Contracts under Moral Hazard and Adverse Selection

GURU GURUGANESH, Google Research
JON SCHNEIDER, Google Research
JOSHUA R. WANG, Google Research

In the classical principal-agent problem, a principal must design a contract to incentivize an agent to perform an action on behalf of the principal. We study the classical principal-agent problem in a setting where the agent can be of one of several types (affecting the outcome of actions they might take). This combines the contract theory phenomena of “moral hazard” (incomplete information about actions) with that of “adverse selection” (incomplete information about types).

We examine this problem through the computational lens. We show that in this setting it is APX-hard to compute either the profit-maximizing single contract or the profit-maximizing menu of contracts (as opposed to in the absence of types, where one can efficiently compute the optimal contract). We then show that the performance of the best linear contract scales especially well in the number of types: if agent has \( n \) available actions and \( T \) possible types, the best linear contract achieves an \( O(n \log T) \) approximation of the best possible profit. Finally, we apply our framework to prove tight worst-case approximation bounds between a variety of benchmarks of mechanisms for the principal.

CCS Concepts: • Theory of computation → Algorithmic game theory; Algorithmic mechanism design.

Additional Key Words and Phrases: contract theory, mechanism design

ACM Reference Format:
Guru Guruganesh, Jon Schneider, and Joshua R. Wang. 2021. Contracts under Moral Hazard and Adverse Selection. In Proceedings of the 22nd ACM Conference on Economics and Computation (EC ’21), July 18–23, 2021, Budapest, Hungary. ACM, New York, NY, USA, 20 pages. https://doi.org/10.1145/3465456.3467637

1 INTRODUCTION

Consider the classical principal-agent problem, wherein one self-interested economic actor (the “agent”) is hired to take actions on behalf of another (the “principal”). The problem arises because the agent has little to no personal stake in the resulting outcome, so the two enter into a contract designed to encourage the agent to take actions in the interest of the principal. Contract theory distinguishes between two fundamental information asymmetries to consider in the design of this contract. The first is moral hazard, which occurs when it may be difficult or costly for the principal to monitor the agent’s actions. In this case, the agent may take inappropriate actions, e.g. an insured party may begin to act more recklessly since they are shielded from risk. The second is adverse selection (also known as screening), where the agent possesses some private information about their type (e.g. how costly it is for them to perform certain actions) before they decide whether to enter the contract. In this case, the agents that enter contract may be biased, e.g. people with pre-existing conditions would prefer to purchase health insurance.

We study the confluence of these two asymmetries, moral hazard and adverse selection, under the computational lens. In particular, we consider a simple hidden-action principal-agent model [10]
but generalized to capture multiple possible agent types. In our generalized model, the agent may be one of \( T \) different types. The agent has the choice between \( n \) actions, each of a different cost to the agent (typically representing different effort levels the agent may exert). After the agent chooses an action, one of \( m \) disjoint outcomes may occur; the probability of a particular outcome depends on both the agent’s type and the chosen action. The principal observes this outcome, but does not observe the chosen action (creating a moral hazard issue) nor the type of the agent (creating an adverse selection issue).

The principal also receives a reward depending on the outcome that occurred. They wish to incentivize the agent to play actions that generate high rewards in expectation. To this end, the principal may offer the agent a contract: a mapping from outcomes to transfer amounts with the promise that if a particular outcome occurs, the principal will transfer the corresponding amount to the agent. The principal wishes to maximize their total profit, the rewards they receive less the amount they must transfer to the agent. We emphasize that this model is quite general and many applications do indeed have a principal subject to both information asymmetries. For example, consider a health insurance provider who can offer several different insurance policies (i.e. a menu of contracts), which incentivize different types of agents (e.g. those with and without pre-existing health conditions) to perform certain actions (e.g. to exercise, to get regular checkups). The principal must worry about both the moral hazard aspect of contracts while also accounting for adverse selection.

1.1 Our Results

Contracts without Types. Before we proceed to our results, we give a brief sketch of analogous results for the principal-agent problem without types. Many of these results appear in [8], which was one of the first papers to study contracts from an algorithmic/computational perspective.

Without types, it is possible to efficiently find the optimal contract for a given principal-agent problem – one way to do so is to, for each action, find the optimal contract that induces the agent to play this action by solving a linear program. However, the optimal contracts are often complex and unnatural: they can be non-monotone (better outcomes for the principal don’t necessarily correspond to better payments for the agent), non-budget-balanced (the principal may pay more than their value for certain outcomes on those outcomes), and can be very sensitive to the parameters of the problem (in particular the forecast probabilities \( F \)). We refer the reader to Dütting et al. [8] for examples and a more detailed discussion.

Such contracts rarely occur in practice. Instead, people tend to use simpler, more-easily understood contracts. One of the simplest classes of contracts is the class of linear contracts where the principal transfers a constant \( \alpha \) fraction \( (\alpha \in [0, 1]) \) of their total reward to the agent.

In [8], the authors also show that when there are \( n \) actions the best linear contract achieves an \( O(n) \)-approximation to the best overall contract. In fact, their proof further shows that the profit of the best linear contract is an \( O(n) \)-approximation to the best profit achievable in the absence of moral hazard, when the principal can directly observe (and contract on) the agent’s action (this is often referred to in economics as the “first-best” outcome). In this case, the principal will persuade the agent to take the action \( i \) (by paying them \( c_i \) if and only if they play \( i \)) which maximizes the principal’s expected reward less the agent’s cost. In addition to being the optimal profit in the absence of moral hazard, this quantity can also be viewed as the maximum possible expected welfare – we will henceforth refer to it as the “optimal welfare”.

Contracts with Types. When the principal only knows that the agent is one of many possible types, the problem becomes more difficult.
To begin with, it is no longer necessarily true that the best mechanism for the principal consists of presenting the agent with a single contract. As is the case in traditional Bayesian mechanism design (which also has hidden types), the principal may wish to present the agent with a menu of different contracts (each specifying how much the principal would transfer for each outcome). Each agent now begins by choosing their most desirable contract from this menu, then picking the action which maximizes their utility as a result of this contract.

In the previous section, it was possible to efficiently (in polynomial time in the inputs) compute the optimal contract. Now it is no longer clear how to efficiently compute the optimal menu for multiple types. In fact, it is not even clear how to efficiently compute the optimal single contract for multiple types – the same LP approach as before now requires solving exponentially many LPs, one for each assignment from types to actions. In the online version of this paper\(^1\), we show that this difficulty is inherent: we show that computation of both the optimal single contract and optimal menu are APX-Hard. That is, not only is it NP-hard to compute the optimal single contract (or optimal menu) exactly, there exists a constant \(\epsilon > 0\) such that getting any \((1 - \epsilon)\)-approximation to the optimal profit is NP-Hard.

**Theorem 1 (APX-Hardness).** Given a principal-agent problem \((c, F, r)\), it is APX-Hard to compute the optimal contract/menu. In other words, there exists a constant \(\epsilon > 0\) such that it is NP-Hard to compute a (incentive-compatible) menu of contracts \(X\) or single contract \(x\) whose value approximates the optimal expected profit by a factor of \((1 - \epsilon)\).

We complement this result by giving a simple exponential time algorithm that compute the optimal contract.

**Lemma 2 (Exponential-Time Algorithm).** There exists an algorithm to compute the optimal contract in time

\[
O(\min(\text{poly}(n^T, m), \text{poly}(n^T T^m)))
\]

where \(n\) is the number of actions, \(m\) is the number of outcomes, and \(T\) is the number of types.

