The Complexity of Contracts [*]

Paul Dütting[†]   Tim Roughgarden[‡]   Inbal-Talgam Cohen[§]

Abstract

We initiate the study of computing (near-)optimal contracts in succinctly representable principal-agent settings. Here optimality means maximizing the principal's expected payoff over all incentive-compatible contracts—known in economics as "second-best" solutions. We also study a natural relaxation to *approximately* incentive-compatible contracts.

We focus on principal-agent settings with succinctly described (and exponentially large) outcome spaces. We show that the computational complexity of computing a near-optimal contract depends fundamentally on the number of agent actions. For settings with a constant number of actions, we present a fully polynomial-time approximation scheme (FPTAS) for the separation oracle of the dual of the problem of minimizing the principal's payment to the agent, and use this subroutine to efficiently compute a δ-incentive-compatible (δ-IC) contract whose expected payoff matches or surpasses that of the optimal IC contract.

With an arbitrary number of actions, we prove that the problem is hard to approximate within any constant c. This inapproximability result holds even for δ-IC contracts where δ is a sufficiently rapidly-decaying function of c. On the positive side, we show that simple linear δ-IC contracts with constant δ are sufficient to achieve a constant-factor approximation of the "first-best" (full-welfare-extracting) solution, and that such a contract can be computed in polynomial time.

1 Introduction

Economic theory distinguishes three fundamentally different problems involving asymmetric information and incentives. In the first—known as *mechanism design* (or *screening*)—the less informed party has to make a decision. A canonical example is Myerson's optimal auction design problem [Myerson, 1981], in which a seller wants to maximize the revenue from selling an item, having only incomplete information about the buyers' willingness to pay. The second problem is known as *signalling* (or *Bayesian persuasion*). Here, as in the first case, information is hidden, but this time the more informed party is the active party. A canonical example is Akerlof's "market for lemons" [Akerlof, 1970]. In this example, sellers are better informed about the quality of the products they sell, and may benefit by sharing (some) of their information with the buyers.

Both of these basic incentive problems have been studied very successfully and extensively from a computational perspective, see, e.g., [Cai et al., 2012a,b, 2013; Babaioff et al., 2014; Cai et al., 2016; Babaioff et al., 2017; Gonczarowski, 2018; Gonczarowski and Weinberg, 2018] and [Dughmi, 2014; Dughmi et al., 2014; Cheng et al., 2015; Dughmi and Xu, 2016].

The third basic problem, *the agency problem* in *contract theory*, has received far less attention from the theoretical computer science community, despite being regarded as equally important in economic theory (see, e.g., the scientific background on the 2016 Nobel Prize for Hart and Holmström [Royal Swedish Academy of Sciences, 2016]). (A notable exception is [Babaioff et al., 2012], which we will discuss with further related work in more detail below.)

The basic scenario of contract theory is captured by the following *hidden-action principal-agent problem* [Grossman and Hart, 1983]: There is one *principal* and one *agent*. The agent can take one of n actions $a_i \in A_n$. Each action a_i is associated with a distribution F_i over m outcomes $x_j \in \mathbb{R}_{\geq 0}$, and has a cost $c_i \in \mathbb{R}_{\geq 0}$. The principal designs a contract p that specifies a payment $p(x_j)$ for each outcome x_j. The agent chooses an action a_i that maximizes expected payment minus cost, i.e., $\sum_j F_{i,j} p(x_j) - c_i$. The principal seeks to set up the contract so that the chosen action maximizes expected outcome minus expected payment, i.e., $\sum_j F_{i,j} x_j - \sum_j F_{i,j} p(x_j)$.

The principal-agent problem is quite different from mechanism design and signalling, where the basic diffi-

culty is the information asymmetry and that part of the information is hidden. In the principal-agent problem the issue is one of *moral hazard*: in and by itself the agent has no intrinsic interest in the expected outcome to the principal.

It is straightforward to see that the optimal contract can be found in time polynomial in n and m by solving n linear programs (LPs). For each action the corresponding LP gives the smallest expected payment at which this action can be implemented. The action that yields the highest expected reward minus payment gives the optimal payoff to the principal, and the LP for this action the optimal contract.

Succinct principal-agent problems. This linear programming-based algorithm for computing an optimal contract has several analogs in algorithmic game theory:

1. *Mechanism design.* For many basic mechanism design problems, the optimal (randomized) mechanism is the solution of a linear program with size polynomial in that of the players' joint type space.
2. *Signalling.* For many computational problems in signalling, the optimal signalling scheme is the solution to a linear program with size polynomial in the number of receiver actions and possible states of nature.
3. *Correlated equilibria.* In finite games, a correlated equilibrium can be computed using a linear program with size polynomial in the number of game outcomes.

These linear-programming-based solutions are unsatisfactory when their size is exponential in some parameter of interest. For example, in the mechanism design and correlated equilibria examples, the size of the LP is exponential in the number of players. A major contribution of theoretical computer science to game theory and economics has been the articulation of natural classes of succinctly representable settings and a thorough study of the computational complexity of optimal design problems in such settings. Examples include work on multidimensional mechanism design that has emphasized succinct type distributions [Cai et al., 2012a,b, 2013, 2016], succinct signalling schemes with an exponential number of states of nature [Dughmi and Xu, 2016], and the efficient computation of correlated equilibria in succinctly representable multi-player games [Papadimitriou and Roughgarden, 2008; Jiang and Leyton-Brown, 2015]. The goal of this paper is to initiate an analogous line of work for succinctly described agency problems in contract theory.

We focus on principal-agent settings with succinctly described (and exponentially large) outcome spaces, along with a reward function that supports value queries

and a distribution for each action with polynomial description. While there are many such settings one can study, we focus on what is arguably the most natural one from a theoretical computer science perspective, where outcomes correspond to vertices of the hypercube, the reward function is additive, and the distributions are product distributions. (Cf., work on computing revenue-maximizing multi-item auctions with product distributions over additive valutions, e.g. [Cai et al., 2012a,b].)

For example, outcomes could correspond to sets of items, where items are sold separately using posted prices. Actions could correspond to different marketing strategies (with different costs), which lead to different (independent) probabilities of sales of various items. Or, imagine that a firm (principal) uses a headhunter (agent) to hire an employee (action). Dimensions could correspond to tasks or skills. Actions correspond to types of employees, costs correspond to the difficulty of recruiting an employee of a given type, and for each employee type there is some likelihood that they will possess each skill (or be able to complete some task). The firm wants to motivate the headhunter to put in enough effort to recruit an employee who is likely to have useful skills for the firm, without actually running extensive interviews to find out the employee's type.

In our model, as in the classic model, there is a *principal* and an *agent*. The agent can take one of n actions $a_i \in A_n$, and each action has a cost $c_i \in \mathbb{R}_{\geq 0}$. Unlike in the original model, we are given a set of items M, with $|M| = m$. Outcomes correspond to subsets of items $S \in 2^M$. Each item has a reward r_j, and the reward of a set S of items is $\sum_{j \in S} r_j$. Every action a_i comes with probabilities $q_{i,j}$ for each item j. If action a_i is chosen, each item j is included in the outcome independently with probability $q_{i,j}$. A contract specifies a payment p_S for each outcome $S \in 2^M$. The goal is to compute a contract that maximizes (perhaps approximately) the principal's payoff in running time polynomial in n and m (which is logarithmic in the size $|2^M|$ of the outcome space). Note that if we wish to describe the contract by its nonzero payments, the running time requirement forces us to use only contracts with polynomial description length, that is, zero payments for all but polynomially-many $S \in 2^M$.

A notion of approximate IC for contracts. The classic approach in contract theory is to require that the agent is incentivized exactly, i.e., he (weakly) prefers the chosen action over every other action. We refer to such contracts as incentive compatible or just IC contracts. Motivated in part by our hardness results for IC contracts (see the next section) and inspired by the success of notions of approximate incentive

compatibility in mechanism design (see [Cai, 2013; Weinberg, 2014; Cai et al., 2016], hereafter referred to as the *CDW framework*), we introduce a notion of approximate incentive compatibility that is suitable for contracts.

Our notion of δ-incentive compatibility (or δ-IC) is that the agent utility of the approximately incentivized action a_i is at least that of any other action $a_{i'}$, less δ. (See Section 2 and full version for details and discussion.) This notion is natural for several reasons. First, it coincides with the usual notion of ϵ-IC in "normalized" mechanism design settings (with all valuations between 0 and 1), as in [Cai, 2013; Weinberg, 2014]. A second reason is behaviorial. There is an increasing body of work in economics on behavioral biases in contract theory [Koszegi, 2014], including strong empirical evidence that such biases play an important role in practice—for example, that agents "gift" effort to the principals employing them [Akerlof, 1982]. The notion of δ-IC offers a mathematical formulation of an agent's bias. Along similar lines, Carroll [2013] advocates generally for approximate IC constraints in settings where the designer can propose their "preferred action" to agents, in which case an agent may be biased against deviating due to the complexities involved in determining the agent-optimal action or the psychological costs of deviating. See also [Englmaier and Leider, 2012] for related discussion in the context of contract theory.

1.1 Our contribution and techniques We prove several positive and negative algorithmic results for computing near-optimal contracts in succinctly described principal-agent settings. Our work reveals a fundamental dichotomy between settings with a constant number of actions and those with an arbitrary number of actions.

Constant number of actions. For a constant number of actions, we prove in Section 3 that while it is NP-hard to compute an optimal IC contract, there is an FPTAS that computes a δ-IC contract with expected principal surplus at least that of the optimal IC contract; the running time is polynomial in m and $1/\delta$.

THEOREM (SEE THEOREM 3.1, COROLLARY 3.2). *For every constant $n \geq 1$ and $\delta > 0$, there is an algorithm that computes a δ-IC contract with expected principal surplus at least that of an optimal IC contract in time polynomial in m and $1/\delta$.*

The starting point of our algorithm is a linear programming formulation of the problem of incentivizing a given action with the lowest possible expected payment. Our formulation has a polynomial number of constraints (one per action other than the to-be-incentivized one) but an exponential number of variables (one per outcome). A natural idea is to then solve the dual linear program using the ellipsoid method. The dual separation oracle is: given a weighted mixture of $n-1$ product distributions (over the m items) and a reference product distribution q^*, minimize the ratio of the probability of outcome S in the mixture distribution and that in the reference distribution. Unfortunately, as we show, this is an NP-hard problem, even when there are only $n = 3$ actions. On the other hand, we provide an FPTAS for the separation oracle in the case of a constant number of actions, based on a delicate multi-dimensional bucketing approach. The standard method of translating an FPTAS for a separation oracle to an FPTAS for the corresponding linear program relies on a scale-invariance property that is absent in our problem. We proceed instead via a strengthened version of our dual linear program, to which our FPTAS separation oracle still applies, and show how to extract from an approximately optimal dual solution a δ-IC contract with objective function value at least that of the optimal solution to the original linear program.

Arbitrary number of actions. The restriction to a constant number of actions is essential for the positive results above (assuming $P \neq NP$). Specifically, we prove in Section 4 that computing the IC contract that maximizes the expected payoff to the principal is NP-hard, even to approximate to within any constant c. This hardness of approximation result persists even if we relax from exact IC to δ-IC contracts, provided δ is sufficiently small as a function of c.

THEOREM (SEE THEOREM 4.1, COROLLARY 4.2). *For every constant $c \in \mathbb{R}$, $c \geq 1$, it is NP-hard to find a IC contract that approximates the optimal expected payoff achievable by an IC contract to within a multiplicative factor of c.*

THEOREM (SEE THEOREM 4.1, COROLLARY 4.3). *For any constant $c \in \mathbb{R}, c \geq 5$ and $\delta \leq (\frac{1}{4c})^c$, it is NP-hard to find a δ-IC contract that guarantees $> \frac{2}{c}OPT$, where OPT is the optimal expected payoff achievable by an IC contract.*

We prove these hardness of approximation results by reduction from MAX-3SAT, using the fact that it is NP-hard to distinguish between a satisfiable MAX-3SAT instance and one in which there is no assignment satisfying more than a $7/8 + \alpha$ fraction of the clauses, where α is some arbitrarily small constant [Håstad, 2001]. Our reduction utilizes the gap between "first best" (full-welfare-extracting) and "second best" solutions in contract design settings, where satisfiable instances of MAX-3SAT map to instances where there

is no gap between first and second best and instances of MAX-3SAT in which no more than $7/8 + \alpha$ clauses can be satisfied map to instances where there is a constant-factor multiplicative gap between the first-best and second-best solutions.

On the positive side, we prove that for every constant δ there is a simple (in fact, linear[1]) contract that achieves a c_δ-approximation, where c_δ is a constant that depends on δ. This approximation guarantee is with respect to the strongest possible benchmark, the first-best solution.[2]

THEOREM (SEE THEOREM 5.1). *For every constant $\delta > 0$ there exists a constant c_δ and a polynomial-time (in n and m) computable δ-IC contract that obtains a multiplicative c_δ-approximation to the optimal welfare.*

Our proof of this result, in Section 5, shows that the optimal social welfare can be upper bounded by a sum of (constantly many in δ) expected payoffs achievable by δ-IC contracts. The best such contract thus obtains a constant approximation to the optimal welfare.

