

One Action Too Many: Inapproximability of Budgeted Combinatorial Contracts

Michal Feldman   

Tel Aviv University, Israel
Microsoft ILDC, Herzliya, Israel

Yoav Gal-Tzur   

Tel Aviv University, Israel

Tomasz Ponitka   

Tel Aviv University, Israel

Maya Schlesinger   

Tel Aviv University, Israel

Abstract

We study multi-agent contract design with combinatorial actions, under budget constraints, and for a broad class of objective functions, including profit (principal’s utility), reward, and welfare. Our first result is a strong impossibility: For submodular reward functions, no randomized poly-time algorithm can approximate the optimal budget-feasible value within *any finite factor*, even with demand-oracle access. This result rules out extending known constant-factor guarantees from either (i) unbudgeted settings with combinatorial actions or (ii) budgeted settings with binary actions, to their combination. The hardness is tight: It holds even when all but one agent have binary actions and the remaining agent has just one additional action. On the positive side, we show that gross substitutes rewards (a well-studied strict subclass of submodular functions) admit a deterministic poly-time $O(1)$ -approximation, using only value queries. Our results thus draw the first sharp separation between budgeted and unbudgeted settings in combinatorial contracts, and identifies gross substitutes as a tractable frontier for budgeted combinatorial contracts. Finally, we present an FPTAS for additive rewards, demonstrating that arbitrary approximation is tractable under any budget. This constitutes the first FPTAS for the multi-agent combinatorial-actions setting, even in the absence of budget constraints.

2012 ACM Subject Classification Theory of computation → Algorithmic game theory

Keywords and phrases Combinatorial Contracts, Algorithmic Contract Design, Budget-Feasible Contracts

Digital Object Identifier 10.4230/LIPIcs.ITCS.2026.58

Related Version *Full Version*: <https://arxiv.org/abs/2511.20110>

Funding This project has been partially funded by the European Research Council (ERC) under the European Union’s Horizon 2020 program (grant agreement No. 866132), by the European Union’s Horizon Europe Program (grant agreement No. 101170373), by an Amazon Research Award, by the Israel Science Foundation Breakthrough Program (grant No. 2600/24), and by a grant from TAU Center for AI and Data Science (TAD), and by the NSF-BSF (grant number 2020788).

1 Introduction

Contract design is a central area in microeconomics that studies how to incentivize strategic agents to exert costly effort in their tasks, e.g., [46, 40, 53, 48, 47]. The significance of this field to modern economics was highlighted by the 2016 Nobel Prize awarded to Hart and Hölmstrom for their contributions to contract theory [54]. In recent years, the rapid emergence of computerized markets for services has fueled increasing interest in the algorithmic aspects of contract design, e.g., [33, 26, 5, 19]; see [32, 37] for recent surveys.



© Michal Feldman, Yoav Gal-Tzur, Tomasz Ponitka, and Maya Schlesinger;
licensed under Creative Commons License CC-BY 4.0

17th Innovations in Theoretical Computer Science Conference (ITCS 2026).

Editor: Shubhangi Saraf; Article No. 58; pp. 58:1–58:24

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

In this work, we focus on the algorithmic challenges of incentivizing teamwork, e.g., [47, 5, 27]. Consider the following illustrative example of a startup hiring a team of engineers. The startup seeks to build a successful product, whose probability of success depends both on the team’s composition (for instance, the degree to which the engineers’ expertise overlaps or complements each other), and on each individual’s effort (for example, whether every engineer carefully reviews code). To encourage engineers to exert costly effort, the startup offers them equity: each employee receives a fraction of ownership in the company.

Scenarios of this type are captured by the multi-agent combinatorial-actions model for contract design [27, 28]. In this model a principal (she, the startup founder) delegates a binary-outcome project to a pool of n potential agents (engineers). The project yields the principal a reward of 1 upon success and 0 if it fails. Each agent $i \in [n]$ has a set T_i of available actions, and may choose any subset $S_i \subseteq T_i$ of them. Each action $j \in T_i$ has a non-negative cost c_j , and the agent i pays the total cost of their chosen actions, $\sum_{j \in S_i} c_j$. A set function f maps every profile of actions $S = S_1 \sqcup \dots \sqcup S_n$ to the project’s success probability $f(S)$. Since the reward is normalized to 1, $f(S)$ also denotes the *expected reward*. To incentivize effort, the principal offers each agent a *linear contract* $\alpha_i \in [0, 1]$, denoting the fraction of the reward that goes to agent i if the project succeeds. So agent i ’s utility, under action profile S , is $\alpha_i f(S) - \sum_{j \in S_i} c_j$. Every contract $\alpha = (\alpha_1, \dots, \alpha_n)$ induces a game among the agents. An action profile S is a *Nash equilibrium* of the game if, given the contract offered, no agent can increase his utility by deviating. In this case, we say that the contract incentivizes action profile S .

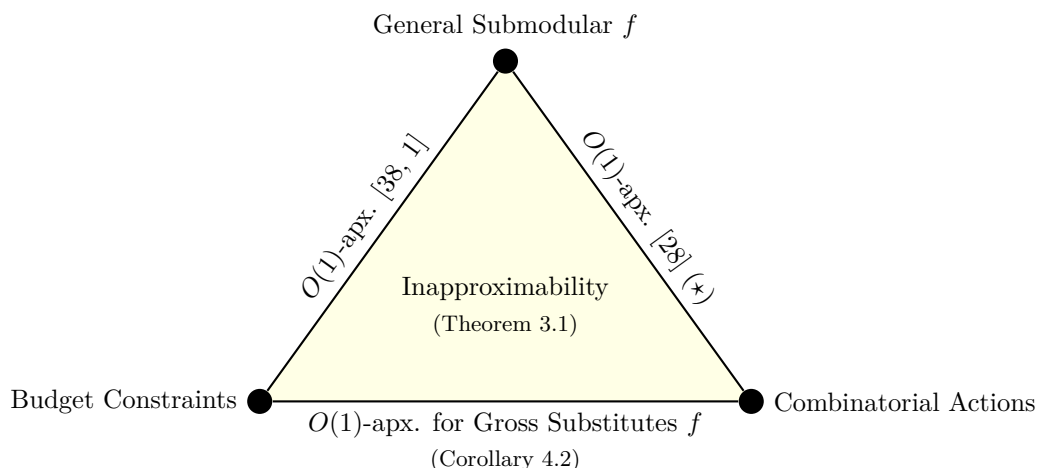
The principal’s objective in [28] is to incentivize an action profile S maximizing her *profit*, defined as her expected reward minus the total expected payments to the agents, i.e., $f(S) \cdot (1 - \sum_{i \in [n]} \alpha_i)$. They present an efficient algorithm that, for submodular f , attains a constant-factor approximation to the optimal profit using demand queries to f (see Section 2 for definitions).

However, practical applications may introduce additional objectives and constraints. Principals may pursue objectives that go beyond profit maximization; for example, maximizing the probability of success $f(S)$, or maximizing total welfare, $f(S) - \sum_{i \in [n]} \sum_{j \in S_i} c_j$. At the same time, they often face constraints, such as budget constraints (e.g., a given portion of the company’s equity must be allocated to investors).

Polynomial-time approximation algorithms for multi-agent contract settings, under key objectives, including reward and welfare, and budget constraints, have been obtained by [38, 1], but only within the *binary-actions* model, where each agent either exerts effort or shirks. In this work, we extend the multi-agent *combinatorial-actions* model of [28] to incorporate budget constraints and to accommodate a broader range of objectives.

1.1 Our Results

We study the design of near-optimal contracts in the multi-agent combinatorial-actions model of [28], under budget constraints, for a broad class of objective functions. In this setting, each agent $i \in [n]$ can select any subset from a given action set, and the principal is restricted by a budget constraint B , i.e., agents may receive at most a B -fraction of the reward. The case of $B = 1$ corresponds to the unbudgeted setting. The class of objectives we consider – termed BEST (BEYond STandard) by [38] – includes reward, welfare, profit, or any convex combination thereof (see Definition 2.4). In what follows we present our main results (also illustrated in Figure 1).



■ **Figure 1** The three vertices of the triangle represent the dimensions along which the settings we consider differ: (i) the structure of f (general submodular vs. gross substitutes), (ii) the presence of budget constraints, and (iii) the type of the agents' action space (binary vs. combinatorial). Any pair of properties admits a constant-factor approximation, as indicated along each edge of the triangle. The figure illustrates that the impossibility arises only from the *combination of all three* properties: the interior inapproximability region corresponds to submodular f with budgets and combinatorial actions simultaneously. (*) All results shown hold for all BEST objectives, except for the $O(1)$ -approximation for the multi-agent combinatorial-actions setting without budget constraints of [28], which holds only for profit maximization.

Inapproximability for Submodular Instances

Our first main result establishes a strong impossibility: when the reward function f is submodular, there is no efficient approximation algorithm to any BEST objective. This impossibility is information theoretic, and does not rely on any computational conjectures.

► **Theorem 1** (Inapproximability for Submodular Instances; Theorem 3.1). *For the class of instances with submodular f , any BEST objective φ (including profit, reward, and welfare), any budget $B \in (0, 1)$, and any approximation guarantee $K : \mathbb{N} \rightarrow [1, \infty)$, any randomized algorithm can achieve a $K(n)$ -approximation with respect to φ under budget B only with exponentially small probability in n , assuming it makes at most polynomially many value and demand queries to f .*

This impossibility is particularly striking in light of known positive results for multi-agent settings with submodular f : (i) constant-factor approximation to profit for combinatorial actions without budget constraints [28], and (ii) constant-factor approximation to any BEST objective in the binary-actions setting with budget constraints [1, 38]. Our result shows that combinatorial actions and budget constraints *together* rule out any approximation. Moreover, our hardness construction constitutes the *minimal* extension needed to obtain such an impossibility: All agents have binary actions, except for one agent which has one extra action.¹

Beyond establishing hardness of approximation, this result provides two important separations. First, between binary and combinatorial actions: Under budget constraints, the optimal-contract problem admits constant-factor approximation guarantees for binary

¹ In fact, we do not use the combinatorial nature of the agent's actions, just that there is more than one.

actions but becomes inapproximable once agents have combinatorial actions. Second, between budgeted and unbudgeted settings: For combinatorial actions, profit maximization admits a constant-factor approximation without budgets, yet becomes inapproximable with budgets.

Constant-Factor Approximation for Gross Substitutes Instances

On the positive side, our second main result shows that efficient constant-factor approximations are possible when f lies in the class of gross substitutes (GS), an important and well-studied subclass of submodular functions [49, 41, 51].

► **Theorem 2** (Constant-Factor Approximation for Gross Substitutes Instances; Corollary 4.2). *For the class of instances with gross substitutes f and any BEST objective φ (including profit, reward, and welfare), there exists a deterministic polynomial-time algorithm that achieves an $O(1)$ -approximation to φ under any budget $B \in [0, 1]$. This algorithm requires only value oracle access to f and φ .*

Theorems 1 and 2 together establish a separation between approximation guarantees under gross substitutes and general submodular functions. Interestingly, a parallel (yet not directly related) separation is known for the problem of computing the *exact* optimal contract in the single-agent combinatorial-actions setting: this problem admits a poly-time algorithm for GS rewards, whereas it is NP-hard for submodular rewards [26].²

Finally, we highlight two other aspects of tightness. Our inapproximability result applies even when the algorithm has access to both value and demand oracles, while our positive approximation guarantee relies only on value oracle access.³ Moreover, our hardness result applies to any randomized algorithm, whereas our proof for gross substitutes f uses a deterministic algorithm.