This algorithm is tight in the following sense: if the number of types \(T\) or number of outcomes \(m\) is constant, then it is polynomial-time. However, the hardness instances that we generate only contain a constant number of actions and hence preclude an algorithm whose runtime is exponential only in the number of actions.

The computational hardness of computing these optimal mechanisms make it even more important to understand the power of simple classes of contracts. For example, how does the profit of the best linear contract (which are efficiently computable) compare to that of the best single contract or best menu of contracts? One way to get such a bound is to look at the best linear contract for each type individually and take the best of these contracts: since each individual linear contract achieves an \(O(n)\)-approximation to the welfare for the corresponding type, and since there are \(T\) types, this method will result in an \(O(nT)\)-approximation to the total welfare (and hence to the best single contract or best menu). In Section 3, we further explore the case where each agent type has the same cost values, and show that this structure means that the best linear contract actually scales as a \(O(n \log T)\)-approximation to the total welfare, and that this is tight.

**Theorem 3 (Linear contracts).** For a given principal-agent problem with types, let \(\text{OPT-LINEAR}\) denote the profit of the best linear contract and let \(\text{WELFARE}\) denote the optimal welfare. Then

\[
\text{OPT-LINEAR} \geq \Omega \left( \frac{1}{n \log T} \right) \text{WELFARE}.
\]

\(^1\)The full online version of this paper is available at https://arxiv.org/abs/2010.06742.
Moreover, this is tight; for any $n$ and $T$ there exists an instance of the principal-agent problem where

\[
\text{Opt-Linear} \leq O\left(\frac{1}{n \log T}\right) \text{Welfare}.
\]

Our analysis pins down the pivotal problem parameter to be the number of unique "cost gaps". Consider each agent type separately, and sort the entries of each type’s cost vector $c$. The cost gaps are the differences between adjacent entries of the sorted cost vector, and our analysis hinges on how these cost gaps are repeated. For our main case where all types have the same costs, cost gaps are repeated $T$ times over the types and this repetition drives our improved upper bound. Other assumptions that reduce the number of unique cost gaps also imply an improved approximation ratio, e.g. an arithmetic sequence of cost values implies a $O(\log(nT))$ approximation ratio (see Corollary 5 for more details).

We next explore the more fine-grained gaps in power between different options available to the principal. For example, how much better is the best single contract than the best linear contract (in the presence of types)? What fraction of welfare is the best menu guaranteed to achieve? Specifically, we consider the following five benchmarks for the principal, in order from least powerful (linear contracts) to most powerful (welfare).

- **Linear Contract.** The expected profit of the best linear contract. We write this benchmark as Opt-Linear.
- **Single Contract.** The expected profit of the best single contract. We write this benchmark as Opt-Single.
- **Menu of Contracts.** The expected profit of the best menu of contracts. We write this benchmark as Opt-Menu.
- **Type-Aware Contract.** The discussion so far assumes that the principal does not (or is not allowed to) observe the type of the agent when designing the mechanism (i.e. the adverse selection asymmetry). If the principal is allowed to offer type-aware contracts, then the best mechanism for the principal is to simply offer each type of agent their optimal contract. We write this benchmark as Opt-TypeAware.
- **Welfare.** The maximum possible profit the principal can hope to obtain is upper bounded by the maximum total expected utility possible – the expected welfare. This is also the profit achievable when the principal can observe both the type of the agent and the action they take (the “first-best” benchmark). We write this benchmark as Welfare.

We characterize the optimal worst-case ratio between nearly all pairs of benchmarks: the results are summarized in Table 1. For all but two of the pairs of benchmarks, we show that the optimal worst-case ratio is equal to $\Theta(n \log T)$; that is, this $\Theta(n \log T)$ separation holds not only between Opt-Linear and Welfare, but at a finer-grained level between most pairs of intermediate benchmarks. One exception to this is the gap between Opt-TypeAware and Welfare, which is $\Theta(n)$ (this follows directly from the definition of Opt-TypeAware and the $O(n)$ approximation ratio of [8]). The other (possible) exception is the gap between Opt-Single and Opt-Menu; this gap we can only prove is between 2 (see Lemma 20 in the full version and $O(n \log T)$). Understanding the true value of this gap is an interesting open problem, and the last remaining piece of Table 1:

**Open Problem 1 (Power of menus).** What is the worst-case gap between the expected profit of the best single contract (Opt-Single) and the expected profit of the best menu of contracts (Opt-Menu)? Does the best single contract achieve a constant-factor approximation (independent of $n$, $m$, and $T$) to the best menu of contracts?
One may also observe that the number of outcomes $m$ does not appear in Table 1. This is not an oversight: all of our upper bounds (specifically Theorem 3) works for any number of outcomes $m \geq 1$, and all of our lower bound counterexamples only require a constant number of outcomes ($m \leq 5$). In other words, the gaps between different benchmarks do not seem to scale meaningfully in the number of outcomes.

1.2 Related Work
Contract Theory is a fundamental branch of economics and has been applied in a wide variety of contexts (see [6]). For a general introduction to contract theory, we refer the interested reader to the books of [5, 13] and the references therein. While the phenomena of moral hazard and adverse selection have both been studied extensively in the contract theory literature, to the best of our knowledge the interaction between these two phenomena has been significantly less explored. One such example is the work [14] which consider the principal-agent model with types where the types and actions are linked. Another (more closely related) example is the work of [9], where the types and effort levels can be arbitrarily related. Their work characterizes a number of different assumptions under which the optimal menu consists of a single contract.

Our work is most closely related (and in some sense is a direct successor) to the work of Dütting et al. ([7, 8]), which is one of the first papers we are aware of to study contract theory with the tools of approximation algorithms and computational complexity. In [8], the authors show that linear contracts are “robust” in a certain sense and that they can achieve an $O(n)$-approximation to welfare. The earlier work of Balmaceda et al. [4] also examines how well contracts could approximate the welfare, including linear contracts. In [7], the authors show that there exist natural principal-agent problems with exponentially-sized (but succinctly representable) outcome spaces for which finding the optimal contract is computationally hard.

Some earlier research studies specific variants of contract problems from a computational perspective. One such line of research is the work of [1] and the related work of [2, 3]. They introduce a model known as “Combinatorial Agency” where a team of agents needs to be incentivized to perform a set of hidden actions, where a combination of their actions yields a certain outcome. Another such line of work studies the problem of learning optimal contracts in a repeated setting [11]. In [15], the authors prove computational results for agents with multiple types (analogous to the results we show in the full version) but for the restricted setting where the principal can contract directly on the agents’ actions (we additionally show an APX-hardness result over the NP-hardness result in their paper). Finally, in [12], the authors study a variant of the principal-agent problem where the agent has their own set of utilities over outcomes (which may differ from those of the principal).

1.3 Organization
In Section 2, we set up the problem formally and introduce the five benchmarks that we will use to evaluate the principal’s strategy. In Section 3, we will show our main result that linear contracts are $\Theta(n \log T)$-competitive with respect to the optimal welfare. We next show how to use this reduction combined with a set of reductions to get tight results for almost all the benchmark comparisons in the full version. We complement these results by showing APX-hardness and exponential time algorithms to compute the optimal menu/contract also in the full version.