Black-box distributions. Product distributions are a rich and natural class of succinctly representable distributions to study, but one could also consider other classes. Perhaps the strongest-imaginable positive result would be an efficient algorithm for computing a near-optimal contract that works with *no* assumptions about each action's probability distribution over outcomes, other than the ability to sample from them efficiently. (Positive examples of this sort in signalling problem include Dughmi and Xu [2016] and in mechanism design include Hartline and Lucier [2015] and its many follow-ups.) Interestingly, the principal-agent problem poses unique challenges to such "black-box" positive results. The moral reason for this is explained, for example, in Salanié [2005]: Rewards play a dual role in contract settings, both defining the surplus from the joint project to be shared between the principal and agent *and* providing a signal to the principal of the agent's action. For this reason, in optimal contracts, the payment to the agent in a given outcome is governed both by the outcome's reward and on its "informativeness," and the latter is highly sensitive to the

precise probabilities in the outcome distributions associated with each action. In Section 6 we translate this intuition into an information-theoretic impossibility result for the black-box model, showing that positive results are possible only under strong assumptions on the distributions (e.g., that the minimum non-zero probability is bounded away from 0).

1.2 Related work The study of computational aspects of contract theory was pioneered by Babaioff, Feldman and Nisan [Babaioff et al., 2012] (see also their subsequent works, notably [Emek and Feldman, 2012] and [Babaioff and Winter, 2014]). This line of work studies a problem referred to as *combinatorial agency*, in which combinations of agents replace the single agent in the classic principal-agent model. The challenge in the new model stems from the need to incentivize multiple agents, while the action structure of each agent is kept simple (effort/no effort). The focus of this line of work is on complex combinations of agents' efforts influencing the outcomes, and how these determine the subsets of agents to contract with. The resulting computational problems are very different from the computational problems in our model.[3]

A second direction of highly related work is [Azar and Micali, 2018]. This work considers a principal-agent model in which the agent action space is exponentially sized but compactly represented, and argue that in such settings indirect (interactive) mechanisms can be better than one-shot mechanisms. Our focus is more algorithmic, and instead of a compactly represented action space we consider a compactly represented outcome space.

A third direction of related work considers a bandit-style model for contract design [Ho et al., 2016]. In their model each arm corresponds to a contract, and they present a procedure that starts out with a discretization of the contract space, which is adaptively refined, and which achieves sublinear regret in the time horizon. Again the result is quite different from our work, where the complexity comes from the compactly represented outcome space, and our result on the black-box model sheds a more negative light on the learning approach.

Further related work comes from Kleinberg and Kleinberg [2018] who consider the problem of delegating a task to an agent in a setting where (unlike in our model) monetary compensation is not an option. Although payments are not available, they show through an elegant reduction to the prophet-inequality problem that constant competitive solutions are possible.

---

[1] A linear contract is defined by a single parameter $\alpha \in [0, 1]$, and sets the payment p_S for any set $S \in 2^M$ to $p_S = \alpha \cdot \sum_{j \in S} r_j$. Linear contracts correspond to a simple percentage commission, and are arguably among the most frequently used contracts in practice. See [Carroll, 2015] and [Dütting et al., 2019] for recent work in economics and computer science in support of linear contracts.

[2] Note that the principal's objective function (reward minus payment to the agent) is a mixed-sign objective; such functions are generally challenging for relative approximation results.

[3] For example, several of the key computational questions in their problem turn out to be #P-hard, while all of the problems we consider are in NP.

A final related line of work was initiated by Carroll [2015] who—working in the classic model (where computational complexity is not an issue)—shows a sense in which linear contracts are max-min optimal (see also the recent work of [Walton and Carroll, 2019]). Dütting et al. [2019] show an alternative such sense, and also provide tight approximation guarantees for linear contracts.

2 Preliminaries

2.1 Succinct principal-agent settings Let n and m be parameters. A principal-agent setting is composed of the following: n *actions* A_n among which the agent can choose, and their *costs* $0 = c_1 \leq \cdots \leq c_n$ for the agent; *outcomes* which the actions can lead to, and their *rewards* for the principal; and a *mapping* from actions to distributions over outcomes. Crucially, the agent's choice of action is hidden from the principal, who observes only the action's realized outcome. Our goal is to study *succinct* principal-agent settings with description size polynomial in n and m; the (implicit) outcome space can have size exponential in m. Throughout, unless stated otherwise, all principal-agent settings we consider are succinct. We focus on arguably one of the most natural models of succinctly-described settings, namely those with *additive* rewards and *product* distributions.

In more detail, let $M = \{1, 2, .., m\}$, where M is referred to as the *item set*. Let the outcome space be $\{0,1\}^M$, that is, every outcome is an item subset $S \subseteq M$. For every item $j \in M$, the principal gets an additive reward r_j if the realized outcome includes j, so the principal's reward for outcome S is $r_S = \sum_{j \in S} r_j$. Every action $a_i \in A_n$ is associated with probabilities $q_{i,1}, ..., q_{i,m} \in [0,1]$ for the items. We denote the corresponding product distribution by q_i. When the agent takes action a_i, item j is included in the realized outcome independently with probability $q_{i,j}$. The probability of outcome S is thus $q_{i,S} = (\prod_{j \in S} q_{i,j})(\prod_{j \notin S}(1-q_{i,j}))$. By linearity of expectation, the principal's expected reward given action a_i is $R_i = \sum_S q_{i,S} r_S = \sum_j q_{i,j} r_j$. Action a_i's expected *welfare* is $R_i - c_i$, and we assume $R_i - c_i \geq 0$ for every $i \in [n]$.

EXAMPLE 2.1 (SUCCINCT PRINCIPAL-AGENT SETTING). *A company (principal) hires an agent to sell its m products. The agent may succeed in selling any subset of the m items, depending on his effort level, where the ith level leads to sale of item j with probability $q_{i,j}$. Reward r_j from selling item j is the profit-margin of product j for the company.*

Representation. A succinct principal-agent setting is described by an n-vector of costs c, an m-vector of rewards r, and an $n \times m$-matrix Q where entry (i, j)

is equal to probability $q_{i,j}$ (and we assume for simplicity that the number of bits of precision for all values is poly(n, m)).

Assumptions. Unless stated otherwise, we assume that all principal-agent settings are *normalized*, i.e., $R_i \leq 1$ for every $a_i \in A_n$ (and thus also $c_i \leq 1$). Normalization amounts to a simple change of "currency", i.e., of the units in which rewards and costs are measured. It is a standard assumption in the context of approximate incentive compatibility—see Section 2.2 (similar assumptions appear in both the CDW framework and in [Carroll, 2013]). We also assume *no dominated actions*: every two actions $a_i, a_{i'}$ have distinct expected rewards $R_i \neq R_{i'}$, and $R_{i'} < R_i$ implies $c_{i'} < c_i$. That is, we assume away any action that simultaneously costs more for the agent and has lower expected reward for the principal than some (dominating) alternative action.

Contracts and incentives. A *contract* p is a vector of payments from the principal to the agent. Payments are non-negative; this is known as *limited liability* of the agent.[4] The contractual payments are contingent on the outcomes and not actions, as the actions are not directly observable by the principal. A contract p can potentially specify a payment $p_S \geq 0$ for every outcome S, but by linear programming (LP) considerations detailed below, we can focus on contracts for which the support size of the vector p is polynomial in n. We sometimes denote by p_i the expected payment $\sum_{S \subseteq M} q_{i,S} p_S$ to the agent for choosing action a_i, and without loss of generality restrict attention to contracts for which $p_i \leq R_i$ for every $a_i \in A_n$ (in particular, $p_i \leq 1$ by normalization).

Given contract p, the agent's expected *utility* from choosing action a_i is $p_i - c_i$. The principal's expected *payoff* is then $R_i - p_i$. The agent wishes to maximize his expected utility over all actions and over an outside option with utility normalized to zero ("individual rationality" or *IR*). Since by assumption the cost c_1 of action a_1 is 0, the outside opportunity is always dominated by action a_1 and so we can omit the outside option from consideration. Therefore, the incentive constraints for the agent to choose action a_i are: $p_i - c_i \geq p_{i'} - c_{i'}$ for every $i' \neq i$. If these constraints hold we say a_i is *incentive compatible (IC)* (and as discussed, in our model IC implies IR). The standard tie-breaking assumption in the contract design literature is that among several IC actions the agent tie-breaks in favor of the principal, i.e. chooses the IC action that maximizes the princi-

---

[4]Limited liability plays a similar role in the contract literature as risk-averseness of the agent. Both reflect the typical situation in which the principal has "deeper pockets" than the agent and is thus the better bearer of expenses/risks.

pal's expected payoff.[5] We say contract p *implements* or *incentivizes* action a_i if given p the agent chooses a_i (namely a_i is IC and survives tie-breaking). If there exists such a contract for action a_i we say a_i is *implementable*, and slightly abusing notation we sometimes refer to the implementing contract as an *IC contract*.

Simple contracts. In a *linear* contract, the payment scheme is a linear function of the rewards, i.e., $p_S = \alpha r_S$ for every outcome S. We refer to $\alpha \in [0,1]$ as the linear contract's *parameter*, and it serves as a succinct representation of the contract. Linear contracts have an alternative succinct representation by an m-vector of item payments $p_j = \alpha r_j$ for every $j \in M$, which induce additive payments $p_S = \sum_{j \in S} p_j$. A natural generalization is *separable* contracts, the payments of which can also be separated over the m items and represented by an m-vector of non-negative payments (not necessarily linear). It is straightforward to see that the optimal linear (resp., separable) contract can be found in polynomial time (see full version for details). We return to linear contracts in Section 5 and provide additional discussion of linear and separable contracts in the full version.

2.2 Contract design and relaxations The goal of contract design is to maximize the principal's expected payoff from the action chosen by the agent subject to IC constraints. A corresponding computational problem is OPT-CONTRACT: The input is a succinct principal-agent setting, and the output is the principal's expected payoff from the optimal IC contract. A related problem is MIN-PAYMENT: The input is a succinct principal-agent setting and an action a_i, and the output is the minimum expected payment p_i^* with which a_i can be implemented (up to tie-breaking). OPT-CONTRACT reduces to solving n instances of MIN-PAYMENT to find p_i^* for every action a_i, and returning the maximum expected payoff to the principal $\max_{i \in [n]}\{R_i - p_i^*\}$. Observe that MIN-PAYMENT can be formulated as an exponentially-sized LP with 2^m variables $\{p_S\}$ (one for each set $S \subseteq M$) and $n-1$ constraints:

$$(2.1) \quad \min \sum_{S \subseteq M} q_{i,S} p_S$$
$$\text{s.t.} \sum_{S \subseteq M} q_{i,S} p_S - c_i \geq$$
$$\sum_{S \subseteq M} q_{i',S} p_S - c_{i'} \quad \forall i' \neq i, i' \in [n],$$
$$p_S \geq 0 \qquad \forall S \subseteq M.$$

The dual LP has $n-1$ nonnegative variables $\{\lambda_{i'}\}$ (one for every action i' other than i), and exponentially-many constraints:

$$(2.2) \quad \max \sum_{i' \neq i} \lambda_{i'}(c_i - c_{i'})$$
$$\text{s.t.} \left(\sum_{i' \neq i} \lambda_{i'}\right) - 1 \leq$$
$$\sum_{i' \neq i} \lambda_{i'} \frac{q_{i',S}}{q_{i,S}} \quad \forall S \subseteq E, q_{i,S} > 0,$$
$$\lambda_{i'} \geq 0 \qquad \forall i' \neq i, i' \in [n].$$

Standard duality considerations imply that there always exists a succinct optimal contract with $n-1$ nonzero payments. However, the ellipsoid method cannot be applied to solve the dual LP in polynomial time because the required separation oracle turns out to be NP-hard except for $n = 2$.

PROPOSITION 2.1. *Solving the separation oracle of dual LP* (2.2) *is NP-hard for $n \geq 3$.*

A proof of this result can be found in the full version. We return to LP (2.1) and to its dual LP (2.2) in Section 3.