Discussion: A Three-Way Barrier to Approximation

The multi-agent settings considered in this work differ along three dimensions: (i) the structure of f (general submodular vs. gross substitutes), (ii) the presence of budget constraints, and (iii) the richness of the agents' action space (binary vs. combinatorial). Combined with previous literature, our results show that the inapproximability arises only from the *combination of all three* properties, whereas any pair admits a constant-factor approximation. This is illustrated in Figure 1.

FPTAS for Special Cases

We give an FPTAS for two special cases of interest: instances with additive f and instances with a single agent.

► **Theorem 3** (FPTAS for Additive Instances; Theorem 5.1). *For any budget $B \in [0, 1]$, each of the objectives profit, reward, and welfare admits a deterministic FPTAS with value queries, when f is additive.*

² Whether a separation between GS and submodular rewards also appears in the multi-agent binary-actions setting remains an open problem: While constant-factor approximations are known for all submodular f [27], it remains unknown whether better approximation (e.g., PTAS/FPTAS) can be achieved for GS. We note that computing the exact optimal contract is hard even for additive f .

³ This follows from the fact that when f is GS, a demand query can be computed with poly-many value queries.

This result establishes the first FPTAS for the multi-agent combinatorial-actions model, even without budgets. It generalizes the FPTAS of [27, 38] for the multi-agent binary-actions model, where the latter also considers budgets. We note that, whereas our previous results apply to all BEST objectives, Theorem 3 applies only to profit, reward, and welfare.

We also generalize the FPTAS of [28] for maximizing profit in the *single-agent* setting with combinatorial actions to the budgeted setting.⁴ The proof is deferred to the full version.

► **Theorem 4** (FPTAS for Single-Agent Instances). *In the single-agent setting, there exists an FPTAS for maximizing the principal’s profit under any budget $B \in [0, 1]$, when f is monotone. This algorithm requires access to a demand oracle to f .*

1.2 Our Techniques

Inapproximability for Submodular Instances

A key property in the binary-actions setting with submodular f is the *best-response monotonicity*: given a contract and a Nash equilibrium S , any agent in S continues to find effort a best response, even if a subset of the remaining agents in S shirk. This property is central to achieving constant-factor approximations in the binary-actions setting, both with and without budget constraints [27, 1, 38]. In contrast, best-response monotonicity *fails* in the combinatorial-actions setting with submodular f : under a fixed contract and an equilibrium, if some agents reduce their action sets, the best response of an agent i may also be to reduce his; see, e.g., [28, Example 1.1]. We identify non-monotonicity of the best response as a core aspect of the complexity of the combinatorial-actions setting and our proof of Theorem 1 leverages a carefully engineered violation of this property.

Our construction consists of $n - 1$ binary-action (work or shirk) agents and one special agent with two non-shirking actions: a good action $\mathcal{G}$ and a bad action $\mathcal{B}$. The reward function is at least $1/2$ when the special agent takes $\mathcal{G}$, and infinitesimally small otherwise. Incentivizing the special agent to take $\mathcal{G}$ alone, however, would necessarily violate the budget constraint. By carefully constructing the reward function f , we ensure that there exists a budget-feasible contract that incentivizes the special agent to take $\mathcal{G}$, but only if it simultaneously incentivizes a unique subset A' of exactly half of the remaining agents to exert effort. Since $\mathcal{G}$ can be incentivized only in combination with other agents, this constitutes a violation of best-response monotonicity. The “good” equilibrium – A' together with $\mathcal{G}$ – yields a profit of at least $(1 - B) \cdot (1/2)$, while any other budget-feasible contract fails to incentivize the action $\mathcal{G}$ and thus achieves only an infinitesimally small reward. Therefore, an algorithm must incentivize the “good” equilibrium to achieve any desirable approximation. Yet the number of candidate subsets A' is exponential, and we prove that no efficient method can identify A' using poly-many value or demand queries, establishing our inapproximability result.

Crucially, this construction breaks down for profit maximization in the unbudgeted setting ($B = 1$), where [28] provide an approximation algorithm. Indeed, when $B = 1$, the “good” equilibrium in our construction yields a profit of $(1 - B) \cdot (1/2) = 0$, nullifying the argument.

⁴ Maximizing welfare and reward can be achieved by paying the agent the entire budget upon success.

Constant-Factor Approximation for Gross Substitutes Instances

The first step toward establishing our positive result is proving that when f is gross substitutes, best-response monotonicity holds even when agents have combinatorial actions. The key idea in the proof of this observation is to connect agents' best responses with demand bundles (see Section 2 for formal definitions). More precisely, we observe that agent i 's best response for a given profile of actions of the other agents is a demand bundle with respect to a carefully chosen price vector. We then show that when the set of actions taken by the other agents is reduced to any subset, the corresponding price vector increases coordinate-wise. Therefore, by the gross-substitutes property, agent i is incentivized to take a (weak) superset of the original actions, establishing best-response monotonicity.

We find that best-response monotonicity, combined with tools from [28], yields an up-to-a-constant-factor equivalence between any two BEST objectives under any two budgets, paralleling the result of [38] for the binary-actions case. This equivalence, together with the $O(1)$ -approximation for profit from [28], implies a constant-factor approximation for any BEST objective under any budget.

1.3 Related Work

Combinatorial Contracts

A combinatorial model for contracting agents with binary actions was introduced by [5, 8], where the principal selects a subset of agents to incentivize, and the reward function f is Boolean. This foundational model was later extended to allow mixed strategies [6], and subsequent work studied free-riding in this setting [7].

[27] generalized this framework by considering richer reward functions drawn from the complements-free hierarchy [50], showing that for XOS functions, constant-factor approximations are achievable using demand and value queries. In subsequent work, the same authors extended this to a multi-action setting [28], where agents choose arbitrary subsets of actions, and demonstrated that for submodular f , poly-many demand queries suffice for approximation. [2] considered a multi-project setting where each project corresponds to a binary-action problem, and showed constant-factor approximations for XOS valuations. [16] examined project allocation under fairness constraints, proving inapproximability of envy-freeness in the general case and tractability in constant-size markets.

In the single-agent variant, [26] introduced a model where the agent selects subsets of n costly actions, and the principal's reward is given by a set function f . They showed tractability for gross-substitutes functions using value queries. Follow-up works [22, 30, 34, 29, 39] further refined the complexity landscape in this combinatorial-action setting. Another flavor of the single agent setting, studied in [35], considers a scenario where the agent takes actions sequentially, rather than simultaneously, observing intermediate outcomes. They provide algorithms and hardness results for independent and correlated actions.

The recent work of [36] relaxes the hidden-action assumption by introducing a combinatorial inspection model, where the principal can pay to verify whether the agent's chosen action belongs to a given set, and the inspection cost is combinatorial.

Linear Contracts

Linear contracts are those in which the principal pays each agent a fixed fraction of the reward. The robustness of linear contracts was first established by [14], who proved that linear contracts are max-min optimal when the principal is unaware of the agent's full

action set. This result was later generalized to randomized actions and contracts by [52]. Robustness was also established in a different model of uncertainty by [33], who additionally provided approximation guarantees for linear contracts relative to the optimal ones. Another desirable trait of linear contracts is ambiguity-proofness. That is, unlike general contracts, the principal cannot benefit from introducing ambiguity into linear contracts [31, 25].

Contracts under Budget Constraints

There has been growing interest in the study of budgets in algorithmic contract design. [44] studied the budgeted multi-agent setting where each agent performs an independent task with an observable (binary) outcome. [23] studied a setting where the agent has a budget constraint and show how to convert algorithms for the agent's utility maximization problem to an approximation algorithm with multiplicative and additive guarantees.

Closer to our setting, [38] and [1] introduced budget constraints into the binary-action model of [27]. [38] introduced the notion of BEST objectives and showed that, via a downsizing lemma, any two BEST objectives and any two budgets are equivalent up to a constant factor. As a result, their work also yields constant-factor approximations for XOS and submodular objectives via demand and value queries, respectively. [1] focused on social welfare maximization and studied its divergence from the principal's profit. They have also given constant factor approximation algorithm for the social welfare when the budget $0 \leq B \leq 2 - \Theta(1)$.

Other Contractual Models

Non-combinatorial multi-agent settings with agent externalities are studied in [55, 56, 11]. A complexity measure for the implementation of multi-agent contracts was proposed by [9]. More recently, [13, 19] have studied multi-agent contracts beyond binary outcomes. Contracts where agents have private types are studied in [4, 3, 15, 17, 43, 42, 18]. Several recent works explore the intersection of contracts and learning [57, 21, 45, 10, 20, 24].

2 Model and Preliminaries

In this section we present the multi-agent combinatorial-actions model of [28], the family of BEST (BEyond STandard) objectives of [38].

2.1 The Multi-Agent Combinatorial-Actions Setting

We consider a contractual setting involving a single principal and a set A of n agents. The principal is the owner of a project, which can either succeed or fail. Upon success, the project yields a reward of 1 for the principal, and 0 otherwise. The success probability of the project is determined by the (costly) actions that the agents perform, as specified below. The principal does not observe these actions, only the outcome. In order to incentivize the agents, she offers each $i \in A$ a *linear contract* $\alpha_i \in [0, 1]$ that specifies their payment upon success, while the payment upon failure is always 0.⁵ As we formally establish in the full version of the paper, for all objectives considered in this paper, it is without loss of generality to restrict attention to linear contracts.

⁵ One can also view this model as a multiple-outcome project, where the principal only observes the reward and is restricted to using only linear contracts.

Each agent $i \in A$ has a set of actions T_i from which they may choose any subset. In particular, $T_i = \{j\}$ corresponds to the binary-actions case [27, 38], as the agent may choose either j or $\emptyset$. We assume that any two agents have disjoint sets of actions, i.e., $T_i \cap T_{i'} = \emptyset$ for any $i \neq i'$. Each action $j \in T_i$ is associated with a non-negative cost $c_j \geq 0$, and the cost of a set of actions is additive, i.e., $c(S_i) = \sum_{j \in S_i} c_j$ for any $S_i \subseteq T_i$. In particular, $c(\emptyset) = 0$. The disjoint union of all possible actions is denoted by $T = \bigsqcup_{i \in A} T_i$. Given a profile of actions $S \subseteq T$, we denote by S_i the actions in S taken by agent i , i.e., $S_i = S \cap T_i$. Similarly, we use $S_{-i} = \bigsqcup_{i' \in A \setminus \{i\}} S_{i'}$ to denote the actions taken by all agents *except for* agent i , in the profile of actions S .

As mentioned, we consider a binary-outcome project that yields a reward of 1 to the principal upon success and 0 otherwise. At the heart of the model is a monotone combinatorial set function $f : 2^T \rightarrow [0, 1]$, which maps every profile of actions, $S = \bigsqcup_{i \in A} S_i$, chosen by the agents to the project's success probability. Since the principal's reward upon success is normalized to 1, $f(S)$ also represents the principal's expected reward. We assume that $f(\emptyset) = 0$.

A problem instance is a tuple $\langle A, \{T_i\}_{i \in A}, f, c \rangle$. Here, A is the set of agents, T_i denotes the set of available actions for each agent $i \in A$, f is the reward function, and $c = \{c_j\}_{j \in T}$ represents the costs associated with each action. Recall that the principal offers a (linear) contract $\alpha \in [0, 1]^A$, where α_i is the transfer from the principal to agent $i \in A$, if the project succeeds.

Utilities and Equilibria. For a given contract α and a profile of actions $S = (S_1, \dots, S_n)$, agent i 's expected utility is given by $\alpha_i \cdot f(S_i \sqcup S_{-i}) - c(S_i)$, i.e., by the expected payment minus cost. We say that a contract α *incentivizes* the profile S if S forms a Nash equilibrium (NE) under α . Namely, for every agent $i \in A$ and every alternative set of actions $S'_i \subseteq T_i$, the following holds:

$$\alpha_i \cdot f(S_i \sqcup S_{-i}) - c(S_i) \geq \alpha_i \cdot f(S'_i \sqcup S_{-i}) - c(S'_i).$$

It was shown in [28] that every contract α incentivizes at least one pure Nash equilibrium. A contract α may incentivize multiple equilibria, and we denote this collection by $\text{NE}(\alpha)$.