2 PRELIMINARIES
Throughout this paper, we use $[x]$ to denote the one-indexed set $\{1, 2, \ldots, x\}$. 

567
| Measure          | Running Time                  | Hardness                        |
|------------------|-------------------------------|---------------------------------|
| WELFARE          | $O(nmT)$                      |                                 |
| Opt-TypeAware    | $\Theta(n) \cdot \Theta(n \log T)$ |                                |
| Opt-Menu         | $\Theta(n \log T) \cdot \Theta(n \log T)$ | $O(n \log T)$ and $\geq 2$ |
| Opt-Single       | $\Theta(n \log T) \cdot \Theta(n \log T)$ | $\Theta(n \log T) \cdot \Theta(n \log T)$ |
| Opt-Linear       | $\Theta(n \log T) \cdot \Theta(n \log T)$ | $\Theta(n \log T) \cdot \Theta(n \log T)$ |

Table 1. Table of our results concerning worst-case ratios between our five benchmarks.

| Measure          | Running Time                  | Hardness                        |
|------------------|-------------------------------|---------------------------------|
| Welfare          | $O(nmT)$                      |                                 |
| Opt-TypeAware    | poly$(n, m, T)$               |                                 |
| Opt-Menu         | poly$(n^T, m)$                | APX-Hard with $n = 5$ (Theorem 1) |
| Opt-Single       | min(poly$(n^T, m)$, poly$(n^2T)^m)$) | APX-Hard with $n = 5$ (Theorem 1) |
| Opt-Linear       | poly$(n, m, T)$               |                                 |

Table 2. Table of our results concerning computability of our five measures.

2.1 The Classical Principal-Agent Problem

In the classical (discrete) principal-agent problem, there is an agent with $n$ actions and $m$ possible (disjoint) outcomes of these actions. Each action $i \in [n]$ incurs a cost $c_i \geq 0$ for the agent, and each outcome $j \in [m]$ endows the principal with some reward $r_j \geq 0$. Taking action $i$ leads to a distribution $D_i$ over outcomes; we write $F_{i,j}$ for the (forecasted) probability that outcome $j \in [m]$ arises as a result of the agent taking action $i \in [n]$. The principal can observe the eventual outcome $j \in [m]$, but not the action $i \in [n]$ that the agent takes (in the contract theory literature, this is referred to as the hidden-action model). Without loss of generality, we will always assume the actions are ordered by increasing cost ($c_1 \leq c_2 \leq \cdots \leq c_n$) and outcomes are ordered by increasing reward ($r_1 \leq r_2 \leq \cdots \leq r_m$). We further assume that there is a zero-cost action ($c_1 = 0$) and a zero-reward outcome ($r_1 = 0$); see the full version for why we can do so without loss of generality. We typically group these rewards together into a reward vector $r$, these forecasts together into a forecast matrix $F$, and these costs together into a cost vector $c$. These three things constitute the principal-agent problem ($c, F, r$).

The goal of the principal in this problem is to incentivize the agent to perform actions which lead to beneficial outcomes for the principal. To this end, the principal can offer the agent a contract. A contract is specified by an $m$-dimensional vector $x$, whose nonnegative entries $x_j \geq 0$ denotes the amount the principal will transfer to the agent in the event of outcome $j$. Given a contract $x$,
the agent chooses an action which maximizes their overall utility; i.e. the action $i^*(x)$ given by

$$i^*(x) \in \arg\max \left\{ \sum_{j=1}^{m} F_{i,j}x_j \right\} - c_i.$$ 

The principal’s expected profit is their expected reward minus the expected transfer:

$$\text{Profit}(c, F, r, x) \triangleq \sum_{j=1}^{m} F_{i^*(x),j}(r_j - x_j).$$

Note that in the instance where multiple actions maximize overall utility, we will tiebreak in favor of the principal and assume that the one that results in the highest profit is chosen. The principal can always approximate this tie-breaking behavior with arbitrary accuracy by adding a vanishing fraction of their rewards to the contract. Sometimes, when the principal-agent problem $(c, F, r)$ is known from context, we will abbreviate $\text{Profit}(c, F, r, x)$ to $\text{Profit}(x)$. It is known that this can be solved in polynomial time by solving $O(n)$ linear programs (each computing the minimum expected transfers to force action $i \in [n]$), but this may lead to strange or unnatural contracts that do not typically appear in practice.

One way to try to avoid the issue of strange or unnatural contracts is to restrict the principal to linear contracts, where the principal simply transfers some fraction $\alpha$ of their reward to the agent. In other words, linear contracts can be completely specified by a transfer coefficient $\alpha \in [0, 1]$: $x_j = \alpha r_j$. When considering linear contracts, we will often abuse notation and use $\alpha$ to refer to the contract itself, writing $i^*_\alpha$ and $\text{Profit}(\alpha)$ to denote the optimal action and profit under the contract $x = \alpha r$ respectively.

Another important quantity in the classical principal-agent problem is the maximum achievable welfare (sum of principal profit and agent utility). Since by assumption the agent always has a zero-cost action to fall back on, the (maximum) expected welfare is an upper bound on the principal’s expected profit. If we let $R_i$ denote the expected reward of the principal from the agent playing action $i$, i.e.

$$R_i \triangleq \sum_{j=1}^{m} F_{i,j}r_j,$$

then we can write the welfare $\text{Welfare}$ as

$$\text{Welfare}(c, F, r) \triangleq \max_i \left( \sum_{j=1}^{m} F_{i,j}(r_j - x_j) \right) = \max_i (R_i - c_i).$$

### 2.2 Typed Agents

In this paper, we want to incorporate an adverse selection information asymmetry: the principal is uncertain about the exact properties of the agent. Specifically, they are uncertain about the probabilities $F_{i,j}$ of action $i$ resulting in outcome $j$. We model this by supposing there are $T$ different types of agent. Although all our results generalize to the setting where the agent types are drawn from an arbitrary discrete distribution, we assume each agent occurs with the same probability to simplify our exposition\(^2\). Each type $t \in [T]$ has its own set of distributions $\{D_{i}^{(t)}\}_{i=1}^{m}$ over outcomes.

\(^2\)The proof of our main positive result, Theorem 4, goes through as written when the definition of $U(\alpha)$ is adjusted accordingly into a weighted sum of functions. Our negative results hold since the uniform distribution is just a specific discrete type distribution.
with associated probabilities \( F_{i,j}^{(t)} \), the probability that a type \( t \) agent playing action \( i \) results in outcome \( j \). Unless otherwise specified, we assume that the cost \( c_i \) of action \( i \in [n] \) remains constant over types\(^3\). This assumption is equivalent to saying that the type of an agent has no effect on what effort levels are available, just how effective each effort level is. As before, the principal only observes the eventual outcome \( j \); not the action \( i \) taken by the agent (hidden-action), nor the type of the agent \( t \in [T] \) (adverse selection).

Again, the principal could offer the agent a single contract parameterized by a transfer vector \( x \). An agent of type \( t \) will play the action \( i_t(x) \) given by

\[
i_t(x) = \arg\max_i \left( \sum_{j=1}^{m} F_{i,j}^{(t)} x_j \right) - c_i.
\]

We assume that each type \( t \) occurs in the population with some probability. In this case the principal’s profit from an agent of type \( t \) is given by

\[
\text{Profit}^{(t)}(c, F, r, x) = \sum_{j=1}^{m} F_{i_t(x),j}^{(t)} (r_j - x_j)
\]

and their overall expected profit is given by

\[
\text{Profit}(c, F, r, x) = \mathbb{E}_t \left[ \text{Profit}^{(t)}(c, F, r, x) \right].
\]

Again, we will tiebreak between multiple utility-maximizing actions in favor of maximizing profit. Unless otherwise specified, for simplicity we assume that all types occur with equal probability, i.e. the distribution over types is uniform. Again, if \( x \) is a linear contract with \( \alpha \) transfer coefficient, we abuse notation and write \( i_t(\alpha) \) and \( \text{Profit}(\alpha) \) for the utility-maximizing action taken by a type \( t \) agent and the total profit respectively.