Relaxed IC. Contract design like auction design is ultimately an optimization problem subject to IC constraints. The state-of-the-art in optimal *auction* design requires a relaxation of IC constraints to ϵ-IC. In the CDW framework, the ϵ loss factor is additive and applies to normalized auction settings. The framework enables polytime computation of an ϵ-IC auction with expected revenue approximating that of the optimal IC auction.[6] Appropriate ϵ-IC relaxations are also studied in multiple additional contexts—see [Carroll, 2013] and references within for voting, matching and competitive equilibrium; and [Papadimitriou, 2006] for Nash equilibrium. We wish to achieve similar results in the context of optimal contracts. For completeness we include the definition of ϵ-IC cast in the language of contracts:

DEFINITION 2.2 (δ-IC ACTION). *Consider a (normalized) contract setting. For $\delta \geq 0$, an action a_i is δ-IC given a contract p if the agent loses no more than additive δ in expected utility by choosing a_i, i.e.: $p_i - c_i \geq p_{i'} - c_{i'} - \delta$ for every action $a_{i'} \neq a_i$.*

(Notation-wise, we will sometimes replace δ by Δ and refer to Δ-IC actions.) As in the IC case, we often slightly abuse notation and refer to the contract p

---

[5]The idea is that one could perturb the payment schedule slightly to make the desired action uniquely optimal for the agent. For further discussion see [Caillaud and Hermalin, 2000, p. 8].

[6]To be precise, the CDW framework focuses on *Bayesian* IC (BIC) and ϵ-BIC auctions.

itself as δ-IC. By this we mean a contract p with an (implicit) action a_i that is δ-IC given p (if there are several such δ-IC actions, by our tie-breaking assumption the agent chooses the one that maximizes the principal's expected payoff). We also say the contract δ-*implements* or δ-*incentivizes* action a_i. Finally if there exists such a contract for a_i then we say this action is δ-*implementable*. Interestingly, by LP duality, any action can be δ-implemented up to tie-breaking even for arbitrarily small δ (see full version for details). We denote by δ-OPT-CONTRACT and δ-MIN-PAYMENT the above computational problems with IC replaced by δ-IC (e.g., the input to δ-OPT-CONTRACT is a succinct principal-agent setting and a parameter δ, and the output is the principal's expected payoff from the optimal δ-IC contract).

In the full version we study the relation between optimal IC and δ-IC contracts in more detail. We in particular show that for every δ-IC contract there is an IC contract with approximately the same expected payoff to the principal up to small—and necessary—multiplicative and additive losses. Thus relaxing IC to δ-IC increases the expected payoff of the principal only to a certain extent. Similar results are known in the context of *auctions* (see [Hartline et al., 2015; Dughmi et al., 2017] for welfare maximization and [Daskalakis and Weinberg, 2012] for revenue maximization).[7]

PROPOSITION 2.3. *Fix a principal-agent setting and* $\delta > 0$. *Let p be a contract that δ-incentivizes action a_i. Denote by $\ell_{\alpha=1}$ the linear contract with parameter $\alpha = 1$ (that transfers the full reward from principal to agent). Then the IC contract p' defined as $(1 - \sqrt{\delta})p + \sqrt{\delta}\ell_{\alpha=1}$ achieves for the principal expected payoff of at least $(1 - \sqrt{\delta})(R_i - p_i) - (\sqrt{\delta} - \delta)$, where $R_i - p_i$ is the expected payoff of contract p.*

We also show that an additive loss is necessary, as even for tiny δ there can be a multiplicative constant-factor gap between the expected payoff of an IC contract and a δ-IC one.

Relaxed IC with exact IR. In our model, IC implies IR due to the existence of a zero-cost action a_1, but this is no longer the case for δ-IC. What if we are willing to relax IC to δ-IC due to the considerations above, but do not want to give up on IR? Suppose we enforce IR by assuming that the agent chooses a δ-IC action only if it has expected utility ≥ 0. The following lemma shows that this has only a small additive effect on the principal's expected payoff, allowing us from now on to focus on δ-IC contracts (IR can be later enforced by applying the lemma):

LEMMA 2.4. *For every δ-IC contract p that achieves expected payoff of Π for the principal, there exists a δ-IC and IR contract p' that achieves expected payoff of $\geq \Pi - \delta$.*

Proof. Fix a principal-agent setting. Let a_i be the action δ-incentivized by contract p and assume a_i is not IR. Observe that the agent's expected utility from a_i is $\geq -\delta$ (otherwise a_i would not be δ-IC with respect to a_1, which has expected utility ≥ 0 for the agent). First, if $\Pi > \delta$, then let p' be identical to p except for an additional δ payment for every outcome. Contract p' still δ-incentivizes action a_i, but now the agent's expected utility from a_i is ≥ 0, as required. Otherwise if $\Pi \leq \delta$, let p' be the contract with all-zero payments. The expected payoff to the principal is zero, which is at most an additive δ loss compared to Π. $\square$

3 Constant number of actions

In this section we begin our exploration of the computational problems OPT-CONTRACT and MIN-PAYMENT by considering principal-agent settings with a constant number n of actions. For every constant $n \geq 3$ these problems are NP-hard, and this holds even if the IC requirement is relaxed to δ-IC (see full version for details). As our main positive result, we establish the tractability of finding a δ-IC contract that matches the expected payoff of the optimal IC contract. In Section 4 we show this result is too strong to hold for non-constant values of n (under standard complexity assumptions), and in Section 5 we provide an approximation result for general settings.

To state our results more formally, fix a principal-agent setting and action a_i; let OPT_i be the solution to MIN-PAYMENT for a_i (or ∞ if a_i cannot be implemented up to tie-breaking without loss to the principal); and let OPT be the solution to OPT-CONTRACT. Observe that $OPT = \max_{i \in [n]}\{R_i - OPT_i\}$. Our main results in this section are the following:

THEOREM 3.1 (MIN-PAYMENT). *There exists an algorithm that receives as input a (succinct) principal-agent setting with a constant number of actions and m items, an action a_i, and a parameter $\delta > 0$, and returns in time $poly(m, \frac{1}{\delta})$ a contract that δ-incentivizes a_i with expected payment $\leq OPT_i$ to the agent.*

COROLLARY 3.2 (OPT-CONTRACT). *There exists an algorithm that receives as input a (succinct) principal-agent setting with a constant number of actions and m items, and a parameter $\delta > 0$, and returns in time $poly(m, \frac{1}{\delta})$ a δ-IC contract with expected payoff $\geq OPT$ to the principal.*

Proof. Apply the algorithm from Theorem 3.1 once per

action a_i to get a contract that δ-incentivizes a_i with expected payoff at least $R_i - OPT_i$ to the principal. Maximizing over the actions we get a δ-IC contract with expected payoff $\geq OPT$ to the principal. $\qquad\square$

Corollary 3.2 shows how to achieve OPT with a δ-IC contract rather than an IC one, in the same vein as the CDW results for auctions. A similar result does not hold for general n unless P=NP (Corollary 4.3). Note that the δ-IC contract can be transformed into an IR one with an additive δ loss by applying Lemma 2.4, and to a fully IC one with slightly more loss by Proposition 2.3, where δ can be an arbitrarily small inverse polynomial in m.

In the rest of the section we prove Theorem 3.1; omitted proofs appear in the full version.

An FPTAS for the separation oracle. We begin by stating the separation oracle problem. LP (2.1) formulates MIN-PAYMENT for action a_i. Its dual LP (2.2) has constraints of the form:

$$(3.3) \qquad (\sum_{i' \neq i} \lambda_{i'}) - 1 \leq \sum_{i' \neq i} \lambda_{i'} \frac{q_{i',S}}{q_{i,S}}.$$

We can rewrite (3.3) as $1 - 1/(\sum_{i' \neq i} \lambda_{i'}) \leq \frac{1}{q_{i,S}} \cdot \sum_{i' \neq i} (\lambda_{i'}/(\sum_{i' \neq i} \lambda_{i'})) \cdot q_{i',S}$. Thus the separation oracle problem for dual LP (2.2) is in fact the following problem: Let n be a constant and m a parameter. The input is $n - 1$ nonnegative weights $\{\alpha_{i'}\}$ that sum to 1; $n - 1$ product distributions $\{q_{i'}\}$; and a product distribution q_i; where all product distributions are over m items M. The goal is to minimize the likelihood ratio $\frac{\sum_{i'} \alpha_{i'} q_{i',S}}{q_{i,S}}$ over all outcomes $S \subseteq M$, where the numerator is the likelihood given S of the weighted combination distribution $\sum_{i'} \alpha_{i'} q_{i'}$, and the denominator is the likelihood given S of distribution q_i.

Denote the optimal solution (i.e. the minimum likelihood ratio) by ρ^*. Solving the separation oracle problem is NP-hard (Proposition 2.1),[8] but we show an FPTAS (Lemma 3.3). Lemma 3.4 gives the guarantee from applying this FPTAS as a separation oracle for dual LP (2.2).

LEMMA 3.3 (FPTAS). *There is an algorithm for the separation oracle problem that returns an outcome S with likelihood ratio $\leq (1 + \delta)\rho^*$ in time polynomial in $m, \frac{1}{\delta}$.*

LEMMA 3.4. *If the separation oracle FPTAS with parameter δ does not find a violated constraint of dual*

---

[8] In fact the problem is strongly NP-hard; but because it involves products of the form $q_{i,S} = (\prod_{j \in S} q_{i,j})(\prod_{j \notin S}(1 - q_{i,j}))$, the strong NP-hardness does not rule out an FPTAS [Papadimitriou and Steiglitz, 1982, Theorem 17.12].

LP (2.2), then for every S the inequality in (3.3) holds approximately up to $(1 + \delta)$:

$$(\sum_{i' \neq i} \lambda_{i'}) - 1 \leq (1 + \delta) \sum_{i' \neq i} \lambda_{i'} \frac{q_{i',S}}{q_{i,S}}.$$

Proof. Assume there exists S such that $(\sum_{i' \neq i} \lambda_{i'}) - 1 > (1 + \delta) \sum_{i' \neq i} \lambda_{i'} \frac{q_{i',S}}{q_{i,S}}$. Then dividing by $(\sum_{i'} \lambda_{i'})$ and using the definition of ρ^* as the minimum likelihood ratio we get $1 - \frac{1}{\sum_{i'} \lambda_{i'}} > (1 + \delta)\rho^*$. Combining this with the guarantee of Lemma 3.3, the FPTAS returns S' with likelihood ratio $< 1 - \frac{1}{\sum_{i'} \lambda_{i'}}$, thus identifying a violated constraint. This completes the proof. $\qquad\square$

Applying the separation oracle FPTAS: The standard method. Given an FPTAS with parameter δ for the separation oracle of a dual LP, for many problems it is possible to find in polynomial time an approximately-optimal, feasible solution to the primal— see, e.g., [Karmarkar and Karp, 1982; Carr and Vempala, 2002; Jain et al., 2003; Nutov et al., 2006; Fleischer et al., 2011; Feldman et al., 2012]. We first describe a fairly standard approach in the literature to utilizing a separation oracle FPTAS, which we refer to as the *standard method*, and explain where we must deviate from this approach. The proof of Theorem 3.1 then applies an appropriately modified approach.

The standard method works as follows: Let OPT_i be the optimal value of the primal (minimization) LP. For a benchmark value Γ, add to the (maximization) dual LP a constraint that requires its objective to be at least Γ, and attempt to solve the dual by running the ellipsoid algorithm with the separation oracle FPTAS.

Assume first that the ellipsoid algorithm returns a solution with value Γ. Since the separation oracle applies the FPTAS, it may wrongly conclude that some solution is feasible despite a slight violation of one or more of the constraints. For example, if we were to apply the FPTAS separation oracle from Lemma 3.3 to solve dual LP (2.2), we could possibly get a solution for which there exists S such that:

$$\sum_{i' \neq i} \lambda_{i'} \frac{q_{i',S}}{q_{i,S}} < (\sum_{i' \neq i} \lambda_{i'}) - 1 \leq (1 + \delta) \sum_{i' \neq i} \lambda_{i'} \frac{q_{i',S}}{q_{i,S}}$$

where the second inequality is by Lemma 3.4. Clearly, the value Γ of an approximately-feasible solution may be higher than OPT_i. In the standard method, the approximately-feasible solution can be *scaled* by $\frac{1}{1+\delta}$ to regain feasibility while maintaining value of $\frac{\Gamma}{1+\delta}$. Scaling thus establishes that $\frac{\Gamma}{1+\delta} \leq OPT_i$. Now assume that for some (larger) value of Γ, the ellipsoid algorithm identifies that the dual LP is infeasible. In this case we can be certain that $OPT_i < \Gamma$, and we can also find

in polynomial time a primal feasible solution with value $< \Gamma$ (more details in the proof of Theorem 3.1 below).

Using binary search (in our case over the range $[c_i, R_i] \subseteq [0,1]$ since R_i is the maximum the principal can pay without losing money), the standard method finds the smallest Γ^* for which the dual is identified to be infeasible, up to a negligible binary search error ϵ. This gives a primal feasible solution that achieves value $\Gamma^* + \epsilon$, and at the same time establishes that $\frac{(\Gamma^*)^-}{1+\delta} \leq OPT_i$ by the scaling argument.[9] So the standard method has found an approximately-optimal, feasible solution to the primal.