Given a contract α and an equilibrium $S \in \text{NE}(\alpha)$, the principal's utility, or *profit*, is defined to be the expected reward minus payment, i.e.,

$$u_P(\alpha, S) = \left(1 - \sum_{i \in A} \alpha_i\right) f(S).$$

Maximizing Profit Under Budget Constraints. Given a budget B , we call a contract α *budget-feasible* if $\sum_{i \in A} \alpha_i \leq B$. For convenience, we denote the set of all pairs of budget-feasible contracts and incentivized equilibria⁶ by $\mathcal{C}(B)$, namely,

$$\mathcal{C}(B) = \{(\alpha, S) \mid \sum_{i \in A} \alpha_i \leq B \text{ and } S \in \text{NE}(\alpha)\}.$$

The standard goal of contract design is to maximize the profit, $u_P(\alpha, S)$, which we use to illustrate the budgeted contract design problem: Given a budget B we seek a contract and an equilibrium which approximately maximize the profit, i.e., $(\alpha, S) \in \mathcal{C}(B)$ such that

⁶ A contract may incentivize multiple equilibria, in which case it will appear in $\mathcal{C}(B)$ more than once.

$\gamma \cdot u_P(\alpha, S) \geq \max_{(\alpha', S') \in \mathcal{C}(B)} u_P(\alpha', S')$, for some constant $\gamma \geq 1$. In this work, we also consider additional important objectives, beyond profit maximization, which we elaborate on below.

In light of the observation made in [38, Proposition 3.13], we cannot guarantee to find a contract α for which *every* incentivized equilibrium $S \in \text{NE}(\alpha)$ yields a γ -approximation to the optimal profit, so we must settle for a contract α for which *some* incentivized equilibrium yields a γ -approximation. Therefore, our algorithms output a contract along with an equilibrium which provide the desired guarantees.

Subset Stability and the Doubling Lemma. The notion of *subset stability* and the *doubling lemma* play a crucial role in the analysis of the algorithm of [28], which computes a near-optimal contract with respect to profit without budget constraints. Subset stability is a relaxation of the Nash equilibrium condition: A profile of actions $S = \bigsqcup_{i \in A} S_i$ is subset-stable with respect to a contract α if no agent $i \in A$ strictly benefits from deviating to a subset of S_i .

► **Definition 2.1** (Subset Stability, Definition 3.2 of [28]). *A set of actions S is subset-stable with respect to contract α , if for every agent i , every subset of his actions $S'_i \subseteq S_i$ satisfies*

$$\alpha_i \cdot f(S_i \sqcup S_{-i}) - c(S_i) \geq \alpha_i \cdot f(S'_i \sqcup S_{-i}) - c(S'_i).$$

The doubling lemma shows how a subset-stable profile S with respect to a contract α can be used to incentivize a Nash equilibrium that guarantees at least a half of the expected reward.

► **Lemma 2.2** (Doubling Lemma, Lemma 3.3 of [28]). *Let f be a submodular function and let $\varepsilon > 0$. If S is a subset-stable set of actions with respect to a contract α , then any equilibrium S' with respect to the contract $2\alpha + \varepsilon$ satisfies $f(S') \geq (1/2) \cdot f(S)$, where $\varepsilon = (\varepsilon, \dots, \varepsilon)$.*

Restricted Contracts. For any contract $\alpha \in \mathbb{R}_+^n$ and any set of agents $G \subseteq A$, we denote by $\alpha|_G$ the contract obtained by restricting payments to the agents in G . Namely, we let

$$\alpha|_G = \begin{cases} \alpha_i & \text{if } i \in G, \\ 0 & \text{otherwise.} \end{cases}$$

When $G = \{i\}$ is a singleton, we often omit the brackets and write $\alpha|_i$.

Reward Functions and Access Oracles. In this work we focus on monotone reward function f . We denote by $f(S' | S)$ the marginal contribution of S' to f , given S , i.e., $f(S' | S) = f(S' \cup S) - f(S)$. We also use the notation $f_S(i) = f(\{i\} | S)$. We consider reward functions that belong to one of the following (nested) classes:

- Additive: $f(S) = \sum_{a \in S} f(\{a\})$ for any $S \subseteq T$.
- Gross-Substitutes: For any two price vectors $p, q \in \mathbb{R}^{|T|}$, such that $p \leq q$ coordinate-wise, and any $S^* \in \arg \max_{S \subseteq T} \{f(S) - \sum_{a \in S} p_a\}$, there exists a *demand bundle* with respect to q , i.e., $S' \in \arg \max_{S \subseteq T} \{f(S) - \sum_{a \in S} q_a\}$, such that $S^* \cap \{j \mid p_j = q_j\} \subseteq S'$.
- Submodular: For any $S' \subseteq S \subseteq T$ and any $a \notin S$, it holds that $f_{S'}(a) \geq f_S(a)$.
- Subadditive: For any $S', S \subseteq T$, it holds that $f(S) + f(S') \geq f(S \cup S')$.

It is well-known that, additive $\subsetneq$ gross-substitutes $\subsetneq$ submodular $\subsetneq$ subadditive [50]. As f may have an exponential representation in n (and $|T|$) we assume algorithms access f via two standard models:

- Value oracle: accepts a set $S \subseteq T$ and returns $f(S)$.
- Demand oracle: accepts a vector $p \in \mathbb{R}^{|T|}$ and returns $S^* \in \arg \max_{S \subseteq T} \{f(S) - \sum_{a \in S} p_a\}$. Generally, demand oracles are strictly stronger than value oracles [12]. However, for gross-substitutes f a demand query can be computed in poly-time with value oracle access [51].

2.2 Objectives and Maximization Problems

Our results apply to a variety of objectives beyond the standard goal of maximizing profit. Analogously to [38], we consider a family of natural objectives that are defined with respect to a contract α and the set of actions $S = \bigsqcup_{i \in A} S_i$ taken by the agents. We begin by presenting a precise definition of an objective.

► **Definition 2.3** (Objectives in the Multi-Agent Combinatorial-Actions Model). *An objective φ is defined by a poly-time algorithm that, given a problem instance $\langle A, \{T_i\}_{i \in A}, f, c \rangle$, a contract α , and a subset of actions $S \subseteq T$, outputs a non-negative real number, denoted $\varphi_{\langle A, \{T_i\}_{i \in A}, f, c \rangle}(\alpha, S)$. This algorithm has value oracle access to f . We omit the subscript if the instance is clear from context.*

Observe that the standard goal of maximizing profit adheres to this definition: Given a contract α and a set of actions S a simple poly-time algorithm can compute $u_P(\alpha, S) = (1 - \sum_{i \in A} \alpha_i) \cdot f(S)$ using a single value query to f .

For a given objective φ and budget $B \in (0, 1]$, we denote by $\text{MAX-}\varphi(B)$ the problem of finding a budget-feasible contract and equilibrium $(\alpha, S) \in \mathcal{C}(B)$, that maximize φ . We sometimes abuse notation and use $\text{MAX-}\varphi(B)$ to denote the maximal value of φ achievable under budget B .

Below we define the class of BEST objectives, which naturally extends the definition for the binary-actions setting of [38].

► **Definition 2.4** (BEST Objectives). *An objective φ belongs to the class of beyond standard (BEST) objectives if, for any instance $\langle A, \{T_i\}_{i \in A}, f, c \rangle$, it is:*

- (i) Sandwiched between profit and reward: *For any α and $S \subseteq T$, $u_P(\alpha, S) \leq \varphi(\alpha, S) \leq f(S)$.*
- (ii) Decomposable: *For any α , any $S \subseteq T$, and any $i \in A$, $\varphi(\alpha, S) \leq f(S_{-i}) + \varphi(\alpha|_i, S_i)$.*
- (iii) Weakly increasing in S : *For any α and any $S \subseteq S' \subseteq T$, $\varphi(\alpha, S) \leq \varphi(\alpha, S')$.*
- (iv) Weakly decreasing in α : *For any $\alpha \leq \alpha'$ (coordinate-wise) and $S \subseteq T$, $\varphi(\alpha, S) \geq \varphi(\alpha', S)$.*

The first two conditions generalize the definition of BEST objectives from [38]. The richer combinatorial-actions setting considered here also requires conditions (iii) and (iv), which are natural, and clearly hold for common objectives such as profit, welfare and reward.

In the full version of the paper we show that the class of BEST objectives includes the standard objectives of profit (maximizing the principal's utility), welfare ($\max_{\alpha, S \in \text{NE}(\alpha)} f(S) - c(S)$) and reward ($\max_{\alpha, S \in \text{NE}(\alpha)} f(S)$). We also show that this class is closed under convex combinations.

3 Inapproximability Result for Submodular Instances

In this section we show that when f is submodular, for any budget $B \in (0, 1)$, and any BEST objective φ , it holds that $\text{MAX-}\varphi(B)$ cannot be efficiently approximated with demand query access. This presents a significant departure from the unbudgeted case, where [28] present a poly-time constant-factor approximation algorithm for profit, i.e., for $\text{MAX-PROFIT}(1)$.

► **Theorem 3.1** (Inapproximability for Submodular Instances). *Fix any BEST objective φ and any budget $B \in (0, 1)$. For any approximation guarantee $K : \mathbb{N} \rightarrow [1, \infty)$, any (randomized) poly-time algorithm with demand oracle access to f may only achieve a $K(n)$ -approximation to $\text{MAX-}\varphi(B)$ with exponentially-small probability (in n).*

By setting $K(n) = 2^n$ in Theorem 3.1, we derive the following corollary.

► **Corollary 3.2** (Exponential Lower Bound on Expected Approximation Ratio). *Fix any BEST objective φ , and any budget $B \in (0, 1)$. Let $\mathcal{A}$ be any (randomized) poly-time algorithm with demand oracle access to f , and denote its output by (α, S) . Then, there exists a family of instances such that $\text{MAX-}\varphi(B)/\mathbb{E}_{\mathcal{A}}[\varphi(\alpha, S)] = 2^{\Omega(n)}$.*

Our result is information-theoretic: We construct an instance with a probability distribution over submodular reward functions and establish an upper bound on the expected performance of any deterministic algorithm with demand oracle access on this randomized input. By Yao's principle, this directly implies the statement of the theorem.

Let us now define the instances used in the proof of Theorem 3.1.

► **Definition 3.3** (Parameterized Instances). *Fix a budget $B \in (0, 1)$, an approximation guarantee $K : \mathbb{N} \rightarrow [1, \infty)$, and any even $n > 0$. Fix any $\varepsilon > 0$ such that:*

$$\varepsilon < \min \left(\frac{1 - B}{K(n) \cdot (n + 4)}, \frac{4 \cdot n}{B} \right).$$

For any $A' \subseteq [n]$ with $|A'| = n/2$, we define an instance $\mathcal{I}^{(A')} = \langle A, \sqcup T_i, f^{(A')}, c \rangle$ as follows.