With the addition of these types, a general principal can sometimes secure additional profit if instead they offer the agent a menu of contracts \( X = \{ x^{(1)}, \ldots, x^{(t)} \} \), where an agent who reports their type to be \( t \) receives the contract \( x^{(t)} \). We say that such a menu is “incentive-compatible” (IC) if no type \( t \) has an incentive to misreport their type as a different type \( t' \neq t \):

\[
\forall t, t' \in [T], \quad \max_i \left( \sum_{j=1}^{m} F_{i,j}^{(t)} x_j^{(t)} \right) - c_i \geq \max_i \left( \sum_{j=1}^{m} F_{i,j}^{(t')} x_j^{(t')} \right) - c_i. \quad \text{(IC)}
\]

The principal’s overall expected profit is given by

\[
\text{Profit}(c, F, r, X) = \mathbb{E}_t \left[ \text{Profit}^{(t)}(c, F, r, x^{(t)}) \right].
\]

\(^3\)We can model type-dependent costs with type-independent costs by adding additional actions. To do so, view the costs of a particular type as a multiset, and let the type-independent costs be the minimal super-multiset of these multisets. For each type’s newly introduced costs, let their corresponding dummy actions have the the same probability distribution as that type’s zero-cost action, ensuring that they are dominated by an existing action. For a detailed discussion of the effect of this reduction to the approximation ratios, see the full version.
We again define expected reward and welfare, the latter of which still upper-bounds the principal’s expected profit:

\[ R_i^{(t)} = \sum_{j=1}^{m} F_{i,j}^{(t)} r_j, \]

\[ \text{Welfare} (c, F, r) \triangleq \mathbb{E}_t \left[ \max_i \sum_{j=1}^{m} F_{i,j}^{(t)} r_j - c_i \right] = \mathbb{E}_t \left[ \max_i R_i^{(t)} - c_i \right]. \]

### 2.3 Five Benchmarks for Typed Principal-Agent Problems

We summarize the definitions of the five benchmarks we consider for principal-agent problems with types:

1. **Welfare**, the maximum expected welfare. This is the maximum expected profit attainable by a principal that can observe (and contract on) both types and actions.

   \[ \text{Welfare} (c, F, r) \triangleq \mathbb{E}_t \left[ \max_i \sum_{j=1}^{m} F_{i,j}^{(t)} r_j - c_i \right] \]

2. **Opt-TypeAware**, the expected profit of the best contract per type. This is the maximum profit attainable by a principal that can observe the types of agents (i.e. if the contracts he assigns can depend on the type of the agent).

   \[ \text{Opt-TypeAware} (c, F, r) \triangleq \max_{(x^{(1)}, x^{(2)}, \ldots, x^{(t)})} \mathbb{E}_t \left[ \text{Profit}^{(t)} (c, F, r, x^{(t)}) \right] \]

3. **Opt-Menu**, the expected profit of the best (incentive-compatible) menu of contracts.

   \[ \text{Opt-Menu} (c, F, r) \triangleq \max_{X = (x^{(1)}, \ldots, x^{(t)}) \text{ is IC}} \mathbb{E}_t \left[ \text{Profit}^{(t)} (c, F, r, x^{(t)}) \right] \]

4. **Opt-Single**, the expected profit of the best single contract.

   \[ \text{Opt-Single} (c, F, r) \triangleq \max_x \mathbb{E}_t \left[ \text{Profit}^{(t)} (c, F, r, x) \right] \]

5. **Opt-Linear**, the expected profit of the best single linear contract.

   \[ \text{Opt-Linear} (c, F, r) \triangleq \max_{\alpha \in [0, 1]} \mathbb{E}_t \left[ \text{Profit}^{(t)} (c, F, r, x = \alpha r) \right] \]

Each benchmark is an upper bound on the next:

\[ \text{Welfare} (c, F, r) \geq \text{Opt-TypeAware} (c, F, r) \geq \text{Opt-Menu} (c, F, r) \geq \text{Opt-Single} (c, F, r) \geq \text{Opt-Linear} (c, F, r). \]

### 3 LINEAR CONTRACTS

In this section, we explore the surprising power of linear contracts, in which the principal gives a constant \( \alpha \) fraction of their reward to the agent. Our main goal is to show that the profit of the optimal linear contract is always within a \( O(n \log T) \) factor of the optimal welfare and that this is tight; there exists a principal-agent problem \((c, F, r)\) where the best linear contract achieves profit that is a \( \Omega(n \log T) \) factor from the optimal welfare.

Central to both our upper and lower bounds are the following two observations about linear contracts. Recall our definition of the expected reward when a type \( t \) agent plays action \( i: R_i^{(t)} = \]
\[ \sum_{j=1}^{m} F_{i,j} r_j. \] The first observation is that \textsc{Welfare} and \textsc{Profit}(\alpha) can be written entirely in terms of \( \alpha, c, \) and the \( R_1^{(t)} \). In particular, the values of \( F_{i,j} \) and \( r_j \) do not affect either of these measures beyond their use in the computation of \( R_1^{(t)} \). Formally, we have that (recall that \( i_*(t) \) is the utility-maximizing action taken by type \( t \) under the linear contract \( \alpha \)):

\[
\textsc{Welfare} = \mathbb{E}_t \left[ \max_i \left( R_1^{(t)} - c_i \right) \right] \quad \text{and} \quad \textsc{Profit}(\alpha) = \mathbb{E}_t \left[ (1 - \alpha) R_1^{(t)} \right].
\]

The second observation is that \textsc{Profit}(\alpha) is a piecewise linear function with at most \( nT \) pieces. This stems from the fact that each \( \textsc{Profit}^{(t)}(\alpha) \) is a piecewise-linear function with at most \( n \) pieces, which is because for each range of \( \alpha \) where \( i_*(t) \) is constant, \( \textsc{Profit}^{(t)}(\alpha) \) is linear (and there are at most \( n \) possible values of \( i_*(t) \)).

In fact, we can say something even stronger. Let’s consider the resulting utility of a type \( t \) agent when offered a linear contract with transfer coefficient \( \alpha \) and the expectation of this quantity over types:

\[
U^{(t)}(\alpha) \doteq \max_i \left( \alpha R_1^{(t)} - c_i \right) \quad \text{and} \quad U(\alpha) \doteq \mathbb{E}_t \left[ U^{(t)}(\alpha) \right].
\]

By similar logic, \( U(\alpha) \) is a piecewise linear function with at most \( nT \) pieces. Moreover, from the definitions of profit and utility we have that:

\[
\textsc{Profit}(\alpha) = (1 - \alpha) \frac{d}{d\alpha} U(\alpha).
\]

Since \( \textsc{Welfare} = U(1) \), Equation (1) allows us to relate the expected profit of the optimal linear contract to the maximum expected welfare and phrase our original problem as an optimization problem over the piecewise linear functions \( U \). Naively solving this optimization problem (over all increasing piecewise linear functions \( U \) with \( nT \) pieces) leads to an \( O(nT) \)-approximation factor, but this does not use any information about types\(^4\). Since each type has the same set of allowable costs, this imposes additional constraints on \( U \). Specifically, the linear segments of \( U \), if extended, can have at most \( T \) different “y-intercepts”. We leverage these constraints to give an improved \( O(n \log T) \)-approximation ratio.