Applying the separation oracle FPTAS: Our method. The issue with applying the standard method to solve MIN-PAYMENT is that the scaling argument does not hold. To see this, consider an approximately-feasible dual solution for which $(\sum_{i' \neq i} \lambda_{i'}) - 1 \leq (1 + \delta) \sum_{i' \neq i} \lambda_{i'} \frac{q_{i',S}}{q_{i,S}}$ for every S, and notice that scaling the values $\{\lambda_{i'}\}$ does not achieve feasibility. We therefore turn to an alternative method to prove Theorem 3.1.

Proof of Theorem 3.1. We apply the standard method using the FPTAS with parameter δ (see Lemma 3.3) as separation oracle to the following *strengthened* version of dual LP (2.2),[10] where the extra $(1+\delta)$ multiplicative factor in the constraints makes them harder to satisfy:

$$(3.4) \quad \max \sum_{i' \neq i} \lambda_{i'} (c_i - c_{i'})$$
$$\text{s.t.} \ (1+\delta)\Big((\sum_{i' \neq i} \lambda_{i'}) - 1\Big) \leq$$
$$\sum_{i' \neq i} \lambda_{i'} \frac{q_{i',S}}{q_{i,S}} \qquad \forall S \subseteq E, q_{i,S} > 0$$
$$\lambda_{i'} \geq 0 \qquad \forall i' \neq i, i' \in [n].$$

Let Γ^* be the infimum value for which dual LP (3.4) would be identified as infeasible. The ellipsoid algorithm is thus able to find an approximately-feasible solution to dual LP (3.4) with objective $(\Gamma^*)^-$. The key observation is that this solution is *fully* feasible with respect to the original dual LP (2.2). This is because if the separation oracle FPTAS does not find a violated constraint of dual LP (3.4), then for every S it holds that $(\sum_{i' \neq i} \lambda_{i'}) - 1 \leq \sum_{i' \neq i} \lambda_{i'} \frac{q_{i',S}}{q_{i,S}}$ (by the same argument as in the proof of Lemma 3.4). From the key observation it follows that

$$(3.5) \qquad (\Gamma^*)^- \leq OPT_i -$$

(despite the fact that the scaling argument does not hold).

Now let $\Gamma^* + \epsilon$ be the smallest value for which the binary search runs the ellipsoid algorithm for dual LP (3.4) and identifies its infeasibility. During its run for $\Gamma^* + \epsilon$, the ellipsoid algorithm identifies polynomially-many separating hyperplanes that constrain the objective to $< \Gamma^* + \epsilon$. Formulate a "small" primal LP with variables corresponding exactly to these hyperplanes. By duality, the small primal LP has a solution with objective $< \Gamma^* + \epsilon$, and moreover since the number of variables and constraints is polynomial we can find such a solution p^* in polynomial time. Observe that p^* is also a feasible solution to the primal LP corresponding to dual (3.4) (the only difference from the small LP is more variables):

$$(3.6) \quad \min \ (1+\delta) \sum_{S \subseteq E} q_{i,S} p_S$$
$$\text{s.t.} \ (1+\delta)\Big(\sum_{S \subseteq E} q_{i,S} p_S\Big) - c_i \geq$$
$$\sum_{S \subseteq E} q_{i',S} p_S - c_{i'} \quad \forall i' \neq i, i' \in [n]$$
$$p_S \geq 0 \qquad \qquad \forall S \subseteq E.$$

We have thus obtained a contract p^* that is a feasible solution to LP (3.6) with objective $(1 + \delta) \sum_{S \subseteq E} q_{i,S} p_S < \Gamma^* + \epsilon$. For action a_i, this contract pays the agent an expected transfer of $\sum_{S \subseteq E} q_{i,S} p_S < \frac{\Gamma^* + \epsilon}{1+\delta}$. We have the following chain of inequalities: $\sum_{S \subseteq E} q_{i,S} p_S \leq \frac{(\Gamma^*)^- + \epsilon}{1+\delta} \leq \frac{OPT_i + \epsilon}{1+\delta} \leq OPT_i$, where the second inequality is by (3.5), and the last inequality is by taking the binary search error to be sufficiently small.[11] To complete the proof we must show that p^* is δ-IC. This holds since the constraints of LP (3.6) ensure that for every action $a_{i'} \neq a_i$, using the notation $p_i = \sum_{S \subseteq E} q_{i,S} p_S$, we have $p_{i'} - c_{i'} \leq (1+\delta)p_i - c_i \leq p_i - c_i + \delta p_i \leq p_i - c_i + \delta$ (the last inequality uses that $p_i \leq R_i \leq 1$ by normalization). □

4 Hardness of approximation

In this section unlike the previous one, the number of actions is no longer assumed to be constant. We show a hardness of approximation result for optimal contracts, based on the known hardness of approximation for MAX-3SAT. In his landmark paper, Håstad [2001] shows that it is NP-hard to distinguish between a satisfiable MAX-3SAT instance, and one in which there is no assignment satisfying more than $7/8 + \alpha$ of the clauses, where α is an arbitrarily-small constant (Theorems 5.6 and 8.3 in [Håstad, 2001]). We build

---

[9]The notation $(\Gamma^*)^-$ means any number smaller than Γ^*.

[10]Strengthened duals appear, e.g., in [Nutov et al., 2006; Feldman et al., 2012].

[11]We use here that $OPT_i \geq c_i$ and that the number of bits of precision is polynomial.

upon this to prove our main technical contribution stated in Theorem 4.1, which immediately leads to our main results for this section in Corollaries 4.2-4.3.

THEOREM 4.1. *Let $c \in \mathbb{Z}, c \geq 3$ be an (arbitrarily large) constant integer. Let $\epsilon, \Delta \in \mathbb{R}, \epsilon > 0, \Delta \in [0, \frac{1}{20^c}]$ be such that $\frac{\epsilon - 2\Delta^{1/c}}{3} \in (0, \frac{1}{20}]$ and $(\frac{\epsilon - 2\Delta^{1/c}}{3})^c$ is an (arbitrarily small) constant. Then it is NP-hard to determine whether a principal-agent setting has an IC contract extracting full expected welfare, or whether there is no Δ-IC contract extracting $> \frac{1}{c} + \epsilon$ of the expected welfare.*

We present two direct implications of Theorem 4.1. First, Corollary 4.2 applies to the OPT-CONTRACT problem, and states hardness of approximation within any constant of the optimal expected payoff by an IC contract. (A similar result can be shown for MIN-PAYMENT; see full version.)

COROLLARY 4.2. *For any constant $c \in \mathbb{R}, c \geq 1$, it is NP-hard to approximate the optimal expected payoff achievable by an IC contract to within a multiplicative factor c.*

Corollary 4.2 suggests that in order to achieve positive results, we may want to follow the approach of the CDW framework and relax IC to Δ-IC. That is, instead of trying to compute in polynomial time an approximately-optimal IC contract, we should try to compute in polynomial time a Δ-IC contract with expected payoff that is guaranteed to approximately exceed that of the optimal IC contract. The next corollary establishes a computational limitation on this approach: Corollary 4.3 fixes a constant approximation factor c, and derives Δ for which a c-approximation by a Δ-IC contract is NP-hard to find. (It is also possible to reverse the roles—fix Δ and derive a constant approximation factor for which NP-hardness holds.) We shall complement this limitation with a positive result in Section 5.

COROLLARY 4.3. *For any constant $c \in \mathbb{R}, c \geq 5$ and $\Delta \leq (\frac{1}{4c})^c$, it is NP-hard to find a Δ-IC contract that guarantees $> \frac{2}{c}OPT$, where OPT is the optimal expected payoff achievable by an IC contract.*[12]

Proof. The corollary follows from Theorem 4.1 by setting $\epsilon = \frac{1}{c}$. □

It also follows from Theorem 4.1 and Corollary 4.3 that for every c, Δ as specified, it is NP-hard to approximate the optimal expected payoff achievable by

---

[12]The relevant hardness notion is more accurately FNP-hardness.

a Δ-IC contract to within a multiplicative factor $c/2$. That is, hardness of approximation also holds for δ-OPT-CONTRACT.

In the remainder of the section we prove Theorem 4.1. After a brief overview, Section 4.2 sets up some tools for the proof, in Section 4.3 we focus on the special case of $c = 2$, and in Section 4.4 we prove the more general statement for any constant c. Missing proofs appear in the full version.

4.1 Proof overview It will be instructive to consider first a version of Theorem 4.1 for the case of $c = 2$:

THEOREM 4.4. *Let $\epsilon, \Delta \in \mathbb{R}, \epsilon > 0, \Delta \in [0, \frac{1}{20^2}]$ be such that $\frac{\epsilon - 2\Delta^{1/2}}{3} \in (0, \frac{1}{20}]$ and $(\frac{\epsilon - 2\Delta^{1/2}}{3})^2$ is an (arbitrarily small) constant. Then it is NP-hard to determine whether a principal-agent setting has an IC contract extracting full expected welfare, or whether there is no Δ-IC contract extracting $> \frac{1}{2} + \epsilon$ of the expected welfare.*

This theorem is already interesting as it shows that even relaxing IC to Δ-IC where $\Delta \gg 0$, approximating the optimal expected payoff within 65% is computationally hard:

COROLLARY 4.5. *For any $\Delta \leq \frac{1}{20^2}$, it is NP-hard to find a Δ-IC contract that guarantees $> 0.65 \cdot OPT$, where OPT is the optimal expected payoff achievable by an IC contract.*

Proof. The corollary follows from Theorem 4.4 by setting $\epsilon = \frac{3}{20}$. □

To establish Theorem 4.4 we present a gap-preserving reduction from any MAX-3SAT instance φ to a principal-agent setting that we call the "product setting" (the reduction appears in Algorithm 2 and is analyzed in Proposition 4.15). The product setting encompasses a 2-action principal-agent "gap setting", in which any δ-IC contract for sufficiently small δ cannot extract much more than $\frac{1}{2}$ of the expected welfare (Proposition 4.8).

The special case of $c = 2$ captures most ideas behind the proof of the more general Theorem 4.1, but the analysis is simplified by the fact that to extract more than roughly $\frac{1}{2}$ of the expected welfare in the 2-action gap setting, there is a single action that the contract could potentially incentivize. The more general case involves gap settings with more actions (the reduction appears in Algorithm 3 and is analyzed in Proposition 4.17). To extract more than $\approx \frac{1}{c}$ of the expected welfare, the contract could potentially incentivize almost any one of these actions (Proposition 4.9).

Barrier to going beyond constant c. Our techniques for establishing Theorem 4.1 do not generalize

beyond constant values of c (the approximation factor). The reason for this is that we do not know of (c, ϵ, f)-gap settings (Definition 4.6) where $f(c, \epsilon) = o(\epsilon^c)$. As long as $f(c, \epsilon)$ is of order ϵ^c, the gap in the MAX-3SAT instance we reduce from must be between $7/8 + \epsilon^c$ and 1, and this gap problem is known to be NP-hard only for constant c. As Håstad [2001] notes, significantly stronger complexity assumptions may lead to hardness for slightly (but not significantly) larger values of c.

4.2 Key ingredients In this section we formalize the notions of "gap" and "SAT" principal-agent settings as well as the notion of an "average action", which will be useful in proving Theorems 4.1 and 4.4. The term "gap setting" reflects the gap between the first-best solution (i.e., the expected welfare), and the second-best solution (i.e., the expected payoff to the principal from the optimal contract). It will be convenient *not* to normalize gap settings (and thus also the product settings encompassing them). This makes our negative results only stronger, as we show next.

Unnormalized settings and a stronger δ-IC notion. Before proceeding we must define what we mean by a δ-IC contract in an unnormalized setting. Moreover we show that if Theorems 4.1 or 4.4 hold for unnormalized settings with the new δ-IC notion, then they also hold for normalized settings with the standard δ-IC notion.

Recall that in a *normalized* setting, action a_i that is δ-incentivized by the contract must satisfy δ-IC constraints of the form $p_i - c_i + \delta \geq p_{i'} - c_{i'}$ for every $i' \neq i$. In an *unnormalized* setting, an additive δ-deviation from optimality is too weak of a requirement; we require instead that a_i satisfy δ-IC constraints of the form

$$(4.7) \qquad (1 + \delta)p_i - c_i \geq p_{i'} - c_{i'} \quad \forall i' \neq i.$$

Two key observations are: (i) The constraints in (4.7) imply the standard δ-IC constraints if $p_i \leq 1$, as is the case if the setting is normalized; (ii) The constraints in (4.7) are invariant to scaling of the setting and contract (i.e., to a change of currency of the rewards, costs and payments). By these observations, a δ-IC contract according to the new notion in an unnormalized setting becomes a standard δ-IC contract after normalization of the setting and payments, with the same fraction of optimal expected welfare extracted as payoff to the principal.