- The set of agents is $A = \{1, \dots, n + 1\}$.
- Each agent $i \in [n]$ controls a single action $T_i = \{i\}$. Additionally, agent $n + 1$ controls two actions $T_{n+1} = \{\mathcal{B}, \mathcal{G}\}$. That is, the total set of actions is $T = [n] \cup \{\mathcal{B}, \mathcal{G}\}$, ($\mathcal{B}$ for “bad”, $\mathcal{G}$ for “good”).
- The costs of the actions are:

$$c_i = \varepsilon^3 \text{ for all } i \in [n], \quad c_{\mathcal{G}} = (1/2) \cdot (B - (n/2) \cdot \varepsilon^2) \quad \text{and} \quad c_{\mathcal{B}} = (3/2) \cdot \varepsilon \cdot B.$$

- The reward function $f^{(A')}$ is defined as a sum of three set functions. Specifically, we let $f^{(A')}(S) = f_1(S) + f_2(S) - f_3(S, A')$, where:

$$f_1(S) = \max((1/2) \cdot \mathbf{1}[\mathcal{G} \in S], \varepsilon \cdot \mathbf{1}[\mathcal{B} \in S])$$

$$f_2(S) = \varepsilon \cdot \min(|S \setminus \{\mathcal{G}\}|, n/2 + 1)$$

$$f_3(S, A') = (\varepsilon/2) \cdot \mathbf{1}[S = \{\mathcal{B}\} \cup A']$$

Whenever it is clear from the context, we omit A' from the reward function, and write simply f . Observe that f_1 is a (weighted) unit-demand function over $\{\mathcal{G}, \mathcal{B}\}$ and f_2 is a uniform $(n/2 + 1)$ -demand⁷ over $[n] \cup \{\mathcal{B}\}$.

We give a brief proof sketch for Theorem 3.1: First, we observe that obtaining a good approximation for any BEST objective requires a good approximation to f . By design, this is only possible when agent $n + 1$ chooses action $\mathcal{G}$, since any set containing $\mathcal{G}$ has value greater than $1/2$, whereas any set that excludes $\mathcal{G}$ has value at most $(n/2 + 2) \cdot \varepsilon$. However, the only way to incentivize agent $n + 1$ to take $\mathcal{G}$ instead of $\mathcal{B}$, while complying with the

⁷ A set function $f : 2^A \rightarrow \mathbb{R}_{\geq 0}$ is *unit-demand* if $f(S) = \max_{i \in S} f(\{i\})$. Moreover, f is *uniform k -demand* if there exists $v \in \mathbb{R}_{\geq 0}$ such that $f(S) = \min\{|S|, k\} \cdot v$.

budget, is to incentivize the set A' to take action. This is because the marginal reward of $\mathcal{B}$ is reduced when the agents of A' exerts effort (as captured by f_3). Consequently, a good approximation can only be achieved when the equilibrium satisfies $S_{-(n+1)} = A'$, implying that the algorithm must effectively “know” the set A' . In the case of value queries alone, a standard “hide a special set” argument shows that any algorithm identifying A' with non-negligible probability must make exponentially many value queries. Finally, to complete the proof, we demonstrate that access to demand queries does not help, as any demand query can be simulated using $O(1)$ value queries for our choice of f .

We now move to the formal analysis. Before proving Theorem 3.1, we present some useful lemmas. We first establish that $f^{(A')}$ is monotone and submodular. The proof is deferred to the full version.

► **Lemma 3.4.** *For any $A' \subseteq [n]$ with $|A'| = n/2$, it holds that $f^{(A')}$ is monotone and submodular.*

We next show that demand queries to $f^{(A')}$ are not more powerful than value queries.

► **Lemma 3.5.** *For any $A' \subseteq [n]$ with $|A'| = n/2$, any demand query to $f^{(A')}$ can be computed with 12 value queries to $f^{(A')}$.*

Proof. Fix a subset $A' \subseteq [n]$ with $|A'| = n/2$. Let $p \in \mathbb{R}_{\geq 0}^{n+1}$ be a price vector. Without loss of generality, assume that $p_1 \leq \dots \leq p_n$. Let k be a maximal index such that $p_k < \varepsilon$ or 0 if no such index exists. Let $\tau = \min\{k, n/2 + 1\}$.

We claim that one of the sets $\{\{1, \dots, \tau\}, \{1, \dots, \tau - 1\}, \{1, \dots, \tau - 2, \tau\}\}$ combined with one of $\{\emptyset, \{\mathcal{G}\}, \{\mathcal{B}\}, \{\mathcal{G}, \mathcal{B}\}\}$ is a demand bundle. Once the claim is proven, it follows that 12 value queries suffice to answer a demand query.

To prove the claim, let $S^* \in \arg \max_{S \subseteq T} \{f(S) - \sum_{j \in S} p_j\}$, be a set in the demand with respect to price vector p , and let $S_0 = S^* \cap \{\mathcal{G}, \mathcal{B}\}$. The marginal utility of $S' \subseteq [n]$ with respect to S is:

$$u(S' \mid S_0) = f(S' \mid S_0) - \sum_{i \in S'} p_i$$

$$= \begin{cases} \varepsilon \cdot \min(|S'|, n/2 + 1) - \sum_{i \in S'} p_i & \text{if } S_0 = \emptyset \text{ or } S_0 = \{\mathcal{G}\} \\ \varepsilon \cdot \min(|S'|, n/2) - \sum_{i \in S'} p_i & \text{if } S_0 = \{\mathcal{B}, \mathcal{G}\} \\ \varepsilon \cdot \min(|S'|, n/2) - (\varepsilon/2) \cdot \mathbf{1}[S' = A'] - \sum_{i \in S'} p_i & \text{if } S_0 = \{\mathcal{B}\} \end{cases}$$

We claim that one of $\{\{1, \dots, \tau\}, \{1, \dots, \tau - 1\}, \{1, \dots, \tau - 2, \tau\}\}$ maximizes $u(S' \mid S_0)$.

Indeed, if $S_0 = \emptyset$ or $S_0 = \{\mathcal{G}\}$, clearly $S' = \{1, \dots, \tau\}$ maximizes $u(S' \mid S_0)$, and we are done. If $S_0 = \{\mathcal{B}, \mathcal{G}\}$, clearly $S' = \{1, \dots, \min(k, n/2)\}$ maximizes $u(S' \mid S_0)$, and by definition of τ , we have $\min(k, n/2) \in \{\tau, \tau - 1\}$, implying that either $\{1, \dots, \tau\}$ or $\{1, \dots, \tau - 1\}$ maximizes $u(S' \mid S_0)$, as needed. Otherwise, $S_0 = \{\mathcal{B}\}$. If $\{1, \dots, \tau - 1\} = A'$, then since $|A'| = n/2$, we must have $\tau = n/2 + 1$, and so either $\{1, \dots, \tau - 2, \tau\}$ or $\{1, \dots, \tau - 1\}$ maximizes $u(S' \mid S_0)$. If $\{1, \dots, \tau\} = A'$, then since $|A'| = n/2$, we must have $p_{\tau+1} \geq \varepsilon$. We get that either $\{1, \dots, \tau - 1\}$ or $\{1, \dots, \tau\}$ maximizes $u(S' \mid S_0)$. If $\{1, \dots, \tau - 1\} \neq A'$ and $\{1, \dots, \tau\} \neq A'$, then, either $\{1, \dots, \tau - 1\}$ or $\{1, \dots, \tau\}$ maximizes $u(S' \mid S_0)$. ◀

Given the lemma above, it is enough to show that no algorithm can obtain a good approximation using only value queries.

The following lemma shows that it is possible to incentivize the set of actions $A' \cup \mathcal{G}$ without violating the budget constraints.

► **Lemma 3.6.** *For any instance $\mathcal{I}^{(A')}$ with $A' \subseteq [n]$ and $|A'| = n/2$, there exists a contract α such that $(\alpha, A' \cup \{\mathcal{G}\}) \in \mathcal{C}(B)$.*

Proof. Consider the contract

$$\alpha_i = \begin{cases} \varepsilon^2 & \text{if } i \in A' \\ B - (n/2) \cdot \varepsilon^2 & \text{if } i = n+1 \\ 0 & \text{otherwise,} \end{cases}$$

and let $S = A' \cup \{\mathcal{G}\}$. Clearly $f(S) \geq f(\mathcal{G}) \geq 1/2$, we now show that $S \in \text{NE}(\alpha)$. Let $j \in [n] \setminus A'$. Since $\alpha_j = 0$, $S_j = \emptyset$ is a best response for agent j . Let $i \in A'$. Note that

$$\alpha_i \cdot f(i \mid S_{-i}) = \varepsilon^2 \cdot f_2(i \mid S_{-i}) = \varepsilon^2 \cdot \varepsilon = c_i,$$

so $S_i = \{i\}$ is a best response for agent i since his only choices are $\{i\}$ and $\emptyset$. We now turn to agent $n+1$. Note that

$$\alpha_{n+1} \cdot f(\mathcal{B} \mid S_{-(n+1)}) = \alpha_{n+1} \cdot f(\mathcal{B} \mid A') = (B - (n/2) \cdot \varepsilon^2) \cdot (3/2) \cdot (\varepsilon) < c_{\mathcal{B}},$$

and so since f is submodular, it holds that agent $(n+1)$'s best response does not contain $\mathcal{B}$. It therefore remains to show that agent $(n+1)$'s utility from $\mathcal{G}$ is non-negative. Indeed,

$$\alpha_{n+1} \cdot f(\mathcal{G} \mid S_{-(n+1)}) = \alpha_{n+1} \cdot f(\mathcal{G} \mid A') = (B - (n/2) \cdot \varepsilon^2) \cdot (1/2) = c_{\mathcal{G}},$$

as needed. ◀

Next, we show that incentivizing $A' \cup \mathcal{G}$ is necessary to get a non-trivial approximation to f .

► **Lemma 3.7.** *For any instance $\mathcal{I}^{(A')}$ with $A' \subseteq [n]$ and $|A'| = n/2$, for any $(\alpha, S) \in \mathcal{C}(B)$ with $S \neq \{\mathcal{G}\} \cup A'$, it holds that $f(S) \leq (n/2 + 2) \cdot \varepsilon$.*

Proof. Fix a subset $A' \subseteq [n]$ with $|A'| = n/2$. Let (α, S) be such a budget-feasible contract and equilibrium. We first observe that it cannot be that $\{\mathcal{G}, \mathcal{B}\} \subseteq S$. This is because if $\{\mathcal{G}, \mathcal{B}\} \subseteq S$, then S is not budget-feasible due to submodularity of f :

$$\alpha_{n+1} \geq c_{\mathcal{B}}/f_S(\mathcal{B}) \geq c_{\mathcal{B}}/f(\mathcal{B} \mid \mathcal{G}) \geq c_{\mathcal{B}}/\varepsilon = (3/2) \cdot B > B.$$

Note that for any $S \subseteq [n] \cup \{\mathcal{B}\}$, it holds that $f(S) \leq f_1(S) + f_2(S) \leq (n/2 + 2) \cdot \varepsilon$, so proving $\mathcal{G} \notin S$ is sufficient. Assume towards contradiction that $\mathcal{G} \in S$. Since $\{\mathcal{G}, \mathcal{B}\} \subsetneq S$, we must have $S_{n+1} = \{\mathcal{G}\}$, and

$$\alpha_{n+1} \geq c_{\mathcal{G}}/f_S(\mathcal{G}) \geq c_{\mathcal{G}}/f(\{\mathcal{G}\}) = B - (n/2) \cdot \varepsilon^2.$$

Observe that $|S \cap [n]| \leq n/2$ since incentivizing any agent $i \in [n]$ to exert effort takes at least $c_i/f(\{i\}) = \varepsilon^2$, and therefore we can only incentivize $n/2$ such agents, as the remaining budget is $B - \alpha_{n+1} \leq (n/2) \cdot \varepsilon^2$. Since $|S \cap [n]| \leq n/2$, $S_{n+1} = \{\mathcal{G}\}$, and $S \neq \{\mathcal{G}\} \cup A'$, we have $f(\mathcal{B} \mid S_{-(n+1)}) = 2\varepsilon$. Then, we have

$$\alpha_{n+1} f(\mathcal{B} \mid S_{-(n+1)}) - c_{\mathcal{B}} \geq (B - (n/2) \varepsilon^2) \cdot (2\varepsilon) - (3/2) B \varepsilon \geq (B/2) \varepsilon - n \varepsilon^3 > (B/2) \varepsilon - n \varepsilon^2.$$

Moreover, by budget feasibility we have $\alpha_{n+1} \leq B$, and by the definition of f we have $f(\mathcal{G} \mid S_{-(n+1)}) \leq 1/2$, which gives:

$$\alpha_{n+1} \cdot f(\mathcal{G} \mid S_{-(n+1)}) - c_{\mathcal{G}} \leq B \cdot (1/2) - (1/2) \cdot (B - (n/2) \cdot \varepsilon^2) = (n/4) \cdot \varepsilon^2 < n \cdot \varepsilon^2.$$

By our choice of ε , we have $\varepsilon < 4n/B$, and agent $n+1$ would therefore benefit from deviating to $S_{n+1} = \{\mathcal{B}\}$, which gives a contradiction. ◀

We are now ready to prove Theorem 3.1.