### 3.1 Linear Contracts: Profit Guarantee

In this subsection, we prove that the optimal linear contract obtains profit within an \( O(n \log T) \) factor of \textsc{Welfare}. In the proof of the following theorem, it will be helpful to assume that all actions are played by all types for some choice of \( \alpha \in [0, 1] \), i.e. for every type \( t \in [T] \) and action \( i \in [n] \), there exists some \( \alpha \in [0, 1] \) such that \( i_*(t) = i \). This holds without loss of generality because if action \( i \) is never chosen by type \( t \), we can increase \( R_1^{(t)} \) until it is barely chosen; this affects neither \textsc{Welfare} nor \textsc{Opt-Linear}.

**Theorem 4.** For any principal-agent problem \((c, F, r)\),

\[
\textsc{Opt-Linear}(c, F, r) \geq \Omega \left( \frac{1}{n \log T} \right) \textsc{Welfare}(c, F, r).
\]

\(^4\)It may be instructive to compare this with the Dütting et al. proof of the \( O(n) \) approximation factor of linear contracts, when types are not an issue [8].
Proof. Rewriting the theorem statement, we must show that there exists an \( \alpha \in [0, 1] \) such that
\[
\text{PROFIT}(\alpha) \geq \frac{\text{WELFARE}}{O(n \log(T))}.
\]

We begin by considering our utility and profit functions. Note that \( U^{(t)}(\alpha) \) is an increasing piecewise linear function with \( n \) pieces (one for each possible action). We denote its \( i^\text{th} \) breakpoint as \( \alpha_i^{(t)} \), which is where \( U^{(t)}(\alpha) \) changes from \( aR_i^{(t)} - c_i \) to \( aR_i^{(t)} - c_{i+1} \). We define some notation to express how much the multiplier on \( \alpha \) increases and how much the constant term decreases:
\[
\begin{align*}
    w_i^{(t)} & \triangleq R_{i+1}^{(t)} - R_i^{(t)} \\
    \delta_i^{(t)} & \triangleq c_{i+1} - c_i
\end{align*}
\]

One way of looking at the situation is that when we increase \( \alpha \) past \( \alpha_i^{(t)} \), we add the linear function \( w_i^{(t)} \alpha - \delta_i^{(t)} \) to \( U^{(t)}(\alpha) \).

The function \( U(\alpha) \) is the convex combination of \( T \) different functions \( U^{(t)}(\alpha) \). It therefore has \((n-1)T \) breakpoints: the union of breakpoints for all the functions \( U^{(t)}(\alpha) \). Sort these \( B \triangleq (n-1)T \) breakpoints in increasing order. Abusing notation, we now call the \( i^\text{th} \) such breakpoint \( \alpha_i \), its corresponding multiplier increase \( w_i \), and its corresponding constant term decrease \( \delta_i \). Note that each such pair of values \( (w_i, \delta_i) \) previously appeared (a factor of \( T \) larger due to expectation over types) as a corresponding pair of values \( (w_i^{(t)}, \delta_i^{(t)}) \) of one of the original functions \( U^{(t)} \). In particular, note that there can only be \( n \) distinct values of \( \delta_i \), as \( \delta_i^{(t)} \) does not depend on \( t \).

Next, consider \( \text{PROFIT}^{(t)}(\alpha) \). Recall that by the definition of \( \text{PROFIT}^{(t)} \), \( \text{PROFIT}^{(t)}(\alpha) = (1 - \alpha) \frac{d}{d\alpha} U^{(t)}(\alpha) \) (at a breakpoint we can arbitrarily choose the larger of the two derivatives). Since differentiation is linear, this property also holds for \( \text{PROFIT} \): \( \text{PROFIT}(\alpha) = (1 - \alpha) \frac{d}{d\alpha} U(\alpha) \). Observe that its maximum will occur when \( \alpha \) equals some breakpoint \( \alpha_i \), zero, or one, as \( \text{PROFIT} \) is piecewise linear. In fact, it cannot occur when \( \alpha \) is zero but not a breakpoint because there is no incentive for the agent to pick a non-null action, and it cannot occur when \( \alpha \) is one because the principal gives all reward to the agent. We now relate \( \text{PROFIT} \) to other quantities.

We define the cumulative terms \( W_i \triangleq \sum_{j=1}^i w_j \) and \( D_i \triangleq \sum_{j=1}^i \delta_j \). Since we have a dummy action, we know that for \( \alpha \in \{\alpha_i, \alpha_{i+1}\} \), \( U(\alpha) = W_i\alpha - D_i \) and therefore \( \text{PROFIT}(\alpha) = W_i(1 - \alpha) \). Since the latter is decreasing in \( \alpha \), in this range, it is maximized at \( \alpha = \alpha_i \) at which \( \text{PROFIT}(\alpha_i) = W_i(1 - \alpha_i) \).

What is \( \alpha_i \)? Well, consider the point \( \alpha = \alpha_i \), where some agent is ambivalent between two actions.
\[
W_i\alpha - D_i = W_{i-1}\alpha_i - D_{i-1}
\]
\[
w_i\alpha_i = \delta_i
\]
\[
\alpha_i = \frac{\delta_i}{w_i}
\]

Plugging this in, we get
\[
\text{PROFIT}(\alpha_i) = W_i \left( 1 - \frac{\delta_i}{w_i} \right).
\]

(2)

Remember that we need to compare this quantity to the welfare \( \text{Welf} = U(1) = W_B - D_B \). To make this comparison, we rewrite both \( \text{Welf} \) and Equation (2) in terms of a cumulative difference \( V_i \) instead of \( W_i \). Define the following differences:
\[
v_i \triangleq w_i - \delta_i
\]
\[
V_i \triangleq \sum_{j=1}^i v_j = W_i - D_i
\]

573
We can now write welfare as $\text{Welf} = V_B$ and our profit as:

$$\text{Profit}(\alpha_i) = (V_i + D_i) \left(1 - \frac{\delta_i}{w_i + \delta_i} \right)$$

$$= (V_i + D_i) \frac{w_i}{w_i + \delta_i}$$

$$= \frac{V_i + D_i}{1 + \delta_i/w_i}$$

$$\geq \min \left\{ V_i, D_i/v_i \right\}.$$  

It now suffices to show that there exists an $i$ such that $\min \left\{ V_i, D_i/v_i \right\} \geq V_B/O(n \log T).$ We do this as follows. First filter out all the $i$ such that $V_i < V_B/O(n \log T).$ The sum of the remaining $v_i$ is at least $(1 - 1/O(n \log T))V_B \geq \frac{1}{2}V_B.$ Now we simply need to show that there exists one $i$ such that

$$\frac{D_i}{\delta_i} v_i \geq \frac{1}{O(n \log T)} V_B.$$

Let's consider these multiplier terms $\beta_i = \frac{D_i}{\delta_i}. We would like to argue that they are quite large. To do so, we use the fact that there are at most $n$ possible values of $\delta_i.$ Specifically, let's group the remaining $i$'s together based on their original indices $i'$ (recall that each $\delta_i$ is just $\frac{1}{T} \delta_i(i').$ This creates at most $n$ groups of $i$'s, each with at most $T$ elements. We can group $i$'s together in an alternate way; let $S_k$ be the set of indices $i$ such that the number of earlier indices $j \leq i$ that derive from the same original index $i'$ is exactly $k$ (i.e., $\{|j \text{ s.t. } j \leq i \text{ and both } i, j \text{ have original index } i'\} = k$). We know the following:

(1) For all $k > T,$ $S_k = \emptyset$ (each original index $i'$ generates $T$ indices $i$).
(2) For all $k,$ $|S_k| \leq n$ (there are only $n$ original indices $i'$).
(3) For $i \in S_k,$ $\beta_i \geq k$ (since $D_i \geq k \delta_i$ at this point).