Assume a negative result holds for unnormalized settings, i.e., it is NP-hard to determine between the two cases stated in Theorem 4.1 (or Theorem 4.4). Assume for contradiction this does not hold for normalized settings. Then given an unnormalized setting, we

can simply scale the expected rewards and costs to normalize it, and then determine whether or not there is an IC contract extracting full expected welfare. If such a contract exists, it is also IC and full-welfare-extracting in the unnormalized setting after scaling back the payments. On the other hand, by the discussion above, if there is no standard-notion Δ-IC contract extracting a given fraction of the expected welfare in the normalized setting, there can also be no such contract with the new Δ-IC notion in any scaling of the setting. We have this reached a contradiction to NP-hardness. We conclude that proving our negative results for unnormalized settings only strengthens these results.

Gap settings and their construction. We now turn to the definition of gap settings.

DEFINITION 4.6 (UNSTRUCTURED GAP SETTING). *Let $f(c, \epsilon) \in \mathbb{R}_{\geq 0}$ be an increasing function where $c \in \mathbb{Z}_{>0}$ and $\epsilon \in \mathbb{R}_{>0}$. An* unstructured (c, ϵ, f)-gap setting *is a principal-agent setting such that for every $0 \leq \delta \leq f(c, \epsilon)$, the optimal δ-IC contract can extract no more than $\frac{1}{c} + \epsilon$ of the expected welfare as the principal's expected payoff.*

For convenience we focus on (structured) gap settings as follows.

DEFINITION 4.7 (GAP SETTING). *A (c, ϵ, f)-gap setting is a setting as in Definition 4.6 with the following structure: there is a single item and c actions; the first action has zero cost; the last action has probability 1 for the item and maximum expected welfare among all actions.*

To construct a gap setting, we construct a principal-agent setting with a single item, c actions and parameter $\gamma \in \mathbb{R}_{>0}, \gamma < 1$. The construction is similar to [Dütting et al., 2019], but requires a different analysis. For every $i \in [c]$, set the probability of action a_i for the item to γ^{c-i}, and set a_i's cost to $c_i = (1/\gamma^{i-1}) - i + (i-1)\gamma$. Set the reward for the item to be $1/\gamma^{c-1}$. Observe that the expected welfare of action a_i is $i - (i-1)\gamma$, so the last action has the maximum expected welfare $c - (c-1)\gamma$. This establishes the structural requirements from a gap setting (Definition 4.7). Propositions 4.8 and 4.9 establish the gap requirements from a gap setting (Definition 4.6) for $c = 2$ and $c \geq 3$, respectively— the separation between these cases is for clarity of presentation. We use the former in Section 4.3, in which we show hardness for the $c = 2$ case; the latter is a generalization to arbitrary-large constant c.

PROPOSITION 4.8 (2-ACTION GAP SETTINGS). *For every $\epsilon \in (0, \frac{1}{4}]$, there exists a $(2, \epsilon, \epsilon^2)$-gap setting.*

PROPOSITION 4.9 (*c*-ACTION GAP SETTINGS). *For every $c \geq 3$ and $\epsilon \in (0, \frac{1}{4}]$, there exists a $(c, \epsilon, \epsilon^c)$-gap setting.*

Average actions and SAT settings. The motivation for the next definition is that given a contract, for an action to be IC or δ-IC it must yield higher expected utility for the agent in comparison to the "average action". Average actions are thus a useful tool for analyzing contracts.

DEFINITION 4.10 (AVERAGE ACTION). *Given a principal-agent setting and a subset of actions, by the* average action *we refer to a hypothetical action with the average of the subset's distributions, and average cost. (If a particular subset is not specified, the average is taken over all actions in the setting.)*

DEFINITION 4.11 (SAT SETTING). *A SAT principal-agent setting corresponds to a MAX-3SAT instance φ. If φ has n clauses and m variables then the SAT setting has n actions and m items. Two conditions hold: (1) φ is satisfiable if and only if there is an item set in the SAT setting that the average action leads to with zero probability; (2) If every assignment to φ satisfies at most $7/8 + \alpha$ of the clauses, then for every item set S the average action leads to S with probability at least $\frac{1-8\alpha}{2^m}$.*

PROPOSITION 4.12. *For every φ the reduction in Algorithm 1 runs in polynomial time on input φ and returns a SAT setting corresponding to φ.*

ALGORITHM 1: SAT setting construction in polytime

Input   : A MAX-3SAT instance φ with n clauses and m variables.
Output : A principal-agent SAT setting (Definition 4.11) corresponding to φ.
begin
> Given φ, construct a principal-agent setting in which every clause corresponds to an action with a product distribution, and for every variable there is a corresponding item. If variable j appears in clause i of φ as a positive literal, then let item j's probability in the ith product distribution be 0, and if it appears as a negative literal then let item j's probability be 1. Set all other probabilities to be $\frac{1}{2}$. We set the costs of all actions and the rewards for all items to be 0.

end

4.3 The $c = 2$ case: Proof of Theorem 4.4

In this section we present a polynomial-time reduction from MAX-3SAT to a product setting, which combines gap and SAT settings. The reduction appears in

Algorithm 2. We then analyze the guarantees of the reduction and use them to prove Theorem 4.4. Most of the analysis appears in Proposition 4.15, which shows that the reduction in Algorithm 2 is gap-preserving. Some of the results are formulated in general terms so they can be reused in the next section (Section 4.4).

Before turning to Proposition 4.15, we begin with two simple observations about the product setting resulting from the reduction.

ALGORITHM 2: Polytime reduction from MAX-3SAT to principal-agent

Input   : A MAX-3SAT instance φ with n clauses and m variables; a parameter $\epsilon \in \mathbb{R}_{\geq 0}$.
Output : A principal-agent *product setting* combining a *SAT setting* and a *gap setting*.
begin
> Combine the SAT setting corresponding to φ (attainable in polytime by Proposition 4.12) with a poly-sized $(2, \epsilon, \epsilon^2)$-gap setting (exists by Proposition 4.8) to get the product setting, as follows:
> - The product setting has $n + 1$ actions and $m + 1$ items: m "SAT items" correspond to the SAT setting items, and the last "gap item" corresponds to the gap setting item.
> - The upper-left block of the product setting's $(n+1) \times (m+1)$ matrix of probabilities is the SAT setting's $n \times m$ matrix of probabilities. The entire lower-left $1 \times m$ block is set to $\frac{1}{2}$. The entire upper-right $n \times 1$ block is set to the probability that action a_1 in the gap setting results in the item. The remaining lower-right 1×1 block is set to the probability that the last action in the gap setting results in the item.
> - In the product setting, the rewards for the m SAT items are set to 0, and the reward for the gap item is set as in the gap setting.
> - The costs of the first n actions in the product setting are the cost of action a_1 in the gap setting; the cost of the last action in the product setting is the cost of the last action in the gap setting.

end

OBSERVATION 4.13. *Partition all actions of the product setting but the last one into blocks of n actions each.[13] Every action in the ith block has the same expected reward for the principal as action a_i in the gap setting, and the last action in the product setting has the same expected reward as the last action in the gap setting.*

COROLLARY 4.14. *The optimal expected welfares of the product and gap settings are the same, and are determined by their respective last actions.*

---

[13]If the number of actions in the gap setting is 2, there is a single such block.

PROPOSITION 4.15 (GAP PRESERVATION BY ALGO-RITHM [2]). *Let φ be a MAX-3SAT instance for which either there is a satisfying assignment, or every assignment satisfies at most $7/8 + \alpha$ of the clauses for $\alpha \leq (0.05)^2$. Let $\Delta \leq (0.05)^2$. Consider the product setting resulting from the reduction in Algorithm [2] run on input $\varphi, \epsilon = 3\alpha^{1/2} + 2\Delta^{1/2} \leq \frac{1}{4}$. Then:*

1. *If φ has a satisfying assignment, the product setting has an IC contract that extracts full expected welfare;*

2. *If every assignment to φ satisfies at most $7/8 + \alpha$ of the clauses, the optimal Δ-IC contract can extract no more than $\frac{1}{2} + \epsilon$ of the expected welfare.*

Proof. First, if φ has a satisfying assignment, then there is a subset of SAT items that has zero probability according to every one of the first n actions. Consider the outcome S^* combining this subset together with the gap item. We construct a full-welfare extracting contract: the contract's payment for S^* is the cost of the last action in the product setting multiplied by 2^m (since the probability of S^* according to the last action is $1/2^m$), and all other payments are set to zero. It is not hard to see that the resulting contract makes the agent indifferent among all actions, so by tie-breaking in favor of the principal, the principal receives the full expected welfare as her payoff.

Now consider the case that every assignment to φ satisfies at most $7/8 + \alpha$ of the clauses, and assume for contradiction that there is a Δ-IC contract p for the product setting that extracts more than $\frac{1}{2} + \epsilon$ of the expected welfare. We derive from p a δ-IC contract p' for the $(2, \epsilon, \epsilon^2)$-gap setting where $\delta \leq \epsilon^2$, which extracts more than $\frac{1}{2} + \epsilon$ of the expected welfare. This is a contradiction to the properties of the gap setting (Definition [4.6]).

It remains to specify and analyze contract p': For brevity we denote the singleton containing the gap item by M', and define

$$(4.8) \quad p'(S') = \tfrac{1-8\alpha}{2^m} \sum_{S \subseteq [m]} p(S \cup S') \quad \forall S' \subseteq M',$$

where S' is either the singleton containing the gap item or the empty set. The starting point of the analysis is the observation that to extract $> \frac{1}{2} + \epsilon$ of the expected welfare in the product setting, contract p must Δ-incentivize the last action (this follows since the expected rewards and costs of the actions are as in the gap setting by Observation [4.13], and so the same argument as in the proof of Proposition [4.8] holds).

Claim [4.16] below establishes that if contract p Δ-incentivizes the last action in the product setting, then contract p' δ-incentivizes the last action in the gap

setting for $\delta = \frac{8\alpha + \Delta}{1 - 8\alpha}$. So indeed

$$\begin{aligned}
\delta &= \frac{8\alpha}{1 - 8\alpha} + \frac{\Delta}{1 - 8\alpha} \\
&\leq 9\alpha + 4\Delta \\
&= (3\alpha^{1/2})^2 + (2\Delta^{1/2})^2 \\
&\leq (3\alpha^{1/2} + 2\Delta^{1/2})^2 \quad = \epsilon^2,
\end{aligned}$$

using that $\alpha, \Delta \leq (0.05)^2$ for the first inequality.

Now observe that the expected payoff to the principal from contract p' δ-incentivizing the last gap setting action is at least that of contract p Δ-incentivizing the last product setting action: the payments of p' as defined in [4.8] are the average payments of p lowered by a factor of $(1 - 8\epsilon)$, and the expected rewards in the two settings are the same (Observation [4.13]). The expected welfares in the two settings are also equal (Corollary [4.14]). We conclude that like contract p in the product setting, contract p' guarantees extraction of $> \frac{1}{2} + \epsilon$ of the expected welfare in the gap setting. This leads to a contradiction and completes the proof of Proposition 4.15 (up to Claim [4.16] proved below). $\square$

The next claim is formulated in general terms so that it can also be used in Section [4.4]. It references the contract p' defined in [4.8].

CLAIM 4.16. *Assume every assignment to the MAX-3SAT instance φ satisfies at most $7/8 + \alpha$ of its clauses where $\alpha < \frac{1}{8}$, and consider the product and gap settings returned by the reduction in Algorithm [2] (resp., Algorithm [3]). If in the product setting the last action is Δ-incentivized by contract p, then in the gap setting the last action is δ-incentivized by contract p' for $\delta = \frac{8\alpha + \Delta}{1 - 8\alpha}$.*

Proof. Let g_i denote the distribution of action a_i in the gap setting and let c be the number of actions in this setting. In the product setting, by construction its last action assigns probability $\frac{g_c(S')}{2^m}$ to every set $S \cup S'$ such that S contains SAT items and $S' \subseteq M'$. Thus the expected payment for the last action given contract p is

$$\begin{aligned}
(4.9) \quad &\sum_{S \subseteq [m]} \sum_{S' \subseteq M'} \frac{g_c(S')}{2^m} p(S \cup S') \\
&= \frac{1}{1 - 8\alpha} \sum_{S' \subseteq M'} g_c(S') p'(S'),
\end{aligned}$$

where the equality follows from the definition of p' in [4.8]. Note that the resulting expression in [4.9] is precisely the expected payment for the last action in the gap setting given contract p', multiplied by factor $1/(1 - 8\alpha)$.

Similarly, for every $i \in c$ consider the average action over the ith block of n actions in the product

setting.[14] Again by construction, the probability this ith average action assigns to $S \cup S'$ is $\geq \frac{g_i(S')(1-8\alpha)}{2^m}$, where we use that the average action of the SAT setting has probability $\geq \frac{1-8\alpha}{2^m}$ for S (Definition 4.11). Thus the expected payment for the ith average action given contract p is at least

$$(4.10) \quad \sum_{S \subseteq [m]} \sum_{S' \subseteq M'} \frac{g_i(S')(1-8\alpha)}{2^m} p(S \cup S')$$
$$= \sum_{S' \subseteq M'} g_i(S') p'(S') \quad \forall i \in [c],$$

where again the equality follows from (4.8). Note that the resulting expression in (4.10) is precisely the expected payment for action a_i in the gap setting given contract p'.