Proof of Theorem 3.1. Fix a BEST objective φ and a budget $B \in (0, 1)$. By Yao's principle, it suffices to prove the statement for a deterministic algorithm against a randomized input. We consider a randomized instance $\mathcal{I}^{(A')}$, where $A' \subseteq [n]$ is chosen uniformly at random from all subsets of size $n/2$. By Lemma 3.4, this defines a distribution over monotone and submodular instances. Now, consider a polynomial-time deterministic algorithm with access to value and demand oracles on this randomized input.

By Lemma 3.6 and the definition of BEST objectives, the optimal value of φ is at least a $(1 - B)$ -fraction of the profit from the budget-feasible contract $(\alpha, A' \cup \mathcal{G}) \in \mathcal{C}(B)$, meaning that:

$$\text{MAX-}\varphi(B) \geq \varphi(\alpha, A' \cup \mathcal{G}) \geq u_P(\alpha, A' \cup \mathcal{G}) \geq (1 - B) \cdot f(A' \cup \mathcal{G}) \geq (1 - B) \cdot (1/2).$$

Moreover, by Lemma 3.7, the value of φ for a contract and equilibrium $(\alpha, S) \in \mathcal{C}(B)$ with $S \neq \{\mathcal{G}\} \cup A'$ is:

$$\varphi(\alpha, S) \leq f(S) \leq (n/2 + 2) \cdot \varepsilon.$$

Thus, unless the algorithm outputs the equilibrium $A' \cup \mathcal{G}$, it achieves at best an approximation of $(1 - B) \cdot (1/2) / ((n/2 + 2) \cdot \varepsilon) > K(n)$, since by our choice of ε we have $\varepsilon < (1 - B) / (K(n) \cdot (n + 4))$.

By Lemma 3.5, a polynomial-time algorithm with access to value and demand queries can be simulated using polynomially many value queries. Thus, it remains to show that any algorithm making only polynomially many value queries cannot output $\mathcal{G} \cup A'$ with better than exponentially small probability.

We assume, without loss of generality, that the algorithm queries the value of the set $S_{-(n+1)} \cup \{\mathcal{B}\}$ (where S is the output equilibrium). This is without loss because any algorithm can be modified to perform one additional value query before terminating, without affecting its polynomial query complexity. We will upper bound the probability that the algorithm queries $A' \cup \{\mathcal{B}\}$, thereby establishing the same upper bound on the probability that it achieves a $K(n)$ -approximation.

Let $S_1, \dots, S_\ell$ be the sequence of value queries that the (deterministic) algorithm makes on the instance $\mathcal{J} = \langle A, \sqcup T_i, f_1 + f_2, c \rangle$. Unless the algorithm queries $A' \cup \{\mathcal{B}\}$, this instance $\mathcal{J}$ is indistinguishable from $\mathcal{I}^{(A')}$. Thus, the probability that the algorithm queries $A' \cup \{\mathcal{B}\}$ is upper bounded by the probability that $A' \cup \{\mathcal{B}\} \in \{S_1, \dots, S_\ell\}$. Therefore, by the union bound, the probability of querying $A' \cup \{\mathcal{B}\}$ is at most $\ell / \binom{n}{n/2}$. Since ℓ is polynomial in n , this probability is exponentially small in n , as needed. $\blacktriangleleft$

4 Constant-Factor Approximation for Gross Substitutes Instances

In this section we establish an up-to-a-constant-factor equivalence between any two BEST objectives, generalizing the result of [38] to the multi-agent combinatorial-actions setting under gross substitutes f . This equivalence is cast in the following theorem.

► **Theorem 4.1** (Equivalence of All BEST Objectives and Budgets). *Fix any two BEST objectives φ, φ' and any two budget $B, B' \in (0, 1]$. For gross substitutes f , there exists a poly-time reduction from $\text{MAX-}\varphi(B)$ to $\text{MAX-}\varphi'(B')$ that loses only a constant factor in the approximation. This reduction requires value oracle access to f .*

The following corollary follows directly from combining the above equivalence with the poly-time algorithm of [28] for maximizing profit with budget $B = 1$ for submodular (and hence also gross substitutes) f .

► **Corollary 4.2** (Constant-Factor Approximations Under Budget Constraints). *When f is gross substitutes, for any BEST objective φ and any budget $B \in [0, 1]$, there exists a polynomial-time algorithm that achieves $O(1)$ -approximation to $\text{MAX-}\varphi(B)$ using value queries.*

To establish Theorem 4.1, we follow a scheme similar to that of [38], which proved an analogous result in the binary-action setting. In Section 4.1, we identify a key property, *best-response monotonicity*, that enables this equivalence. This property holds both in the binary-action case for submodular f and in the combinatorial-actions case for gross substitutes f (Lemma 4.3). Crucially, for combinatorial actions with submodular f , best-response monotonicity fails to hold, an observation we exploit in the negative result of Section 3. We use best-response monotonicity to prove a downsizing lemma (Lemma 4.5) for the combinatorial-actions setting, analogous to the argument in [38].

In Section 4.2 we define an auxiliary problem, $\text{MAX-REWARD-BOUNDED}(B)$. We show that for any BEST objective φ and any budget B , a constant-factor approximation to $\text{MAX-}\varphi(B)$ can be obtained either from the solution to $\text{MAX-REWARD-BOUNDED}(B)$ or from a contract that incentivizes only a single agent to take some actions.

4.1 Best-Response Monotonicity for Gross Substitutes Instances

In this section we present the main property of gross substitutes that allows us to apply our techniques. Roughly speaking, we show that when f is gross substitutes, incentivizing an agent to take a given subset of actions is always cheapest for the principal when all other agents do nothing. This does not hold for submodular f , as exemplified by our construction in Section 3.

► **Lemma 4.3** (Best-Response Monotonicity). *Consider any instance with a gross substitutes f . Fix a contract α , an equilibrium $S \in \text{NE}(\alpha)$, and an agent $i \in A$. Take any subset of actions $S'_{-i} \subseteq S_{-i}$. Then, there exists $S'_i \subseteq T_i$ such that $S_i \subseteq S'_i$ and S'_i is agent i 's best response to S'_{-i} , i.e., for every alternative set of actions $\tilde{S}_i \subseteq T_i$, it holds that $\alpha_i \cdot f(S'_i \sqcup S'_{-i}) - c(S'_i) \geq \alpha_i \cdot f(\tilde{S}_i \sqcup S'_{-i}) - c(\tilde{S}_i)$.*

Proof. Fix agent $i \in A$, contract α , actions $S \in \text{NE}(\alpha)$, and $S'_{-i} \subseteq S_{-i}$ satisfying the above conditions. Consider two price vectors $\{p_a\}_{a \in T}$ and $\{q_a\}_{a \in T}$ defined as:

$$p_a = \begin{cases} -1 & \text{if } a \in S_{-i} \\ 2 & \text{if } a \in T_{-i} \setminus S_{-i} \\ (1/\alpha_i) \cdot c_a & \text{if } a \in T_i, \end{cases} \quad q_a = \begin{cases} -1 & \text{if } a \in S'_{-i} \\ 2 & \text{if } a \in T_{-i} \setminus S'_{-i} \\ (1/\alpha_i) \cdot c_a & \text{if } a \in T_i. \end{cases}$$

We observe that the following equivalences hold for any action profile $S' = S'_i \sqcup S'_{-i}$:

$$\begin{aligned}
& S'_i \text{ is agent } i\text{'s best response to } S'_{-i} \text{ given } \alpha_i \\
\iff & S'_i \in \arg \max_{U'_i \subseteq T_i} \left\{ f(U'_i \cup S'_{-i}) - \sum_{a \in U'_i} (1/\alpha_i) \cdot c_a \right\} \\
\iff & S' = S'_i \sqcup S'_{-i} \in \arg \max_{V \subseteq T} \left\{ f(V) - \sum_{a \in V} q_a \right\}.
\end{aligned}$$

The first equivalence is observed by [28]. We next show the second equivalence. Let $Q \in \arg \max_{V \subseteq T} \{f(V) - \sum_{a \in V} q_a\}$. We argue that $Q_{-i} = S'_{-i}$; combined with the definition of q , this implies the equivalence. On the one hand, since $f(U) \leq 1$ for all $U \subseteq T$ and $q_a > 1$ for all $a \in T_{-i} \setminus S'_{-i}$, we have $Q_{-i} \subseteq S'_{-i}$. On the other hand, since f is weakly monotone and $q_a = -1$ for all $a \in S'_{-i}$, it must be that $S'_{-i} \subseteq Q_{-i}$. Thus, $Q_{-i} = S'_{-i}$.

A similar equivalence also holds with respect to S and p . Namely, $U_i \subseteq T_i$ is agent i 's best response to S_{-i} given α_i if and only if $U_i \cup S_{-i} \in \arg \max_{V \subseteq T} \{f(V) - \sum_{a \in V} p_a\}$. Thus, since S is an equilibrium, it is a demand bundle with respect to f and p . Note that $p_a \leq q_a$ for all $a \in T$ by the assumption that $S'_{-i} \subseteq S_{-i}$. Thus, by the gross substitutes property of f , there exists a set S'_i such that $S_i \subseteq S'_i \subseteq T_i$ and $S'_i \cup S'_{-i} \in \arg \max_{V \subseteq T} \{f(V) - \sum_{a \in V} q_a\}$, as needed. $\blacktriangleleft$

Lemma 4.3 immediately yields the following important corollary. Recall that $\alpha|_i$ is the contract which offers α_i to agent i and zero to all other agents.

► **Corollary 4.4.** *Consider an instance with a gross substitutes f . Fix a contract α , an equilibrium $S \in \text{NE}(\alpha)$, and any agent $i \in A$. Then there exists an equilibrium $S' \in \text{NE}(\alpha|_i)$ such that $S_i \subseteq S'_i$ and $S'_j = \emptyset$ for any $j \neq i$.*

Corollary 4.4 together with Lemma 2.2 enable us to prove a downsizing lemma akin to [38, Lemma 3.2] for the combinatorial-actions setting. We defer the details of the algorithm and the proof of the lemma to the full version.

► **Lemma 4.5 (Downsizing Lemma for Combinatorial Actions).** *Let $\langle A, \{T_i\}_{i \in A}, f, c \rangle$ be any multi-agent combinatorial-actions instance with gross substitutes f . For any integer $M \geq 3$ and any $(\alpha, S) \in \mathcal{C}(B)$, there exists $(\alpha', S') \in \mathcal{C}(B)$ such that:*

$$\left(\sum_{i \in A} \alpha'_i \leq \frac{5}{M} \cdot \sum_{i \in A} \alpha_i \text{ or } \exists i \in A \text{ s.t. } \alpha' = \alpha|_i \text{ and } S' \subseteq T_i \right) \text{ and } f(S') \geq \frac{f(S)}{2M-2}.$$

Moreover, such a pair $(\alpha', S') \in \mathcal{C}(B)$ can be computed in poly-time with value query access to f .