Let $\sigma_k = \sum_{i \in S_k} v_i.$ By (1), we know that $\sum_{k=1}^{T} \sigma_k \geq \frac{1}{2}V_B.$ It follows that there exists an $k$ such that $k \sigma_k \geq \frac{V_B}{O(n \log T)},$ if there weren't, then we would have $\sigma_k \leq o\left(\frac{V_B}{k \log T}\right)$ for all $k$ and summing over $k$ we would have that $\sum_{k=1}^{T} \sigma_k \leq o(1)V_B,$ a contradiction. By (2), it follows that there exists a $i \in S_k$ such that $kv_i \geq \frac{V_B}{O(n \log T)}.$ Finally, by (3) it follows that $\beta_i v_i \geq \frac{V_B}{O(n \log T)},$ as desired. This completes the proof. \hfill \Box

A closer examination of the above proof shows that the approximation ratio we obtain only depends on the multiset of the values $\delta_i$ (i.e., the differences between $c_{i+1} - c_i$ for consecutive costs). Let $S$ be this multi-set. Assume that $S$ has $U$ distinct values which appear with multiplicities $m_1, m_2, \ldots m_U$. Let $\Delta(c) = \sum_{i=1}^{U} \log m_i.$ Then a direct generalization of the proof of Theorem 6 gives the following, more general corollary.

**Corollary 5.** For any (possibly cost-varying) principal-agent problem $(c, F, r),$ let $S(c)$ be the multi-set of cost differences $\{c_{i+1}(t) - c_i(t) \mid i \in [n-1], t \in [T]\}$. Assume $S(c)$ has $U$ distinct values which appear with multiplicities $m_1, m_2, \ldots, m_U$. Then

$$\text{OPT-LINEAR} (c, F, r) \geq \Omega \left( \frac{1}{\Delta(c)} \right) \text{WELFARE} (c, F, r),$$

where $\Delta(c) = \sum_{i=1}^{U} \log m_i.$

This has a number of immediate consequences:
For a standard principal-agent problem with no constraints on the costs \( c, \Delta(c) \leq n \). This immediately implies the \( O(n) \)-approximation ratio proved in [8].

For a standard principal-agent problem where the costs \( c \), lie in an arithmetic sequence, \( \Delta(c) = O(\log n) \). In particular, if the set of costs is generated (for example) by discretizing a continuum of actions, we can hope for exponentially better approximation ratios than in the general case.

For a typed principal-agent problem with \( n \) actions and \( T \) types, \( \Delta(c) \leq O(n \log T) \). This recovers the result of Theorem 4 proved above.

For a typed principal-agent problem with \( n \) actions and \( T \) types whose costs \( c \) lie in an arithmetic sequence, \( \Delta(c) = O(\log(nT)) \).

For a typed principal-agent problem with \( n \) actions and \( T \) types whose costs \( c \) may vary between types, \( \Delta(c) = O(nT) \) (see the full version for a discussion of cost-varying typed principal-agent problems).

### 3.2 Linear Contracts: Tight Counterexample

We will now prove that the bound in Theorem 4 is tight. The following example follows almost immediately from the proof of Theorem 4; we choose a \( U(\alpha) \) to satisfy all necessary inequalities in the proof of Theorem 4, and then implement this \( U(\alpha) \) via a principal-agent problem.

**Theorem 6.** For all \( n, T > 0 \), there exists a principal-agent problem \((c, F, r)\) with \( T \) types, \( n + 1 \) actions, and two outcomes that satisfies

\[
\text{Opt-Line} \leq O\left(\frac{1}{n \log T}\right) \text{Welfare}(c, F, r).
\]

**Proof.** Consider the following principal-agent problem. As stated, we will have \( T \) types, \( n + 1 \) actions, and two outcomes. For this proof, we zero-index our actions (zero and one) and outcomes (zero through \( n \)) Action zero will have cost \( c_0 \neq 0 \), and action \( i \in [n] \) will have cost \( c_i = \sum_{j=1}^{i} \lambda^j \), where \( \lambda \) is a constant to be determined later (we will take \( \lambda \to \infty \)).

As for outcomes, outcome zero contributes reward \( r_0 = 0 \). Outcome one will contribute reward \( r_1 = c_n + \frac{n}{T \log T} \). For each type \( t \in [T] \), and action \( i \in \{0, \ldots, n\} \), define

\[
R_{i(t)} = c_i + \frac{i}{tn \log T}.
\]

Let \( p_{i(t)} = \frac{R_{i(t)}}{r_t} \). If type \( t \) plays action \( i \), then outcome one will occur with probability \( p_{i(t)} \) and outcome zero will occur with probability \( 1 - p_{i(t)} \). Note that the expected reward for the principal of type \( t \) playing action \( i \) is exactly \( R_{i(t)} \).

We begin by computing Welfare for this instance. Note that the contribution to welfare from type \( t \) is given by

\[
\max_{i} (R_{i(t)} - c_i) = \max_{i} \frac{i}{tn \log T} = \frac{1}{t \log T}.
\]

Averaging this over all types \( t \in [T] \), we have that

\[
\text{Welfare} = \frac{1}{T} \sum_{t=1}^{T} \frac{1}{t \log T} = \Theta\left(\frac{1}{T}\right).
\]

It thus suffices to show that \( \text{Profit}(\alpha) = O(1/nT \log T) \) for all \( \alpha \in [0, 1] \). We begin by computing the breakpoints \( \alpha_{i(t)} \) where the function \( \text{Profit}(\alpha) \) changes slope. These points satisfy \( \alpha_{i(t)} R_{i(t)} - c_i = \)
\[ \alpha^{(1)}_i R^{(t)}_{i+1} - c_{i+1}, \]  

so

\[ \alpha^{(t)}_i = \frac{c_{i+1} - c_i}{R^{(t)}_{i+1} - R^{(t)}_i} = \frac{\lambda^{i+1}}{\lambda^{i+1} + (nt \log T)^{-1}}. \]

For sufficiently large \( \lambda \), these \( \alpha^{(t)}_i \) are ordered first by type and then by action, i.e. \( \alpha^{(1)}_0 < \alpha^{(2)}_0 < \cdots < \alpha^{(T)}_0 < \alpha^{(1)}_1 < \cdots < \alpha^{(T)}_1 < \cdots < \alpha^{(T)}_{n-1} \). (One way to see this is to notice that \((\alpha^{(t)}_i)^{-1} = 1 + (nt \lambda^{i+1} \log T)^{-1}\); as long as \( \lambda > T \), this will satisfy the previous ordering). Now, for \( \alpha \in [\alpha^{(t)}_i, \alpha^{(t+1)}_i] \), note that the first \( t \) types are playing action \( i + 1 \) and the remaining types are playing action \( i \). It follows that