We now use the assumption that in the product setting, contract p Δ-incentivizes the last action. This means the agent Δ-prefers the last action to the ith average action, which has cost zero. Combining (4.9) and (4.10) we get

$$(4.11) \quad \frac{1+\Delta}{1-8\alpha} \sum_{S' \subseteq M'} g_c(S') p'(S') - \mathcal{C}$$
$$\geq \sum_{S' \subseteq M'} g_i(S') p'(S') \quad \forall i \in [c],$$

where $\mathcal{C}$ denotes the cost of the last action in the product and gap settings. By definition of δ-IC, Inequality (4.11) immediately implies that in the gap setting, the last action is δ-IC given contract p' where $\delta = \frac{8\alpha+\Delta}{1-8\alpha}$, thus completing the proof of Claim 4.16. □

We can now use Proposition 4.15 to prove Theorem 4.4.

Proof of Theorem 4.4. Recall that $\frac{(\epsilon-2\Delta^{1/2})^2}{9}$ is a constant $\leq (0.05)^2$. Assume a polynomial-time algorithm for determining whether a principal-agent setting has a (fully-IC) contract that extracts the full expected welfare, or whether no Δ-IC contract can extract more than $\frac{1}{2} + \epsilon$. Then given a MAX-3SAT instance φ for which either there is a satisfying assignment or every assignment satisfies at most $\frac{7}{8} + \frac{(\epsilon-2\Delta^{1/2})^2}{9}$ of the clauses, by Proposition 4.15 the product setting (constructed in polynomial time) either has a full-welfare extracting contract or has no Δ-IC contract that can extract more than $\frac{1}{2} + \epsilon$. Since the algorithm can determine among these two cases, it can solve the MAX-3SAT instance φ. But by Håstad [2001] and since $\frac{(\epsilon-2\Delta^{1/2})^2}{9}$ is a constant, we know that there is no polynomial-time

algorithm for solving such MAX-3SAT instances unless $P = NP$. This completes the proof of Theorem 4.4. □

4.4 The general case: Proof of Theorem 4.1 In this section we formulate and analyze the guarantees of the reduction in Algorithm 3.

ALGORITHM 3: Generalized polytime reduction from MAX-3SAT to principal-agent

---

Input : A MAX-3SAT instance φ with n clauses and m variables; parameters $\epsilon \in \mathbb{R}_{\geq 0}$ and $c \in \mathbb{Z}_{>0}$ where $c \geq 3$.

Output : A principal-agent *product setting* combining copies of a *SAT setting* and a *gap setting*.

begin

Combine multiple copies of the SAT setting corresponding to φ (attainable in polytime by Proposition 4.12) with a poly-sized $(c, \epsilon, \epsilon^c)$-gap setting (exists by Proposition 4.9) to get the product setting, as follows:

- The product setting has $cn + 1$ actions and $m + 1$ items: m "SAT items" correspond to the SAT setting items, and the last "gap item" corresponds to the gap setting item.
- For every $i \in [c]$, consider the ith block of n rows of the product setting's $(cn + 1) \times (m + 1)$ matrix of probabilities. The ith block consists of row $(i-1) \cdot n + 1$ to row $i \cdot n$ and forms a submatrix of size $n \times (m + 1)$. The first m columns of the submatrix are set to a copy of the SAT setting's $n \times m$ matrix of probabilities, and the entire last column is set to the probability that action a_i in the gap setting results in the item. Finally, the first m entries of the last row of the product setting's matrix (i.e., row $cn + 1$) are set to $\frac{1}{2}$, and the last entry (the lower-right corner of the matrix) is set to the probability that the last action in the gap setting results in the item.
- In the product setting, the rewards for the m SAT items are set to 0, and the reward for the gap item is set as in the gap setting.
- For every $i \in [c]$, the costs of the n actions in block i are the cost of action a_i in the gap setting; the cost of the last action in the product setting is the cost of the last action in the gap setting.

end

PROPOSITION 4.17 (GAP PRESERVATION BY ALGORITHM 3). *Let $c \in \mathbb{Z}, c \geq 3$. Let φ be a MAX-3SAT instance for which either there is a satisfying assignment, or every assignment satisfies at most $7/8 + \alpha$ of the clauses for $\alpha \leq (0.05)^c$. Let $\Delta \leq (0.05)^c$. Consider the product setting resulting from the reduction in Algorithm 3 run on input $\varphi, c, \epsilon = 3\alpha^{1/c} + 2\Delta^{1/c} \leq \frac{1}{4}$. Then:*

---

[14] If $c = 2$ there is a single such block.

1. *If φ has a satisfying assignment, the product setting has an IC contract that extracts full expected welfare;*

2. *If every assignment to φ satisfies at most $7/8 + \alpha$ of the clauses, the optimal Δ-IC contract can extract no more than $\frac{1}{c} + \epsilon$ of the expected welfare.*

Proof. First, if φ has a satisfying assignment, then there is a subset of SAT items that has zero probability according to every one of the actions in the product setting except for the last action, and so we can construct a full-welfare extracting contract as in the proof of Proposition 4.15. From now on consider the case that every assignment to φ satisfies at most $7/8 + \alpha$ of the clauses, and assume for contradiction there is a Δ-IC contract p for the product setting that extracts more than $\frac{1}{c} + \epsilon$ of the expected welfare.

Consider the case that p Δ-incentivizes the last action in the product setting. Then we can derive from it a δ-IC contract p' for the $(c, \epsilon, \epsilon^c)$-gap setting where $\delta \leq \epsilon^c$, which extracts more than $\frac{1}{c} + \epsilon$ of the expected welfare. This is a contradiction to the properties of the gap setting (Definition 4.6). The construction of p' and its analysis are as in the proof of Proposition 4.15 (where Equation (4.8) defines p'), and so are omitted here except for the following verification: we must verify that indeed $\delta \leq \epsilon^c$. We know from Claim 4.16 that $\delta = \frac{8\alpha + \Delta}{1 - 8\alpha}$. As in the proof of Proposition 4.15 this is $\leq 9\alpha + 4\Delta$, and it is not hard to see that

$$9\alpha + 4\Delta \leq (3\alpha^{1/c})^c + (2\Delta^{1/c})^c \leq (3\alpha^{1/c} + 2\Delta^{1/c})^c = \epsilon^c,$$

where the first inequality uses that $c \geq 3$.

In the remaining case, p Δ-incentivizes an action a_{i^*k} in the product setting which is the kth action in block $i^* \in [c]$ (recall each block has n actions). We derive from p a contract p'_k (depending on k) for the gap setting that Δ-incentivizes a_{i^*} at the same expected payment. As in the proof of Proposition 4.17, this means that p'_k extracts $> \frac{1}{c} + \epsilon$ of the expected welfare in the gap setting. Since $\Delta \leq \delta = \frac{8\alpha + \Delta}{1 - 8\alpha}$ it follows from the argument above that $\Delta \leq \epsilon^c$, and so we have reached a contradiction to the properties of the gap setting (Definition 4.6).

We define p'_k as follows: Let s_k denote the distribution of action a_k in the SAT setting. For every subset $S' \subseteq M'$ of gap items,

$$(4.12) \quad p'_k(S') = \sum_{S \subseteq [m]} p(S \cup S') s_k(S) \quad \forall S' \subseteq M',$$

where S' is either the singleton containing the gap item or the empty set.

For the analysis, let g_i denote the distribution of action a_i in the gap setting. In the product setting, for every $i \in [c], k \leq n$ the expected payment for action a_{ik} by contract p is

$$(4.13) \quad \sum_{S \in [m]} \sum_{S' \subseteq M'} s_k(S) g_i(S') p(S \cup S').$$

In the gap setting, the expected payment for a_i by contract p'_k is $\sum_{S' \subseteq M'} g_i(S') p'(S')$, and by definition of p'_k in (4.12) this coincides with the expected payment in (4.13). We know that contract p Δ-incentivizes a_{i^*k} in the product setting, in particular against any action a_{ik} where $i \in [c] \setminus \{i^*\}$ (i.e., against actions in the same position k but in different blocks). This implies that contract p'_k Δ-incentivizes a_{i^*} in the gap setting against any action a_i, completing the proof. $\square$

We can now use Proposition 4.17 to prove Theorem 4.1. The proof is identical to that of Theorem 4.4 and so is omitted here.

5 Approximation guarantees

In this section we show that for any constant δ there is a simple, namely linear, δ-IC contract that extracts as expected payoff for the principal a c_δ-fraction of the optimal welfare, where c_δ is a constant that depends only on δ. Recall that a linear contract is defined by a parameter $\alpha \in [0, 1]$, and pays the agent $p_S = \alpha \sum_{j \in S} r_j$ for every outcome $S \subseteq M$.

THEOREM 5.1. *Consider a principal-agent setting with n actions. For every $\gamma \in (0, 1)$ and every $\delta > 0$ there is a δ-IC linear contract with expected payoff ALG where*

$$ALG \geq \left((1 - \gamma) \frac{1}{\lceil \log_{1+\delta}(\frac{1}{\gamma}) \rceil + 1} \right) \max_{i \in [n]} \{R_i - c_i\}.$$

An immediate corollary of Theorem 5.1 is that we can compute a δ-IC linear contract that achieves the claimed constant-factor approximation in polynomial time. By Corollary 4.2 we cannot achieve a similar result for IC (rather than δ-IC) contracts unless $P = NP$. In fact, an even stronger lower bound holds for the class of exactly IC linear (or, more generally, separable) contracts. These contracts cannot achieve an approximation ratio better than n (see [Dütting et al., 2019] and full version for details).

Geometric understanding of linear contracts. To prove Theorem 5.1 we will rely on the following geometric understanding of linear contracts developed in [Dütting et al., 2019]. Fix a principal-agent setting. For a linear contract with parameter $\alpha \in [0, 1]$ and an action a_i, the expected reward $R_i = \sum_S q_{i,S} r_S$ is split between the principal and the agent, leaving the principal with $(1 - \alpha) R_i$ in expected utility and the agent

with $\alpha R_i - c_i$ (the sum of the players' expected utilities is action a_i's expected welfare). The agent's expected utility for choosing action a_i *as a function of* α is thus a line from $-c_i$ (for $\alpha = 0$) to $R_i - c_i$ (for $\alpha = 1$). Drawing these lines for each of the n actions, we trace the agent's utility for his best action as α goes from 0 to 1. This gives us the *upper envelope* diagram for linear contracts in the given principal-agent setting.

We now analyze some properties of the actions along the upper envelope diagram (i.e., as the linear contract gives an increasingly higher fraction of the rewards to the agent). Let I_N be the subset of $N \leq n$ actions implementable by some linear contract. We subdivide the interval $[0,1]$ into $N \leq n$ intervals $T_1 = [\ell_1, r_1), T_2 = [\ell_2, r_2), \ldots, T_N = [\ell_N, r_N]$, with $\ell_1 = 0$, $\ell_i = r_{i-1}$ for $2 \leq i \leq N$, and $r_N = 1$. The subdivision is such that there is a bijection τ between the indices of actions $a_i \in I_N$ and between those of intervals $T_{\tau(i)}$, with the following properties:

1. For every $i \in [N]$ and α in the ith interval T_i, the linear contract with parameter α incentivizes action $a_{\tau^{-1}(i)}$. It follows that every action $a_i \in I_N$ appears on the upper envelope once, and the smallest α that incentivizes it is the left endpoint $\ell_{\tau(i)}$ of this action's interval.
2. For every $i \in [N]$ it holds that $c_{\tau^{-1}(i)} \leq c_{\tau^{-1}(i+1)}$, $R_{\tau^{-1}(i)} \leq R_{\tau^{-1}(i+1)}$, and $R_{\tau^{-1}(i)} - c_{\tau^{-1}(i)} \leq R_{\tau^{-1}(i+1)} - c_{\tau^{-1}(i+1)}$.

Notation. Our proof of Theorem 5.1 uses the following notation:

- We renumber the actions as they appear on the upper envelope from left to right. By the second property above, we get actions sorted by increasing cost, increasing expected reward and increasing expected welfare. In particular, the final action (action a_N after renaming) must be the action in A_n with the highest expected welfare.[15]
- For every linearly-implementable action $a_i \in I_N$, we denote by α_i the smallest parameter α of a linear contract that incentivizes a_i (i.e., the left endpoint of a_i's corresponding interval).

Approximation guarantee proof. With these definitions at hand, we are now ready to prove the theorem.

Proof of Theorem 5.1. Given $\gamma \in (0,1)$ and $\delta > 0$, set $\kappa = \lceil \log_{1+\delta}(\frac{1}{\gamma}) \rceil$. Subdivide the range $[0,1]$ of α-

---

parameters into $\kappa + 1$ intervals:

$$[0, \gamma(1+\delta)^0), [\gamma(1+\delta)^0, \gamma(1+\delta)^1),$$
$$[\gamma(1+\delta)^1, \gamma(1+\delta)^2), \ldots, [\gamma(1+\delta)^{\kappa-1}, 1].$$

For each interval $k \in [\kappa + 1]$, denote by $a_{h(k)}$ the action a_i with the highest expected reward for which α_i falls into this interval (for simplicity of presentation we assume without loss of generality that such an action exists for each interval).