4.2 BEST Objectives Are Equivalent

To establish the up-to-a-constant-factor equivalence between any two BEST under any two budgets, we define two maximization problems (i) finding the optimal budget-feasible contract which incentivizes a single agent, and (ii) finding an optimal budget-feasible contract and an equilibrium where the payment to each agent is at most a $3/4$ -fraction of the budget.

► **Definition 4.6** ($\text{BEST-SINGLE}_i\text{-}\varphi(B)$). For any given objective φ and budget $B \in (0, 1]$, the problem of $\text{BEST-SINGLE}_i\text{-}\varphi(B)$ is the problem of finding an optimal single-agent contract for agent i :

$$\text{BEST-SINGLE}_i\text{-}\varphi(B) = \max_{(\alpha, S) \in \mathcal{C}(B)} \varphi(\alpha, S) \quad \text{subject to} \quad \alpha = \alpha|_i \text{ and } S \subseteq T_i.$$

When clear from context, we also use $\text{BEST-SINGLE}_i\text{-}\varphi(B)$ to denote a pair $(\alpha, S) \in \mathcal{C}(B)$ maximizing φ subject to $\alpha = \alpha|_i$ and $S \subseteq T_i$.

► **Definition 4.7** ($\text{MAX-REWARD-BOUNDED}(B)$). Let $\langle A, \{T_i\}_{i \in A}, f, c \rangle$ be an problem instance. For any $B \in (0, 1]$, the $\text{MAX-REWARD-BOUNDED}(B)$ problem is defined as

$$\text{MAX-REWARD-BOUNDED}(B) = \max_{(\alpha, S) \in \mathcal{C}(B)} f(S) \quad \text{subject to} \quad \alpha_i \leq 3B/4 \text{ for all } i \in A.$$

The best-response monotonicity of gross substitutes instances (Corollary 4.4) is crucial for the proof of the following lemma.

► **Lemma 4.8** (Decomposition Lemma). Fix an instance $\langle A, \{T_i\}_{i \in A}, f, c \rangle$ with gross substitutes f , a budget $B \in (0, 1]$, and a BEST objective φ . It holds that

$$\text{MAX-}\varphi(B) \leq 2 \cdot \text{MAX-REWARD-BOUNDED}(B) + \max_{i \in A} \text{BEST-SINGLE}_i\text{-}\varphi(B).$$

Proof. Let $(\alpha^*, S^*) \in \mathcal{C}(B)$ be a solution to $\text{MAX-}\varphi(B)$.

If $\alpha_i^* \leq (3/4) \cdot B$ for all agents $i \in A$, then $f(S^*) \leq \text{MAX-REWARD-BOUNDED}(B)$ and we get

$$\begin{aligned} \text{MAX-}\varphi(B) &= \varphi(\alpha^*, S^*) && \text{(by the choice of } (\alpha^*, S^*)) \\ &\leq f(S^*) && \text{(by Definition 2.4i)} \\ &\leq \text{MAX-REWARD-BOUNDED}(B) && \text{(as } (\alpha^*, S^*) \in \mathcal{C}(B) \text{ and by assumption),} \end{aligned}$$

as needed.

Otherwise, let $i \in A$ be the agent such that $\alpha_i^* > (3/4) \cdot B$; observe that by budget-feasibility there can be at most one such agent. It follows from Definition 2.4ii that

$$\text{MAX-}\varphi(B) = \varphi(\alpha^*, S^*) \leq f(S_{-i}^*) + \varphi(\alpha^*|_i, S_i^*).$$

It remains to bound each of the summands, i.e., $f(S_{-i}^*) \leq 2 \cdot \text{MAX-REWARD-BOUNDED}(B)$, and $\varphi(\alpha^*|_i, S_i^*) \leq \text{BEST-SINGLE}_i\text{-}\varphi(B)$.

Let us first bound $f(S_{-i}^*)$. Note that

$$\sum_{j \in A \setminus \{i\}} \alpha_j^* = \sum_{j \in A} \alpha_j^* - \alpha_i^* < B - (3/4) \cdot B = (1/4) \cdot B. \quad (1)$$

By applying the doubling lemma (Lemma 2.2) to $\alpha^*|_{-i}$ with $\varepsilon = (1/4) \cdot (B/n)$, we obtain a contract $\alpha' = 2\alpha^*|_{-i} + \varepsilon$, where $\varepsilon = (\varepsilon, \dots, \varepsilon)$, that satisfies the following properties:

- (i) α' is budget-feasible, since $\sum_{j \in A} \alpha'_j = 2 \sum_{j \in A \setminus \{i\}} \alpha_j^* + n\varepsilon \leq (1/2) \cdot B + (1/4) \cdot B < B$, where the first inequality follows from Inequality (1).
- (ii) The payment to every agent $j \in A \setminus \{i\}$ is $\alpha'_j = 2 \cdot \alpha_j^* + \varepsilon \leq (1/2) \cdot B + (1/4) \cdot B \leq (3/4) \cdot B$.
- (iii) The payment to agent i is $\alpha'_i = \varepsilon < (3/4) \cdot B$.
- (iv) Any equilibrium $S' \in \text{NE}(\alpha')$ satisfies $f(S') \geq (1/2) \cdot f(S_{-i}^*)$, by Lemma 2.2.

Thus, taking any $S' \in \text{NE}(\alpha')$, we get

$$\begin{aligned} f(S_{-i}^*) &\leq 2 \cdot f(S') && \text{(by (iv))} \\ &\leq 2 \cdot \text{MAX-REWARD-BOUNDED}(B) && \text{(by (i)-(iii)).} \end{aligned}$$

Next, let us bound $\varphi(\alpha^*|_i, S_i^*)$. Observe that, from Corollary 4.4 it holds that there exists an equilibrium S' of the contract $\alpha^*|_i$ such that $S' \subseteq T_i$ and $S_i^* \subseteq S'$. Thus,

$$\varphi(\alpha^*|_i, S_i^*) \leq \varphi(\alpha^*|_i, S') \leq \text{BEST-SINGLE}_i\text{-}\varphi(B),$$

where the first inequality is by Definition 2.4iii. This concludes the proof. $\blacktriangleleft$

Our reductions will take the better contract between the one achieved by (approximately) solving the problem we reduce to, $\text{MAX-REWARD-BOUNDED}(B)$, and the best single-agent contract. In the binary-actions case, the best single-agent contract for agent $i \in A$ is simply the ratio between their cost and the success probability when only i exerts effort. In the combinatorial-actions case, solving $\text{BEST-SINGLE}_i\text{-}\varphi(B)$ is not as straightforward, but the following lemma shows we can still do so in polynomial time when f is gross substitutes.

► **Lemma 4.9.** *Fix some objective φ and budget $B \in (0, 1]$. When f is gross substitutes, there exists a poly-time algorithm which (exactly) solves $\text{BEST-SINGLE}_i\text{-}\varphi(B)$ with value oracle access to f .*

The proof of Lemma 4.9 relies on the “critical point” analysis of [26], who proved the lemma for the special case of maximizing profit for $B = 1$, and is deferred to the full version. We now have all the building blocks for our reductions. We begin by proving a reduction from $\text{MAX-}\varphi(B)$ to $\text{MAX-REWARD-BOUNDED}(B)$.

► **Lemma 4.10** (Reduction to $\text{MAX-REWARD-BOUNDED}(B)$). *Fix an instance of the problem, $\langle A, \{T_i\}_{i \in A}, f, c \rangle$, with gross substitutes f , a budget $B \in (0, 1]$, and a BEST objective φ . For any $(\alpha, S) \in \mathcal{C}(B)$ that is a γ -approximation to $\text{MAX-REWARD-BOUNDED}(B)$, let (α', S') be the result of applying the downsizing lemma (Lemma 4.5) to (α, S) with $M = 6$. Then, it holds that one of $\{\text{BEST-SINGLE}_i\text{-}\varphi(B)\}_{i \in A} \cup \{(\alpha', S')\}$ is a $(120\gamma + 1)$ -approximation to $\text{MAX-}\varphi(B)$.*

Proof. Let $(\alpha, S) \in \mathcal{C}(B)$ be a γ -approximation to $\text{MAX-REWARD-BOUNDED}(B)$ and let (α', S') be the result of applying Lemma 4.5 to (α, S) with $M = 6$. This yields a contract-equilibrium pair $(\alpha', S') \in \mathcal{C}(B)$ such that $f(S') \geq (1/10) \cdot f(S)$ and either $\alpha' = \alpha|_i$ for some $i \in A$ or $\sum_{i \in A} \alpha'_i \leq (5/6) \cdot \sum_{i \in A} \alpha_i \leq (5/6) \cdot B \leq 5/6$. In the case where there exists an agent $i \in A$ such that $\alpha' = \alpha|_i$, we have $\sum_{j \in A} \alpha'_j = \alpha_i \leq (3/4) \cdot B \leq 3/4$, where the first inequality is because (α, S) satisfies the constraints of $\text{MAX-REWARD-BOUNDED}(B)$. Therefore, it holds in both cases that $\sum_{j \in A} \alpha'_j \leq 5/6$. Now, it follows that:

$$\begin{aligned} \varphi(\alpha', S') &\geq u_P(\alpha', S') && \text{(since } \varphi \text{ is a BEST objective)} \\ &= \left(1 - \sum_{i \in A} \alpha'_i\right) f(S') && \text{(by the definition of } u_P) \\ &\geq (1/6) \cdot f(S') && \text{(since } \sum_{j \in A} \alpha'_j \leq 5/6) \\ &\geq (1/60) \cdot f(S) && \text{(since } f(S') \geq (1/10) \cdot f(S)) \\ &\geq (1/60\gamma) \cdot \text{MAX-REWARD-BOUNDED}(B) && \text{(since } S \text{ is } \gamma\text{-approximation)} \end{aligned}$$

Let $V = \max\{\{\text{BEST-SINGLE}_i\text{-}\varphi(B)\}_{i \in A} \cup \{\varphi(\alpha', S')\}\}$. By Lemma 4.8, we get:

$$\begin{aligned} \text{MAX-}\varphi(B) &\leq 2 \cdot \text{MAX-REWARD-BOUNDED}(B) + \max_{i \in A} \text{BEST-SINGLE}_i\text{-}\varphi(B) \\ &\leq (120\gamma) \cdot \varphi(\alpha', S') + \max_{i \in A} \text{BEST-SINGLE}_i\text{-}\varphi(B) \\ &\leq (120\gamma + 1) \cdot V. \end{aligned}$$

This concludes the proof. $\blacktriangleleft$

Next, we establish reduction from $\text{MAX-REWARD-BOUNDED}(B)$ to $\text{MAX-}\varphi(B)$.