\[
\text{PROFIT}(\alpha) = \frac{1}{T} (1 - \alpha) \left( \sum_{s=1}^t R^{(s)}_{i+1} + \sum_{s=t+1}^T R^{(s)}_i \right) 
= \frac{1}{T} (1 - \alpha) \left( \sum_{s=1}^T R^{(s)}_i + \sum_{s=1}^t (R^{(s)}_{i+1} - R^{(s)}_i) \right) 
= \frac{1}{T} (1 - \alpha) \left( \sum_{s=1}^T (c_i + \frac{i}{sn \log T}) + \sum_{s=1}^t (\lambda^{i+1} + (sn \log T)^{-1}) \right) 
\leq \frac{1}{T} (1 - \alpha^{(t)}_i) \left( Tc_i + \frac{i}{n} + t\lambda^{i+1} + \frac{\log t}{n \log T} \right) 
\leq \frac{1}{T} (1 - \alpha^{(t)}_i) \left( T \left( \sum_{j=1}^t \lambda^j \right) + t\lambda^{i+1} + 2 \right) 
\leq \frac{1}{T} (1 - \alpha^{(t)}_i) (2t\lambda^{i+1}) \quad \text{for sufficiently large } \lambda 
= \frac{1}{T} \frac{(2nt \log T)^{-1}}{\lambda^{i+1} + (nt \log T)^{-1}(t\lambda^{i+1})} 
= \frac{1}{T} nt\lambda^{i+1} \log T + 1 
\leq \frac{2}{nT \log T}.
\]

A similar computation for \( \alpha \in [\alpha^{(T)}_i, \alpha^{(t+1)}_i] \) shows that in this case as well, \( \text{PROFIT}(\alpha) \leq \frac{2}{nT \log T} \). It follows that \( \text{PROFIT}(\alpha) \leq O(1/(n \log T)) \cdot \text{WELFARE} \), as desired.

\[ \square \]

4 FINE-GRAINED GAPS

In this section, we utilize Theorem 4 and Theorem 6 to get our worst-case bounds on the ratios between nearly all pairs of benchmarks.

We first discuss the easy side, namely upper-bounds on these ratios. From Theorem 4, we know that the expected profit of the optimal linear contract, OPT-LINEAR, can be at most \( O(n \log T) \) away from the maximum expected welfare, WELFARE. Since these are the lowest and highest benchmarks, respectively, we know that the ratio between any pair of benchmarks can be at most \( O(n \log T) \). The one pair where we obtain a better bound is between OPT-TYPE-AWARE and WELFARE. In this case, the problem decomposes by type. Since we can always just construct a linear contract for each type, the ratio is at most \( O(n) \). As a result, we need no additional machinery to obtain all our ratio upper-bounds.
We now discuss the hard side, namely lower-bounds on these ratios. On this side, we can use lower-bounds on the ratios between closer benchmarks to get lower-bounds on the ratio between further benchmarks. All the lower-bounds in Table 1 hence follow from lower-bounds on the ratios between adjacent benchmarks.

From Theorem 6, we know there is a two-outcome family of counterexamples which shows that the ratio between Opt-Linear and Welfare is $\Omega(n \log T)$. In Section 4.1, we show that in two-outcome principal-agent problems, the optimal menu of contracts cannot extract any more profit than the optimal linear contract. In Section 4.2, we show that we can edit any counterexample so that single contracts extract the entire welfare as profit but linear contracts do no better. In Section 4.3, we show that we can edit any counterexample so that type-aware principals extract the entire welfare as profit but menus do no better. Combining these ideas yields that the ratio between Opt-Linear and Opt-Single and the ratio between Opt-Menu and Opt-TypeAware are both $\Omega(n \log T)$.

This leaves two pairs of adjacent benchmarks. In the full version, we directly show that the ratio between Opt-Single and Opt-Menu is at least $(2 - \epsilon)$ for every constant $\epsilon > 0$. As for the ratio between Opt-TypeAware and Welfare, the problem again decomposes by type and it was already known that the gap between first-best and second-best outcomes was $\Omega(n)$ [8].

See Figure 1 for a helpful depiction of this plan.

Fig. 1. Proof road map of our upper and lower bound results between our various contract benchmarks.

4.1 The Two Outcome Case
In this subsection, we consider the case where there are only two outcomes ($m = 2$). Intuitively, the only purpose of the contract is to encourage the higher-valued outcome and hence it only makes
sense to offer zero on the lower-valued outcome and something on the higher-valued outcome. Given our assumption that the first reward $r_1$ is zero, such a contract is linear. In fact, we might hope that this reasoning applies to every contract in a menu and hence linear contracts are as powerful as menus in this scenario. Lemma 7 and its proof (found in the full version) formalize this intuition.

**Lemma 7.** Suppose we have a principal-agent problem $(c, F, r)$ with $m = 2$ outcomes. Then linear contracts can extract as much profit as menus of contracts:

$$\text{OPT-MENU}(c, F, r) = \text{OPT-LINEAR}(c, F, r).$$

Applying Lemma 7 to Theorem 6 yields the following corollary.

**Corollary 8.** For all $n, T > 0$, there exists a principal-agent problem $(c, F, r)$ with $T$ types, $n + 1$ actions, and two outcomes that satisfies

$$\text{OPT-MENU}(c, F, r) \leq O\left(\frac{1}{n \log T}\right) \text{WELFARE}(c, F, r).$$

### 4.2 Reduction: Nonlinearity Disparity

In this subsection, we give a transformation for (typed) principal-agent problems that helps the optimal single contract extract the entire welfare as profit without allowing linear contracts to extract more profit.

**Lemma 9.** Suppose we have a principal-agent problem $(c, F, r)$ with $T$ types, $n$ actions, and $m$ outcomes. We can construct another contract problem $(c', F', r')$ with $T$ types, $n$ actions, and $m + 2$ outcomes such that:

$$\text{OPT-SINGLE}(c', F', r') = \text{WELFARE}(c, F, r)$$

$$\text{OPT-LINEAR}(c', F', r') = \text{OPT-LINEAR}(c, F, r)$$

**Proof.** Our goal is to permit a single contract to extract the full welfare as profit but not allow linear contracts to extract any additional profit. We will achieve this by adding a new zero-reward outcome that can be used to encourage welfare-maximizing actions; note such linear contracts are blind to such outcomes. We also will need an additional zero-reward action to rebalance probabilities.

Zooming in, we want to arrange things so that a single contract that only offers a positive transfer on outcome $(m + 1)$ winds up perfectly paying the cost of the welfare-maximizing action. Note that the identity of this action changes with the agent’s type $t$; denote the identity of this action for type $t$ with $i^*(t)$, breaking ties arbitrarily. We will need the probability of outcome $(m + 1)$ to be proportional to the cost of action $i^*(t)$, since the expected transfers are this probability times the same base transfer amount. Let $\epsilon \equiv (2 \max_t c^*_t)^{-1}$, and let for all $t \in [T]$ let $\epsilon_t \equiv \epsilon c^*_t$ which is at most one-half by construction; this outcome will occur for the type $t$’s welfare-maximizing action with probability $\epsilon_t$.

We plan to dedicate one-half probability mass total to our two new outcomes, so outcome $m + 2$ will pick up the remaining probability. We also need to scale down the probability of existing outcomes by a factor of one-half so everything sums to one. To compensate for this probability scaling, we also scale up the rewards by a factor of two.
Welfare completely capture the new welfare \( t \) the maximum welfare of a type \( i \) expected to be transferred \( c \) distributions, because for any type \( t \) follows (see Table 3 for a depiction of this construction in tabular form).

This completes our construction. We now argue that the optimal single contract is able to completely capture the new welfare Welfare \( (c', F', r') \) from the nonlinearity disparity reduction. For each type \( t \in [T] \), we use outcome \( m + t \) as a way for single contracts to extract the entire welfare. Outcome \( m + T + 1 \) is serves to rebalance the total probability mass.