Note that $h(k) < h(k+1)$ due to renaming and because actions appear on the upper envelope in non-decreasing order of expected reward. We require the following definition: For $k \geq 2$, define

$$\alpha_{h(k-1),h(k)} = \frac{c_{h(k)} - c_{h(k-1)}}{R_{h(k)} - R_{h(k-1)}},$$

i.e., $\alpha_{h(k-1),h(k)}$ is the α that makes the agent indifferent between actions $h(k-1)$ and $h(k)$. For $k = 1$ define $\alpha_{h(k-1),h(k)} = 0$.

The proof now proceeds by two claims. The first claim derives an upper bound on $\max_{i \in [n]}(R_i - c_i) = R_N - c_N$. The proof appears in the full version.

CLAIM 5.2. $\max_{i \in [n]}(R_i - c_i) = R_N - c_N \leq \sum_{k=1}^{\kappa+1}(1 - \alpha_{h(k-1),h(k)})R_{h(k)}$.

The second crucial observation is that while $\alpha_{h(k-1),h(k)}$ is generally smaller than $\alpha_{h(k)}$ and thus does not incentivize action $h(k)$, it still δ-incentivizes it.

CLAIM 5.3. *For $k = 2, \ldots, \kappa + 1$, the linear contract with $\alpha = \alpha_{h(k-1),h(k)}$ ensures that $\alpha R_{h(k)} - c_{h(k)} + \delta \geq \alpha R_i - c_i$ for every $i \in [n]$.*

Proof of Claim 5.3. The lines $R_{h(k)} - c_{h(k)}$ and $R_{h(k-1)} - c_{h(k-1)}$ intersect at $\alpha_{h(k-1),h(k)}$. By construction, their intersection must fall between, on the one hand, the left endpoint $\gamma(1+\delta)^{k-2}$ of the interval in which $\alpha_{h(k)}$ falls, and $\alpha_{h(k)}$ on the other hand. This shows that $(1+\delta)\alpha_{h(k-1),h(k)} \geq (1+\delta)\gamma(1+\delta)^{k-2} = \gamma(1-\delta)^{k-1} \geq \alpha_{h(k)}$. Combining this with the fact that $a_{h(k)}$ is incentivized exactly at $\alpha_{h(k)}$, we obtain that $\alpha_{h(k-1),h(k)}R_{h(k)} - c_{h(k)} + \delta \geq (1+\delta)\alpha_{h(k-1),h(k)}R_{h(k)} - c_{h(k)} \geq \alpha_{h(k)}R_{h(k)} - c_{h(k)} \geq \alpha_{h(k)}R_i - c_i$ for all $i \in [n]$, where the first inequality holds since $R_{h(k)} \leq 1$ by normalization. This completes the proof of Claim 5.3. $\triangle$

Using Claims 5.2 and 5.3, the theorem follows from the fact that to obtain a δ-IC linear contract we can either incentivize $a_{h(1)}$ at $\alpha = \alpha_{h(1)}$ or δ-incentivize one

---

of the actions $a_{h(2)}, \ldots, a_{h(\kappa+1)}$ at $\alpha = \alpha_{h(k-1),h(k)}$ with $2 \le k \le \kappa + 1$. Namely,

$$
\begin{aligned}
ALG &\ge \max\{(1 - \alpha_{h(1)})R_{h(1)}, (1 - \alpha_{h(1),h(2)})R_{h(2)}, \ldots, \\
&\qquad (1 - \alpha_{h(\kappa),h(\kappa+1)})R_{h(\kappa+1)}\} \\
&\ge (1 - \gamma) \max\{(1 - \alpha_{h(0),h(1)})R_{h(1)}, \\
&\qquad (1 - \alpha_{h(1),h(2)})R_{h(2)}, \ldots, \\
&\qquad (1 - \alpha_{h(\kappa),h(\kappa+1)})R_{h(\kappa+1)}\} \\
&\ge (1 - \gamma)\frac{1}{\kappa + 1}\sum_{i=1}^{\kappa+1}(1 - \alpha_{h(k-1),h(k)})R_{h(k)} \\
&\ge (1 - \gamma)\frac{1}{\kappa + 1}OPT,
\end{aligned}
$$

where for the first inequality applied to ALG we use Claim 5.3, for the second inequality we use $\alpha_{h(1)} \le \gamma$ and $\alpha_{h(0),h(1)} \ge 0$, for the third inequality we lower bound the maximum by the average, and for the final inequality we use Claim 5.2. □

6 Black-box model

We conclude by considering a *black-box model* which concerns non-necessarily succinct principal-agent settings. In this model, the principal knows the set of actions A_n, the cost c_i of each action $a_i \in A_n$, the set of items M and the rewards r_j for each item $j \in M$, but does not know the probabilities $q_{i,S}$ that action a_i assigns to outcome $S \subseteq M$. Instead, the principal has *query access* to the distributions $\{q_i\}$. Upon querying distribution q_i of action a_i, a (random) set is returned where S is selected with probability $q_{i,S}$. Our goal is to study how well a δ-IC contract in this model can approximate the optimal IC contract if limited to a polynomial number of queries (where the guarantees should hold with high probability over the random samples). Black-box models have been studied in other algorithmic game theory contexts such as signaling—see [Dughmi and Xu, 2016] for a successful example.

Let $\eta = \min\{q_{i,S} \mid i \in [n], S \subseteq M, q_{i,S} \ne 0\}$ be the minimum non-zero probability of any set of items under any of the actions. Note that then either $q_{i,S} = 0$ or $q_{i,S} \ge \eta$ for every S. In Section 6.1 we address the case in which η is inverse super-polynomial and obtain a negative result; in Section 6.2 we show a positive result for the case of inverse polynomial η.

6.1 Inverse super-polynomial probabilities

We show a negative result for the case where the minimum probability η is inverse super-polynomial, by showing that $\text{poly}(1/\sqrt{\eta})$ samples are required to obtain a constant factor multiplicative approximation better than ≈ 1.15. The negative result holds even for succinct settings, in which the unknown distributions are product distributions.

THEOREM 6.1. *Assume $\eta \le \eta_0 = 1/625$ and $\delta \le \delta_0 = 1/100$. Even with $n = 2$ actions and $m = 2$ items, achieving a multiplicative ≤ 1.15 approximation to the optimal IC contract through a δ-IC contract, where the approximation guarantee is required to hold with probability at least $1 - \gamma$, may require at least $s \ge -\log(\gamma)/(9\sqrt{\eta})$ queries.*

Proof. We consider a scenario with two settings, both of which have $n = 2$ actions and $m = 2$ items, and which differ only in the probabilities of the items given the second action. Let τ be some constant > 2 (to be fixed later), and let $\mu = \frac{\sqrt{\eta}}{\tau}$. Let $\beta = (1 + \frac{1}{\tau^2})^{-1}$ and note that $\beta < 1$.

Setting I:

	$r_1 = \frac{\beta}{\tau^2\mu}$	$r_2 = \frac{\beta}{\tau^2\mu}$	
$a_1:$	$\tau\mu$	$\tau\mu$	$c_1 = 0$
$a_2:$	$\tau^2\mu$	μ	$c_2 = \frac{\tau-1}{\tau^3}\frac{1}{1-\mu}\beta$

Setting II:

	$r_1 = \frac{\beta}{\tau^2\mu}$	$r_2 = \frac{\beta}{\tau^2\mu}$	
$a_1:$	$\tau\mu$	$\tau\mu$	$c_1 = 0$
$a_2:$	μ	$\tau^2\mu$	$c_2 = \frac{\tau-1}{\tau^3}\frac{1}{1-\mu}\beta$

Note further that the minimum probability of any set of items in both settings is $q_{2,\{1,2\}} = \tau^2\mu^2 = \eta$, as required by definition of η.

The expected reward achieved by the two actions in the two settings is $R_1 = 2\beta/\tau < 1$ and $R_2 = (1 + 1/\tau^2)\beta = 1$. Moreover, the cost of action 2 is $c_2 \le \beta/\tau^2$. So the welfare achieved by the two actions is $R_1 - c_1 < \beta$ and $R_2 - c_2 \ge \beta$.

In both settings the optimal IC contract incentivizes action 2, by paying only for the set of items that maximizes the likelihood ratio. In Setting 1 this is $\{1\}$, in Setting 2 it is $\{2\}$. The payment for this set in both cases is $c_2/(\tau^2\mu(1 - \mu) - \tau\mu(1 - \tau\mu)) = c_2/(\tau^2\mu - \tau\mu)$. This leads to an expected payment of $\tau^2\mu(1 - \mu) \cdot c_2/(\tau^2\mu - \tau\mu) = \beta/\tau^2$. The resulting payoff (and our benchmark) is therefore $R_2 - \beta/\tau^2 = \beta$.

We now argue that if we cannot distinguish between the two settings, then we can only achieve a ≈ 1.1568 approximation. Of course, we can always pay nothing and incentivize action 1, but this only yields a payoff of $2\beta/\tau$. We can also try to δ-incentivize action 2 in both settings, by paying for outcome $\{1\}$ and $\{2\}$. But (as we show below) the payoff that we can achieve this way is (for $\delta \to 0$ and $\mu \to 0$) at most $(1 + 1/\tau^2 - (\tau^2 + 1)/((\tau - 1)\tau^3))\beta$. Now $\max\{2/\tau, 1 + 1/\tau^2 - (\tau^2 + 1)/((\tau - 1)\tau^3)\}$ is

minimized at $\tau = 1+\sqrt{2}$ where it is $2/(1+\sqrt{2}) \approx 0.8284$. The upper bound on the payoff from action 2 for this choice of τ is actually increasing in both μ and δ and $\approx 0.8644 \cdot \beta$ at the upper bounds $\mu_0 = \sqrt{\eta_0}/(2^2) = 1/100$ and $\delta_0 = 1/100$, implying that the best we can achieve without knowing the setting is a $\approx 1/0.8644 \approx 1.1568$ approximation.

So if we want to achieve at least a ≤ 1.15 approximation with probability at least $1 - \gamma$, then we need to be able to distinguish between the two settings with at least this probability. A necessary condition for being able to distinguish between the two settings is that we see at least some item in one of our queries to action 2. So,

$$1 - \gamma \leq 1 - (1 - \tau^2\mu)^{2s},$$

which implies that $s \geq \log(\gamma)/(2\log(1 - \tau^2\mu) \geq -\log(\gamma)/(2 \cdot \mu \cdot \tau^2) \geq -\log(\gamma)/(18\mu)$. Plugging in μ we get $s \geq -\log(\gamma)/(18\frac{\sqrt{\mu}}{\tau}) > -\log(\gamma)/(9\sqrt{\mu})$.

We still need to prove our claims regarding the payoff that we can achieve if we want to δ-incentivize action 2 in both settings. To this end consider the IC constraints for δ-incentivizing action 2 over action 1 in Setting I and Setting II, respectively:

$$\tau^2\mu(1 - \mu)p_{\{1\}} + (1 - \tau^2\mu)\mu p_{\{2\}} - c_2 \geq$$
$$\tau\mu(1 - \tau\mu)p_{\{1\}} + (1 - \tau\mu)\tau\mu p_{\{2\}} - \delta$$
$$(1 - \tau^2\mu)\mu p_{\{1\}} + \tau^2\mu(1 - \mu)p_{\{2\}} - c_2 \geq$$
$$\tau\mu(1 - \tau\mu)p_{\{1\}} + (1 - \tau\mu)\tau\mu p_{\{2\}} - \delta$$

Adding up these constraints yields

$$(\tau^2\mu(1 - \mu) + (1 - \tau^2\mu)\mu - 2\tau\mu(1 - \tau\mu)) \cdot (p_{\{1\}} + p_{\{2\}})$$
$$\geq 2c_2 - 2\delta$$

We maximize the minimum performance across the two settings by choosing $p_{\{1\}} = p_{\{2\}}$. Letting $p = p_{\{1\}} = p_{\{2\}}$ we thus obtain

$$(\tau^2\mu(1 - \mu) + (1 - \tau^2\mu)\mu - 2\tau\mu(1 - \tau\mu))p \geq c_2 - \delta.$$

It follows that

$$p \geq \frac{c_2 - \delta}{\tau^2\mu + \mu - 2\tau\mu}$$

The performance of the optimal contract that δ-incentivizes action 2 in both settings thus achieves an expected payoff of

$$R_2 - (\tau^2\mu(1 - \mu) + (1 - \tau^2\mu)\mu)\frac{c_2 - \delta}{\tau^2\mu + \mu - 2\tau\mu} =$$
$$R_2 - \frac{\tau^2(1 - 2\mu) + 1}{(\tau - 1)^2}(c_2 - \delta)$$

Plugging in R_2 and c_2 and letting $\delta \to 0$ and $\mu \to 0$ we obtain the aforementioned $1 + 1/\tau^2 - (\tau^2 + 1)/((\tau-1)\tau^3)\beta$. Finally, to see that the expected payoff evaluated at $\tau = 1+\sqrt{2} > 2$ is increasing in both δ and μ observe that the derivative in δ is simply the probability term $(\tau^2(1 - 2\mu) + 1)/(\tau - 1)^2$ which is positive and that both this probability term and the cost c_2 are decreasing in μ implying that as μ increases we subtract less. $\square$

6.2 Inverse polynomial probabilities We show a positive result for the case where the minimum probability η is inverse polynomial. Namely, let OPT denote the expected payoff of the optimal IC contract; then with $\text{poly}(n, m, \frac{1}{\eta}, \frac{1}{\epsilon}, \frac{1}{\gamma})$ queries it is possible to find with probability at least $(1 - \gamma)$ a 4ϵ-IC contract with payoff at least $OPT - 5\epsilon$. Formally:

THEOREM 6.2. *Fix $\epsilon > 0$, and assume $\epsilon \leq 1/2$. Fix distributions Q such that $q_{i,S} \geq \eta$ for all $i \in [n]$ and $S \subseteq M$. Denote the expected payoff of the optimal IC contract for distributions Q by OPT. Then there is an algorithm that with $s = (3\log(\frac{2n}{\eta\gamma}))/(\eta\epsilon^2)$ queries to each action and probability at least $1 - \gamma$, computes a contract $\tilde{p}$ which (i) is 4ϵ-IC on the actual distributions Q; and (ii) has expected payoff Π on the actual distributions satisfying $\Pi \geq OPT - 5\epsilon$.*

To prove Theorem 6.2, we first prove a series of lemmas (Lemmas 6.3 to 6.7). Proofs appear in the full version.