► **Lemma 4.11** (Reduction from $\text{MAX-REWARD-BOUNDED}(B)$). *Fix an instance of the problem $\mathcal{I} = \langle A, \{T_i\}_{i \in A}, f, c \rangle$ with gross substitutes f and two budgets $B, B' \in (0, 1]$, and let $\mathcal{I}' = \langle A, \{T_i\}_{i \in A}, f, c' \rangle$, be an instance with scaled costs, $c' = c \cdot (4/3) \cdot (B'/B)$. For any BEST objective φ , if (α', S') is a γ -approximation to $\text{MAX-}\varphi(B')$ in instance $\mathcal{I}'$, then one of $\{\text{BEST-SINGLE}_i\text{-}f(B)\}_{i \in A} \cup \{(\alpha' \cdot (3/4) \cdot (B/B'), S')\}$ is a 50γ -approximation to $\text{MAX-REWARD-BOUNDED}(B)$ in instance $\mathcal{I}$.*

Proof. Let $(\alpha^*, S^*) \in \mathcal{C}(B)$ be a solution to $\text{MAX-REWARD-BOUNDED}(B)$ in the instance $\mathcal{I}$, and let $(\alpha, S) \in \mathcal{C}(B)$ be the result of applying the downsizing lemma (Lemma 4.5) on (α^*, S^*) with $M = 14$. By the guarantees of Lemma 4.5, $f(S) \geq (1/26) \cdot f(S^*)$ and either (i) $\alpha = \alpha|_i$ and $S \subseteq T_i$ for some $i \in A$, or (ii) $\sum_{i \in A} \alpha_i \leq (5/14) \cdot \sum_{i \in A} \alpha_i^* \leq (5/14) \cdot B$.

In case (i) holds, we get that $\text{BEST-SINGLE}_i\text{-}f(B)$ is a 26-approximation to the problem of $\text{MAX-REWARD-BOUNDED}(B)$, which concludes the proof.

Suppose that case (ii) holds. Recall that (α, S) is a contract-equilibrium pair in $\mathcal{I}$, and $\mathcal{I}'$ is the same as $\mathcal{I}$ except the costs are scaled by $(4/3) \cdot (B'/B)$. Thus, if we scale the agent payments by the same factor, namely $\alpha^\dagger = (4/3) \cdot (B'/B) \cdot \alpha$, we have that $(\alpha^\dagger, S)$ is a contract-equilibrium pair in $\mathcal{I}'$. Observe that:

$$\sum_{i \in A} \alpha_i^\dagger \leq (4/3) \cdot (B'/B) \cdot (5/14) \cdot B = (10/21) \cdot B' \leq \min(10/21, B').$$

Therefore, $\alpha^\dagger$ is budget-feasible with respect to B' , and the principal's profit from the contract-equilibrium pair $(\alpha^\dagger, S)$ is at least:

$$\begin{aligned} u_P(\alpha^\dagger, S) &\geq (1 - 10/21) \cdot f(S) = (11/21) \cdot f(S) \\ &\geq (11/21) \cdot (1/26) \cdot f(S^*) > (1/50) \cdot f(S^*). \end{aligned} \tag{2}$$

Let (α', S') be a γ -approximation to $\text{MAX-}\varphi(B')$ in $\mathcal{I}'$. Then it holds that:

$$\begin{aligned} f(S') &\geq \varphi_{\mathcal{I}'}(\alpha', S') && (\text{since } \varphi \text{ is BEST}) \\ &\geq \text{MAX-}\varphi(B')/\gamma && (\text{since } (\alpha', S') \text{ is a } \gamma\text{-approximation}) \\ &\geq \varphi(\alpha^\dagger, S)/\gamma && (\text{since } \alpha^\dagger \text{ is budget-feasible w.r.t. } B') \\ &\geq u_P(\alpha^\dagger, S)/\gamma && (\text{since } \varphi \text{ is BEST}) \\ &\geq (1/50\gamma) \cdot f(S^*) && (\text{by Equation (2)}) \end{aligned}$$

Let $\alpha^\dagger = (3/4) \cdot (B/B') \cdot \alpha'$. We show that the pair $(\alpha^\dagger, S')$ makes up a 50γ -approximation to $\text{MAX-REWARD-BOUNDED}(B)$. First, since (α', S') is a contract-equilibrium pair in $\mathcal{I}'$, $(\alpha^\dagger, S')$ make up such a pair with respect to $\mathcal{I}$. Since $f(S') \geq (1/50\gamma) \cdot f(S^*)$, it remains to show that the contract $\alpha^\dagger$ satisfies the feasibility constraints. Observe that $\alpha^\dagger$ is budget-feasible with respect to B , as $\sum_{i \in A} \alpha_i^\dagger = (3/4) \cdot (B/B') \cdot \sum_{i \in A} \alpha'_i \leq (3/4) \cdot B$, where the inequality follows from budget feasibility of α' with respect to B' . Similarly, for any $i \in A$, $\alpha'_i \leq B'$ and thus $\alpha_i^\dagger = (3/4) \cdot (B/B') \cdot \alpha'_i \leq (3/4) \cdot B$. This concludes the proof. $\blacktriangleleft$

Together, the two lemmas above imply Theorem 4.1.

5 FPTAS for Additive Instances

In this section, we consider instances with additive f . Specifically, we show that the FPTAS for multi-agent binary-action settings of [27], later generalized to budget constraints by [38], can be adapted to the combinatorial-actions setting.

► **Theorem 5.1.** *When f is additive, for each objective of maximizing profit, reward, and welfare, there exists a deterministic FPTAS under any budget $B \in [0, 1]$, using only value oracle access to f .*

At a high level, our proof for profit maximization relies on discretizing the function f and minimizing payments using the dynamic programming approach as in [27], while adapting the core ideas to the combinatorial-actions setting. For reward and welfare maximization, [38] noted that these problems can be reduced to the KNAPSACK problem. However, this is not the case in the combinatorial-actions setting, and so we use the ideas from profit maximization to solve those problems as well.

Fix an additive function $\varphi : 2^T \rightarrow [0, 1]$ with $\varphi(\emptyset) = 0$ and a real number $b \in [0, 1]$. We will later specify $\varphi(S)$ to be either the reward $f(S)$ or the welfare $f(S) - c(S)$. We begin by defining a discretization $\tilde{\varphi}$ of φ . Let $\delta = \epsilon/|T|$. Define $\tilde{\varphi}(S) = \sum_{a \in S} \lfloor \varphi(\{a\})/(\delta b) \rfloor (\delta b)$. Note that $\tilde{\varphi}(S)$ is a multiple of δb for every S .

For each $j \in \{0, \dots, n\}$ and $x \in \{0, \delta b, 2\delta b, \dots, \lceil |T|/\delta \rceil \delta b\}$, we define

$$A^{(\varphi)}(j, x) = \min_{S, \alpha} \left\{ \sum_{i \in A} \alpha_i \mid \tilde{\varphi}(S) \geq x, S \in \text{NE}(\alpha), S \subseteq T_1 \cup \dots \cup T_j \right\}.$$

This table can be computed in polynomial time via dynamic programming, as we show below.

► **Lemma 5.2.** *The table $A^{(\varphi)}$ can be computed in polynomial time in $|T|$ and ϵ .*

Proof. Observe that for $j = 0$, we have $A(0, 0) = 0$ and $A(0, x) = \infty$ for all $x > 0$.

Let us now fix $j > 0$. Let $T_j = \{a_1, \dots, a_k\}$ and assume without loss of generality that $c_{a_1}/f(\{a_1\}) \leq c_{a_2}/f(\{a_2\}) \leq \dots \leq c_{a_k}/f(\{a_k\})$. Note that for a given payment α_i , agent i 's best response S_i for S_{-i} belongs to:

$$\begin{aligned} \arg \max_{S_i \subseteq T_i} \{ \alpha_i \cdot f(S_i \sqcup S_{-i}) - c(S_i) \} &= \arg \max_{S_i \subseteq T_i} \{ \alpha_i \cdot f(S_i) + \alpha_i \cdot f(S_{-i}) - c(S_i) \} \\ &= \arg \max_{S_i \subseteq T_i} \{ \alpha_i \cdot f(S_i) - c(S_i) \} \\ &= \arg \max_{S_i \subseteq T_i} \left\{ \sum_{a \in S_i} (\alpha_i \cdot f(\{a\}) - c_a) \right\} \end{aligned}$$

Therefore, the agent's best response includes all actions such that $c_a \leq \alpha_i \cdot f(\{a\})$, or equivalently $c_a/f(\{a\}) \leq \alpha_i$. In particular, any contract incentivizes a prefix of $\{a_1, \dots, a_k\}$, and hence:

$$A^{(\varphi)}(j, x) = \min_{\ell \in \{0, 1, \dots, k\}} \left\{ A(j-1, x - \tilde{\varphi}(\{a_1, \dots, a_\ell\})) + \frac{c_{a_\ell}}{f(\{a_\ell\})} \right\},$$

where we treat the term $c_{a_\ell}/f(\{a_\ell\})$ as 0 when $\ell = 0$. This completes the proof. ◀

We are now ready to prove the main theorem. Below we present the proof for profit maximization, and we defer the proofs for reward and welfare maximization to the full version.

Proof of Theorem 5.1 for Profit Maximization. We prove the existence of an FPTAS for profit maximization under budget B . First, let (α^*, S^*) be the optimal contract-equilibrium pair under budget B . Let $b = \max_{a \in S^*} f(\{a\})$. Note that the algorithm does not have access to b , as it does not have access to the optimal solution, but there are polynomially many candidate values, and we can iterate over all of them. From now on, we assume that we know the value of b .

Note that by definition $A^{(f)}(n, x)$ is increasing in x . Let:

$$\bar{x} = \max_{x \in \{0, \delta b, 2\delta b, \dots, \lceil |T|/\delta \rceil \delta b\}} \left\{ x \mid A^{(f)}(n, x) \leq B \right\} \text{ and } x' \in \arg \max_{x \in \{0, \delta b, \dots, \bar{x}\}} (1 - A^{(f)}(n, x)) \cdot x.$$

Let (α', S') be the contract-equilibrium pair (with respect to the original f) that minimizes the sum of payments in the definition of $A^{(f)}(n, x')$. We will argue that (α', S') yields a $(1 - \epsilon)$ -approximation to the optimal profit, which will imply the theorem.

We begin by observing that $\tilde{f}(S^*) \leq \lceil |T|/\delta \rceil \delta b$ and that $\tilde{f}(S^*) \leq \bar{x}$. For the first inequality, note that $\tilde{f}(S^*) = \sum_{a \in S^*} \tilde{f}(\{a\}) \leq |S^*| \cdot b \leq \lceil |T|/\delta \rceil \delta b$. For the second inequality, we have $\sum_{i \in A} \alpha_i^* \leq B$ by the budget-feasibility of the optimal contract. Therefore, by the choice of x' , we have:

$$(1 - A^{(f)}(n, \tilde{f}(S^*))) \cdot \tilde{f}(S^*) \leq (1 - A^{(f)}(n, x')) \cdot x'. \quad (3)$$

Moreover, we observe that:

$$\tilde{f}(S^*) \geq \sum_{a \in S^*} f(\{a\}) - |S^*| \cdot \delta b \geq \sum_{a \in S^*} f(\{a\}) - \epsilon \cdot \max_{a \in S^*} f(\{a\}) \geq (1 - \epsilon) \cdot f(S^*). \quad (4)$$

It follows that:

$$\begin{aligned} (1 - \epsilon) \cdot u_P(\alpha^*, S^*) &= (1 - \epsilon) \cdot \left(1 - \sum_{i \in A} \alpha_i^* \right) \cdot f(S^*) && \text{(by the definition of } u_P) \\ &\leq \left(1 - \sum_{i \in A} \alpha_i^* \right) \cdot \tilde{f}(S^*) && \text{(by Equation (4))} \\ &\leq (1 - A(n, \tilde{f}(S^*))) \cdot \tilde{f}(S^*) && \text{(by the definition of } A) \\ &\leq (1 - A(n, x')) \cdot x' && \text{(by Equation (3))} \\ &\leq \left(1 - \sum_{i \in A} \alpha_i' \right) \cdot \tilde{f}(S') && \text{(by definition of } (\alpha', S')) \\ &\leq \left(1 - \sum_{i \in A} \alpha_i' \right) \cdot f(S') && \text{(since } \tilde{f}(S) \leq f(S) \text{ for all } S) \\ &= u_P(\alpha', S') && \text{(by the definition of } u_P) \end{aligned}$$

Therefore, our algorithm returns a $(1 - \epsilon)$ -approximation, which completes the proof. $\blacktriangleleft$

References

- 1 Gil Aharoni, Martin Hoefer, and Inbal Talgam-Cohen. Welfare and beyond in multi-agent contracts. In *EC 2025*, 2025.
- 2 Tal Alon, Matteo Castiglioni, Junjie Chen, Tomer Ezra, Yingkai Li, and Inbal Talgam-Cohen. Multi-project contracts. In *Proceedings of the 26th ACM Conference on Economics and Computation*, pages 580–598, 2025. doi:10.1145/3736252.3742600.

