

[2] Georgios Birmpas, Evangelos Markakis, and Guido Schaefer. 2019. Cost sharing over combinatorial domains: Complement-free cost functions and beyond. In *Proceedings of the 27th European Symposium on Algorithms*. 20:1–20:17.

[3] Yvonne Bleischwitz and Florian Schoppmann. 2008. Group-strategyproof cost sharing for metric fault tolerant facility location. In *Proceedings of the 1st International Symposium on Algorithmic Game Theory*. Burkhard Monien and Ulf-Peter Schroeder (Eds.), Lecture Notes in Computer Science, Vol. 4997, Springer, 350–361. DOI: https://doi.org/10.1007/978-3-540-79309-0_31

[4] Janina A. Brenner and Guido Schäfer. 2007. Cost sharing methods for makespan and completion time scheduling. In *Proceedings of the 24th Annual Symposium on Theoretical Aspects of Computer Science*. Wolfgang Thomas and Pascal Weil (Eds.), Lecture Notes in Computer Science, Vol. 4393, Springer, 670–681. DOI: https://doi.org/10.1007/978-3-540-70918-3_57

[5] Shuchi Chawla, Tim Roughgarden, and Mukund Sundararajan. 2006. Optimal cost-sharing mechanisms for steiner forest problems. In *Proceedings of the 2nd International Workshop on Internet and Network Economics.* , Paul G. Spirakis, Marios Mavronicolas, and Spyros C. Kontogiannis (Eds.), Lecture Notes in Computer Science, Vol. 4286, Springer, 112–123. DOI: https://doi.org/10.1007/11944874_11

[6] Edward Hammond Clarke. 1971. Multipart pricing of public goods. *Public Choice* 11, 1 (1971), 17–33. DOI: https://doi.org/10.1007/BF01726210

[7] Nikhil R. Devanur, Milena Mihail, and Vijay V. Vazirani. 2005. Strategyproof cost-sharing mechanisms for set cover and facility location games. *Decision Support Systems* 39, 1 (2005), 11–22. DOI: https://doi.org/10.1016/j.dss.2004.08.004

[8] Shahar Dobzinski, Aranyak Mehta, Tim Roughgarden, and Mukund Sundararajan. 2018. Is shapley cost sharing optimal? *Games Economic Behavior.* 108, C (2018), 130–138. DOI: https://doi.org/10.1016/j.geb.2017.03.008

[9] Shahar Dobzinski and Shahar Ovadia. 2018. Combinatorial cost sharing. *SIGecom Exchanges* 16, 2 (2018), 38–44. DOI: https://doi.org/10.1145/3331041.3331046

[10] Tomer Ezra, Michal Feldman, Tim Roughgarden, and Warut Suksompong. 2020. Pricing multi-unit markets. *ACM Transactions on Economics and Computation* 7, 4 (2020), 20:1–20:29. DOI: https://doi.org/10.1145/3373715

[11] Joan Feigenbaum, Arvind Krishnamurthy, Rahul Sami, and Scott Shenker. 2003. Hardness results for multicast cost sharing. *Theoreretical Computer Science* 304, 1–3 (2003), 215–236. DOI: https://doi.org/10.1016/S0304-3975(03)00085-9

[12] Jerry Green, Elon Kohlberg, and Jean Jacques Laffont. 1976. Partial equilibrium approach to the free rider problem. *Journal of Public Economics* 6, 4 (1976), 375–394. DOI: https://doi.org/10.1016/0047-2727(76)90049-9

[13] Theodore Groves. 1973. Incentives in teams. *Econometrica* 41, 3 (1973), 617–631. DOI: https://doi.org/10.2307/1914085

[14] Anupam Gupta, Jochen Könemann, Stefano Leonardi, Ramamoorthi Ravi, and Guido Schäfer. 2015. Efficient cost-sharing mechanisms for prize-collecting problems. *Mathematical Programming* 152, 1–2 (2015), 147–188. DOI: https://doi.org/10.1007/s10107-014-0781-1

[15] Yijie Han. 2008. Matching for graphs of bounded degree. In *Proceedings of the 2nd Annual International Workshop on Frontiers in Algorithmics.* Franco P. Preparata, Xiaodong Wu, and Jianping Yin (Eds.), Lecture Notes in Computer Science, Vol. 5059, Springer, 171–173. DOI: https://doi.org/10.1007/978-3-540-69311-6_19

[16] Shai Herzog, Scott Shenker, and Deborah Estrin. 1997. Sharing the cost of multicast trees: An axiomatic analysis. *IEEE/ACM Transactions on Networking* 5, 6 (1997), 847–860.

[17] Benny Lehmann, Daniel Lehmann, and Noam Nisan. 2006. Combinatorial auctions with decreasing marginal utilities. *Games Economic Behavior* 55, 2 (2006), 270–296. DOI: https://doi.org/10.1016/j.geb.2005.02.006

[18] E. Loehman and A. Whinston. 1974. An axiomatic approach to cost allocation for public investment. *Public Finance Quarterly* 2, 2 (1974), 236–251.

[19] Aranyak Mehta, Tim Roughgarden, and Mukund Sundararajan. 2009. Beyond moulin mechanisms. *Games Economic Behavior* 67, 1 (2009), 125–155. DOI: https://doi.org/10.1016/j.geb.2008.06.005

[20] Herve Moulin. 1999. Incremental cost sharing: Characterization by coalition strategy-proofness. *Social Choice Welfare* 16, 2 (1999), 279–320. DOI: https://doi.org/10.1007/s003550050145

[21] Herve Moulin and Scott Shenker. 2001. Strategyproof sharing of submodular costs: Budget balance vs efficiency. *Economic Theory* 18, 3 (2001), 511–533. DOI: https://doi.org/10.1007/PL00004200

[22] Kevin Roberts. 1979. The characterization of implementable choice rules. In *Proceedings of the Aggregation and Revelation of Preferences*. Jean Jacques Laffont (Ed.), Amsterdam: North Holland.

[23] Tim Roughgarden and Mukund Sundararajan. 2009. Quantifying inefficiency in cost-sharing mechanisms. *Journal of the ACM* 56, 4 (2009), 23:1–23:33. DOI: https://doi.org/10.1145/1538902.1538907

[24] Philip H. Straffin and James P. Heaney. 1981. Game theory and the tennessee valley authority. *International Journal of Game Theory* 10, 1 (1981), 35–43.

[25] William Vickrey. 1961. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of Finance* 16, 1 (1961), 8–37. DOI: https://doi.org/10.1111/j.1540-6261.1961.tb02789.x

[26] H. P. Young, N. Okada, and T. Hashimoto. 1982. Cost allocation in water resources development. *Water Resources Research* 18, 3 (1982), 463–475.

Received January 2021; revised August 2021; accepted November 2021