LEMMA 6.3. *Consider the algorithm that issues s queries to each action $i \in N$, and sets $\tilde{q}_{i,S}$ to be the empirical probability of set S under action i. With $s = (3\log(\frac{2n}{\eta\gamma}))/(\eta\epsilon^2)$ queries to each action, with probability at least $1 - \gamma$, for all $i \in [n]$ and $S \subseteq M$,*

$$(1 - \epsilon)q_{i,S} \leq \tilde{q}_{i,S} \leq (1 + \epsilon)q_{i,S}.$$

LEMMA 6.4. *Suppose that $(1 - \epsilon)q_{i,S} \leq \tilde{q}_{i,S} \leq (1 + \epsilon)q_{i,S}$ for all $i \in [n]$ and $S \subseteq M$. Consider contract p. If a_i is the action that is incentivized by this contract under the actual probabilities Q, then the payoff of a_i under the empirical distributions $\tilde{Q}$ is at least as high as that of any other action up to an additive term of 2ϵ.*

LEMMA 6.5. *Suppose that $(1 - \epsilon)q_{i,S} \leq \tilde{q}_{i,S} \leq (1 + \epsilon)q_{i,S}$ for all $i \in [n]$ and $S \subseteq M$. Consider contract $\tilde{p}$. If a_i is the action that is δ-incentivized by this contract under the empricial probabilities $\tilde{Q}$, then the payoff of a_i under the actual distributions is at least as high as that of any other action up to an additive term of $\delta + 2\epsilon$.*

LEMMA 6.6. *Suppose that $(1 - \epsilon)q_{i,S} \leq \tilde{q}_{i,S} \leq (1 + \epsilon)q_{i,S}$ for all $i \in [n]$ and $S \subseteq M$. If action a_i achieves payoff*

$\tilde{\Pi}$ *under contract $\tilde{p}$ when evaluated on the empirical distributions $\tilde{Q}$, then it achieves payoff $\Pi \geq \tilde{\Pi} - 2\epsilon$ when evaluated on the actual distributions Q.*

LEMMA 6.7. *Assume $\epsilon \leq 1/2$. Suppose that $(1 - \epsilon)q_{i,S} \leq \tilde{q}_{i,S} \leq (1+\epsilon)q_{i,S}$ for all $i \in [n]$ and $S \subseteq M$. If action a_i achieves payoff P under contract p when evaluated on the actual distributions Q, then it achieves payoff $\tilde{P} \geq P - 3\epsilon$ when evaluated on the empirical distributions Q.*

We are now ready to prove the theorem.

Proof of Theorem 6.2. Compute empirical probabilities $\tilde{Q}$ by querying each action s times. By Lemma 6.3, with probability at least $1 - \gamma$, the empirical probabilities obtained in this way will satisfy $(1 - \epsilon)q_{i,S} \leq \tilde{q}_{i,S} \leq (1 + \epsilon)q_{i,S}$ for all $i \in [n]$ and $S \subseteq M$.

Suppose we compute the optimal 2ϵ-IC contract $\tilde{p}$ on the empirical distributions $\tilde{Q}$. Denote the expected payoff achieved by this contract on $\tilde{Q}$ by $\tilde{\Pi}$, and the expected payoff it achieves on Q by Π. Likewise, consider the optimal IC contract p on the actual distributions Q. Denote the expected payoff OPT achieved by this contract on the actual distributions Q by P, and the expected payoff it achieves on $\tilde{Q}$ by $\tilde{P}$.

Note that by Lemma 6.5, contract $\tilde{p}$ which is 2ϵ-IC on $\tilde{Q}$ is 4ϵ-IC on Q. Furthermore, by Lemma 6.4, contract p which is IC on Q is 2ϵ-IC on $\tilde{Q}$. This implies that $\tilde{\Pi} \geq \tilde{P}$. Together with Lemma 6.6 and Lemma 6.7 we obtain

$$\Pi \geq \tilde{\Pi} - 2\epsilon \geq \tilde{P} - 2\epsilon \geq P - 5\epsilon,$$

which proves the theorem. $\square$

References

G. A. Akerlof. The market for "lemons": Quality uncertainty and the market mechanism. *The Quarterly Journal of Economics*, 84(3):488–500, 1970.

G. A. Akerlof. Labor contracts as partial gift-exchange. *The Quarterly Journal of Economics*, 97:543–569, 1982.

P. D. Azar and S. Micali. Computational principal-agent problems. *Theoretical Economics*, 13:553–578, 2018.

M. Babaioff and E. Winter. Contract complexity. In *EC'14*, page 911, 2014.

M. Babaioff, M. Feldman, N. Nisan, and E. Winter. Combinatorial agency. *Journal of Economic Theory*, 147(3):999–1034, 2012.

M. Babaioff, N. Immorlica, B. Lucier, and S. M. Weinberg. A simple and approximately optimal mechanism for an additive buyer. In *FOCS'14*, pages 21–30, 2014.

M. Babaioff, Y. A. Gonczarowski, and N. Nisan. The menu-size complexity of revenue approximation. In *STOC'17*, pages 869–877, 2017.

Y. Cai. *Mechanism design: A new algorithmic framework*. PhD thesis, Massachusetts Institute of Technology (MIT), 2013.

Y. Cai, C. Daskalakis, and S. M. Weinberg. An algorithmic characterization of multi-dimensional mechanisms. In *STOC'12*, pages 459–478, 2012a.

Y. Cai, C. Daskalakis, and S. M. Weinberg. Optimal multi-dimensional mechanism design: Reducing revenue to welfare maximization. In *FOCS'12*, pages 130–139, 2012b.

Y. Cai, C. Daskalakis, and S. M. Weinberg. Understanding incentives: Mechanism design becomes algorithm design. In *FOCS'13*, pages 618–627, 2013.

Y. Cai, N. R. Devanur, and S. M. Weinberg. A duality based unified approach to bayesian mechanism design. In *STOC'16*, pages 926–939, 2016.

B. Caillaud and B. E. Hermalin. Hidden-information agency. Lecture notes available from http://faculty.haas.berkeley.edu/hermalin/mechread.pdf, 2000.

R. D. Carr and S. Vempala. Randomized metarounding. *Random Structures and Algorithms*, 20(3):343–352, 2002.

G. Carroll. A quantitative approach to incentives: Application to voting rules. Working paper, 2013.

G. Carroll. Robustness and linear contracts. *American Economic Review*, 105(2):536–563, 2015.

Y. Cheng, H. Y. Cheung, S. Dughmi, E. Emamjomeh-Zadeh, L. Han, and S. Teng. Mixture selection, mechanism design, and signaling. In *FOCS'15*, pages 1426–1445, 2015.

C. Daskalakis and S. M. Weinberg. Symmetries and optimal multi-dimensional mechanism design. In *EC'12*, pages 370–387, 2012.

S. Dughmi. On the hardness of signaling. In *FOCS'14*, pages 354–363, 2014.

S. Dughmi and H. Xu. Algorithmic Bayesian persuasion. In *STOC'16*, pages 412–425, 2016.

S. Dughmi, N. Immorlica, and A. Roth. Constrained signaling in auction design. In *SODA'14*, pages 1341–1357, 2014.

S. Dughmi, J. D. Hartline, R. Kleinberg, and R. Niazadeh. Bernoulli factories and black-box reductions in mechanism design. In *STOC'17*, pages 158–169, 2017.

P. Dütting, T. Roughgarden, and I. Talgam-Cohen. Simple versus optimal contracts. In *EC'19*, pages 369–387, 2019.

Y. Emek and M. Feldman. Computing optimal contracts in combinatorial agencies. *Theoretical Computer Science*, 452:56–74, 2012.

F. Englmaier and S. Leider. Contractual and organizational structure with reciprocal agents. *American Economic Journal: Microeconomics*, 4(2):146–183, 2012.

M. Feldman, G. Kortsarz, and Z. Nutov. Improved approximation algorithms for directed Steiner forest. *Journal of Computer and System Sciences*, 78(1):279–292, 2012.

L. Fleischer, M. X. Goemans, V. S. Mirrokni, and M. Sviridenko. Tight approximation algorithms for maximum separable assignment problems. *Mathematics of Operations Research*, 36(3):416–431, 2011.

Y. A. Gonczarowski. Bounding the menu-size of approximately optimal auctions via optimal-transport duality. In *STOC'18*, pages 123–131, 2018.

Y. A. Gonczarowski and S. M. Weinberg. The sample complexity of up-to-ϵ multi-dimensional revenue maximization. In *FOCS'18*, pages 416–426, 2018.

S. J. Grossman and O. D. Hart. An analysis of the principal-agent problem. *Econometrica*, 51(1):7–45, 1983.

J. D. Hartline and B. Lucier. Non-optimal mechanism design. *American Economic Review*, 105(20):3102–3124, 2015.

J. D. Hartline, R. Kleinberg, and A. Malekian. Bayesian incentive compatibility via matchings. *Games and Economic Behavior*, 92:401–429, 2015.

J. Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, 2001.

C. Ho, A. Slivkins, and J. W. Vaughan. Adaptive contract design for crowdsourcing markets: Bandit algorithms for repeated principal-agent problems. *Journal of Artificial Intelligence Research*, 55:317–359, 2016.

K. Jain, M. Mahdian, and M. R. Salavatipour. Packing Steiner trees. In *SODA'03*, pages 266–274, 2003.

A. X. Jiang and K. Leyton-Brown. Polynomial-time computation of exact correlated equilibrium in compact games. *Games and Economic Behavior*, 91:347–359, 2015.

N. Karmarkar and R. M. Karp. An efficient approximation scheme for the one-dimensional bin-packing problem. In *FOCS'82*, pages 312–320, 1982.

J. M. Kleinberg and R. Kleinberg. Delegated search approximates efficient search. In *EC'18*, pages 287–302, 2018.

B. Koszegi. Behavioral contract theory. *Journal of Economic Literature*, 52(4):1075–1118, 2014.

R. B. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1):58–73, 1981.

Z. Nutov, I. Beniaminy, and R. Yuster. A $(1\text{-}1/e)$-approximation algorithm for the generalized assignment problem. *Operations Research Letters*, 34(3):283–288, 2006.

C. H. Papadimitriou. The complexity of finding Nash equilibria. In N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, editors, *Algorithmic Game Theory*, chapter 2, pages 29–51. Cambridge University Press, 2006.

C. H. Papadimitriou and T. Roughgarden. Computing correlated equilibria in multi-player games. *Journal of the ACM*, 55(3):14:1–14:29, 2008.

C. H. Papadimitriou and K. Steiglitz. *Combinatorial optimization: Algorithms and complexity*. Prentice-Hall, 1982.

Royal Swedish Academy of Sciences. Scientific background on the 2016 Nobel Prize in Economic Sciences, 2016.

B. Salanié. *The Economics of Contracts: A Primer*. MIT Press, 2005.

D. Walton and G. Carroll. When are robust contracts linear? Working paper, 2019.

S. M. Weinberg. *Algorithms for strategic agents*. PhD thesis, Massachusetts Institute of Technology (MIT), 2014.