- 3 Tal Alon, Paul Duetting, Yingkai Li, and Inbal Talgam-Cohen. Bayesian analysis of linear contracts. In *EC 2023*, page 66, 2023. doi:10.1145/3580507.3597795.
- 4 Tal Alon, Paul Dütting, and Inbal Talgam-Cohen. Contracts with private cost per unit-of-effort. In *EC 2021*, pages 52–69, 2021. doi:10.1145/3465456.3467651.
- 5 Moshe Babaioff, Michal Feldman, and Noam Nisan. Combinatorial agency. In *EC 2006*, pages 18–28, 2006. doi:10.1145/1134707.1134710.
- 6 Moshe Babaioff, Michal Feldman, and Noam Nisan. Mixed strategies in combinatorial agency. In *WINE 2006*, pages 353–364, 2006. doi:10.1007/11944874_32.
- 7 Moshe Babaioff, Michal Feldman, and Noam Nisan. Free-riding and free-labor in combinatorial agency. In *SAGT 2009*, pages 109–121, 2009. doi:10.1007/978-3-642-04645-2_11.
- 8 Moshe Babaioff, Michal Feldman, Noam Nisan, and Eyal Winter. Combinatorial agency. *J. Econ. Theory*, 147(3):999–1034, 2012. doi:10.1016/J.JET.2012.01.010.
- 9 Moshe Babaioff and Eyal Winter. Contract complexity. *EC*, 14:911, 2014. doi:10.1145/2600057.2602852.
- 10 Francesco Bacchiocchi, Matteo Castiglioni, Alberto Marchesi, and Nicola Gatti. Learning optimal contracts: How to exploit small action spaces. In *ICLR 2024*, 2024.
- 11 Shai Bernstein and Eyal Winter. Contracting with heterogeneous externalities. *American Economic Journal: Microeconomics*, 4(2):50–76, 2012.
- 12 Liad Blumrosen and Noam Nisan. On the computational power of iterative auctions i: demand queries. Technical report, Discussion paper, 2005.
- 13 Federico Cacciamani, Martino Bernasconi, Matteo Castiglioni, and Nicola Gatti. Multi-agent contract design beyond binary actions. In *EC 2024*, page 1293, 2024. doi:10.1145/3670865.3673633.
- 14 Gabriel Carroll. Robustness and linear contracts. *Am. Econ. Rev.*, 105(2):536–563, 2015.
- 15 Matteo Castiglioni, Junjie Chen, Minming Li, Haifeng Xu, and Song Zuo. A reduction from multi-parameter to single-parameter bayesian contract design. In *SODA 2025*, pages 1795–1836, 2025. doi:10.1137/1.9781611978322.56.
- 16 Matteo Castiglioni, Junjie Chen, and Yingkai Li. Fair contracts. *arXiv preprint arXiv:2507.11214*, 2025. doi:10.48550/arXiv.2507.11214.
- 17 Matteo Castiglioni, Alberto Marchesi, and Nicola Gatti. Bayesian agency: Linear versus tractable contracts. In *EC 2021*, pages 285–286, 2021. doi:10.1145/3465456.3467602.
- 18 Matteo Castiglioni, Alberto Marchesi, and Nicola Gatti. Designing menus of contracts efficiently: the power of randomization. In *EC 2022*, pages 705–735, 2022. doi:10.1145/3490486.3538270.
- 19 Matteo Castiglioni, Alberto Marchesi, and Nicola Gatti. Multi-agent contract design: How to commission multiple agents with individual outcomes. In *EC 2023*, pages 412–448, 2023. doi:10.1145/3580507.3597793.
- 20 Yurong Chen, Zhaohua Chen, Xiaotie Deng, and Zhiyi Huang. Are bounded contracts learnable and approximately optimal? In *EC 2024*, pages 315–344, 2024. doi:10.1145/3670865.3673483.
- 21 Alon Cohen, Argyrios Deligkas, and Moran Koren. Learning approximately optimal contracts. *Theoretical Computer Science*, 980:114219, 2023. doi:10.1016/J.TCS.2023.114219.
- 22 Ramiro Deo-Campo Vuong, Shaddin Dughmi, Neel Patel, and Aditya Prasad. On supermodular contracts and dense subgraphs. In *SODA 2024*, pages 109–132, 2024. doi:10.1137/1.9781611977912.6.
- 23 Ilan Doron-Arad, Hadas Shachnai, Gilad Shmerler, and Inbal Talgam-Cohen. An algorithm-to-contract framework without demand queries. *arXiv preprint arXiv:2507.20038*, 2025. doi:10.48550/arXiv.2507.20038.
- 24 Paul Duetting, Michal Feldman, Tomasz Ponitka, and Ermis Soumalias. The pseudo-dimension of contracts. In *EC 2025*, 2025.
- 25 Paul Duetting, Michal Feldman, and Yarden Rashti. Succinct ambiguous contracts. *arXiv preprint arXiv:2503.02592*, 2025. doi:10.48550/arXiv.2503.02592.

- 26 Paul Dütting, Tomer Ezra, Michal Feldman, and Thomas Kesselheim. Combinatorial contracts. In *FOCS 2021*, pages 815–826, 2021. doi:10.1109/FOCS52979.2021.00084.
- 27 Paul Dütting, Tomer Ezra, Michal Feldman, and Thomas Kesselheim. Multi-agent contracts. In *STOC 2023*, pages 1311–1324, 2023. doi:10.1145/3564246.3585193.
- 28 Paul Dütting, Tomer Ezra, Michal Feldman, and Thomas Kesselheim. Multi-agent combinatorial contracts. In *SODA 2025*, pages 1857–1891, 2025. doi:10.1137/1.9781611978322.58.
- 29 Paul Dütting, Michal Feldman, and Yoav Gal-Tzur. Combinatorial contracts beyond gross substitutes. In *SODA 2024*, pages 92–108, 2024. doi:10.1137/1.9781611977912.5.
- 30 Paul Dütting, Michal Feldman, Yoav Gal-Tzur, and Aviad Rubinstein. When contracts get complex: Information-theoretic barriers. In *SODA 2026*, 2026.
- 31 Paul Dütting, Michal Feldman, Daniel Peretz, and Larry Samuelson. Ambiguous contracts. *Econometrica*, 92(6):1967–1992, 2024.
- 32 Paul Dütting, Michal Feldman, and Inbal Talgam-Cohen. Algorithmic contract theory: A survey. *Found. Trends Theor. Comput. Sci.*, 16(3-4):211–412, 2024. doi:10.1561/0400000113.
- 33 Paul Dütting, Tim Roughgarden, and Inbal Talgam-Cohen. Simple versus optimal contracts. In *EC 2019*, pages 369–387, 2019. doi:10.1145/3328526.3329591.
- 34 Tomer Ezra, Michal Feldman, and Maya Schlesinger. On the (in)approximability of combinatorial contracts. In *ITCS*, volume 287 of *LIPIcs*, pages 44:1–44:22. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPIcs.ITCS.2024.44.
- 35 Tomer Ezra, Michal Feldman, and Maya Schlesinger. Contract design for sequential actions. In *SODA 2026*, 2026.
- 36 Tomer Ezra, Stefano Leonardi, and Matteo Russo. Contracts with inspections. In *SODA 2026*, 2026.
- 37 Michal Feldman. Combinatorial contract design: Recent progress and emerging frontiers. *arXiv preprint arXiv:2510.15065*, 2025. doi:10.48550/arXiv.2510.15065.
- 38 Michal Feldman, Yoav Gal-Tzur, Tomasz Ponitka, and Maya Schlesinger. Budget-feasible contracts. In *EC 2025*, pages 353–353, 2025.
- 39 Michal Feldman and Liat Yashin. Ultra-efficient contracts: Breaking the substitutes barrier in combinatorial contracts. *arXiv preprint arXiv:2506.18008*, 2025. doi:10.48550/arXiv.2506.18008.
- 40 Sanford J Grossman and Oliver D Hart. *An analysis of the principal-agent problem*. Springer, 1992.
- 41 Faruk Gul and Ennio Stacchetti. Walrasian equilibrium with gross substitutes. *Journal of Economic theory*, 87(1):95–124, 1999.
- 42 Guru Guruganesh, Jon Schneider, and Joshua R Wang. Contracts under moral hazard and adverse selection. In *EC 2021*, pages 563–582, 2021. doi:10.1145/3465456.3467637.
- 43 Guru Guruganesh, Jon Schneider, Joshua R. Wang, and Junyao Zhao. The power of menus in contract design. In *EC 2023*, pages 818–848, 2023. doi:10.1145/3580507.3597735.
- 44 Wade Hann-Caruthers and Sumit Goel. Optimality of weighted contracts for multi-agent contract design with a budget. In *EC 2024*, page 1295, 2024. doi:10.1145/3670865.3673598.
- 45 Chien-Ju Ho, Aleksandrs Slivkins, and Jennifer Wortman Vaughan. Adaptive contract design for crowdsourcing markets: Bandit algorithms for repeated principal-agent problems. In *EC 2014*, pages 359–376, 2014. doi:10.1145/2600057.2602880.
- 46 Bengt Holmström. Moral hazard and observability. *The Bell Journal of Economics*, pages 74–91, 1979.
- 47 Bengt Holmstrom. Moral hazard in teams. *The Bell Journal of Economics*, 13(2):324–340, 1982.
- 48 Robert D Innes. Limited liability and incentive contracting with ex-ante action choices. *Journal of economic theory*, 52(1):45–67, 1990.
- 49 Kelso Jr, Alexander S and Crawford, Vincent P. Job matching, coalition formation, and gross substitutes. *Econometrica: Journal of the Econometric Society*, pages 1483–1504, 1982.

- 50 Benny Lehmann, Daniel Lehmann, and Noam Nisan. Combinatorial auctions with decreasing marginal utilities. *Games Econ. Behav.*, 55(2):270–296, 2006. doi:10.1016/J.GEB.2005.02.006.
- 51 Renato Paes Leme. Gross substitutability: An algorithmic survey. *Games and Economic Behavior*, 106:294–316, 2017. doi:10.1016/J.GEB.2017.10.016.
- 52 Bo Peng and Zhihao Gavin Tang. Optimal robust contract design. In *Proceedings of the 25th ACM Conference on Economics and Computation*, pages 1294–1294, 2024.
- 53 Stephen A Ross. The economic theory of agency: The principal’s problem. *The American economic review*, 63(2):134–139, 1973.
- 54 Royal Swedish Academy of Sciences. Scientific background on the 2016 nobel prize in economic, 2016.
- 55 Ilya Segal. Contracting with externalities. *The Quarterly Journal of Economics*, 114(2):337–388, 1999.
- 56 Ilya Segal. Coordination and discrimination in contracting with externalities: Divide and conquer? *Journal of Economic Theory*, 113(2):147–181, 2003. doi:10.1016/S0022-0531(03)00114-5.
- 57 Banghua Zhu, Stephen Bates, Zhuoran Yang, Yixin Wang, Jiantao Jiao, and Michael I. Jordan. The sample complexity of online contract design. In *EC 2023*, page 1188, 2023. doi:10.1145/3580507.3597673.