### Table 3. Resulting principal-agent problem \( (c', F', r') \) from the nonlinearity disparity reduction.

| Type \( t \in [T] \) | Outcome 1 \( r_1 \) | \( \ldots \) | Outcome \( t \) \( r_t \) | \( \ldots \) | Outcome \( m \) \( r_m \) | \( \ldots \) | Outcome \( m + 1 \) \( r_{m+1} \) | \( \ldots \) | Outcome \( m + 2 \) \( r_{m+2} \) |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| Action 1 \( c_1 \) | \( \frac{1}{2}F_{1,1}^{(t)} \) | \( \ldots \) | \( \frac{1}{2}F_{1,m}^{(t)} \) | \( \ldots \) | 0 | \( \ldots \) | 0 | \( \ldots \) |
| Cost \( c_2 \) | \( \frac{1}{2}F_{2,1}^{(t)} \) | \( \ldots \) | \( \frac{1}{2}F_{2,m}^{(t)} \) | \( \ldots \) | 0 | \( \ldots \) | 0 | \( \ldots \) |
| Action \( c_{i(t)} \) | \( \frac{1}{2}F_{i(t),1}^{(t)} \) | \( \ldots \) | \( \frac{1}{2}F_{i(t),m}^{(t)} \) | \( \ldots \) | \( \epsilon_t \) | \( \ldots \) | \( \frac{1}{2} - \epsilon_t \) | \( \ldots \) |
| Action \( n \) | \( \frac{1}{2}F_{n,1}^{(t)} \) | \( \ldots \) | \( \frac{1}{2}F_{n,m}^{(t)} \) | \( \ldots \) | 0 | \( \ldots \) | 0 | \( \ldots \) |

Combining the above ideas, we formally define our new principal-agent problem \( (c', F', r') \) as follows (see Table 3 for a depiction of this construction in tabular form).

\[
i \in [n] \quad c'_j \triangleq c_j \quad \forall t \in [T], i \in [n], j \in [m+2] \quad F'_{i,j}^{(t)} \triangleq \begin{cases} 
\frac{1}{2}F_{i,j}^{(t)} & \text{if } j \in [m] \\
\epsilon_t I(i = i_{t(t)}) & \text{if } j = m + 1 \\
\frac{1}{2} - \epsilon_t I(i \neq i_{t(t)}) & \text{if } j = m + 2 
\end{cases} \\
\forall j \in [m+2] \quad r'_j \triangleq \begin{cases} 
2r_j & \text{if } j \in [m] \\
0 & \text{if } j \in \{m + 1, m + 2\}
\end{cases}
\]

Note that we have correctly guaranteed that the new forecast tensor is made of valid probability distributions, because for any type \( t \in [T] \) and action \( i \in [n] \), the relevant row sums to one:

\[
\sum_{j=1}^{m+2} F'_{i,j}^{(t)} = \left[ \sum_{j=1}^{m} \frac{1}{2}F_{i,j}^{(t)} \right] + \left[ F'_{i,m+1}^{(t)} + F'_{i,m+2}^{(t)} \right] = 1
\]

This completes our construction. We now argue that the optimal single contract is able to completely capture the new welfare Welfare \( (c', F', r') \). Consider the contract which transfers \((1/\epsilon)\) on outcome \((m + 1)\): \( x = (1/\epsilon) \tilde{e}_{m+1} \). Under this contract, an agent that picks action \( i \) is expected to be transfered \( c_{i_{t(t)}} \) if \( i = i_{t(t)} \) and nothing otherwise. Since costs are nonnegative, this means that \( i_{t(t)} \) is among the actions that maximize the utility of a type \( t \) agent. Furthermore, since the principal only transfers the bare minimum, i.e. \( c_{i_{t(t)}} \), they keep \( R'_{i(t)} - c'_{i(t)} \). This is by definition the maximum welfare of a type \( t \) agent.

\[
\text{OPT-SINGLE} (c', F', r') = \text{Welfare} (c', F', r')
\]
By construction, it is easy to see that the welfare has not changed. We scaled down all probabilities by one-half but scaled up all rewards by two, so expected rewards are the same. Since costs have remained constant, the welfare of each action is the same. Hence we get one of our desired statements:

\[ \text{Welfare} (c', F', r') = \text{Welfare} (c, F, r) \]
\[ \text{Opt-Single} (c', F', r') = \text{Welfare} (c, F, r) \]

It remains to argue that linear contracts do not gain any profit. Since our two new outcomes have zero reward, they cannot be utilized by linear contracts. Hence, linear contracts are facing a rescaled version of the original problem, and the linear contract with transfer coefficient \( \alpha \in [0, 1] \) achieves the same profit in both \( (c', F', r') \) as it did in \( (c, F, r) \).

\[ \text{Opt-Linear} (c', F', r') = \text{Opt-Linear} (c, F, r) \]

This completes the proof.

Applying Lemma 9 to Theorem 6 yields the following corollary.

**Corollary 10.** For all \( n, T > 0 \), there exists a principal-agent problem \( (c, F, r) \) with \( T \) types, \( n + 1 \) actions, and four outcomes that satisfies

\[ \text{Opt-Linear} (c, F, r) \leq O \left( \frac{1}{n \log T} \right) \text{Opt-Single} (c, F, r) \]

### 4.3 Reduction: Information is Power

In this subsection, we give a transformation for (typed) principal-agent problems that enable a type-aware principal to extract the entire welfare as profit, while principals subject to adverse selection cannot extract much more profit than they could in the base problem. For this transformation, we will need the following theoretical tool, capturing the notion that very small additive perturbations to the agent’s utility cannot drastically alter the profit.

**Definition 11 (Cost-Stability).** We say that a principal-agent benchmark \( B \) is cost-stable if for every principal-agent problem \( (c, F, r) \) there exist constants \( s \geq 0, \tau > 0 \) such that for all cost perturbations \( c' \in \mathbb{R}^n, d_\infty (c, c') \leq \tau \), we have that

\[ B(c', F, r) \leq B(c, F, r) + sd_\infty (c, c') \]

Cost-stability, when applied to the \( \text{Opt-Single} \), is related to Dütting et al.’s notion of \( \delta \)-incentive compatible contracts [7]. Namely, we define cost-stability to measure how much a benchmark can change (increase) under these shifts, aiming to prevent contracts from suddenly becoming better when proving negative results. In contrast, \( \delta \)-incentive-compatibility allows a contract to slightly violate incentive compatibility, which allows for positive (approximation) results. In the full version, we prove the following result, namely that all five of our benchmarks are cost-stable. However, this subsection’s reduction only uses the cost-stability of \( \text{Opt-Menu} \) and \( \text{Opt-Single} \).

**Corollary 12.** The following principal-agent measures are all cost-stable: \( \text{Welfare}, \text{Opt-TypeAware}, \text{Opt-Menu}, \text{Opt-Single}, \text{and Opt-Linear} \).

We now present the main theorem of this subsection. We give a proof sketch here and defer the full proof to the full version.
THEOREM 13. Suppose we have a principal-agent problem \((c, F, r)\) with \(T\) types, \(n\) actions, and \(m\) outcomes and a constant \(\zeta > 0\). We can construct another principal-agent problem \((c', F', r')\) with \(T + 1\) types, \(n\) actions, and \(m + 2\) outcomes such that:

\[
\text{OPT-TYPEAWARE} (c', F', r') = \frac{T}{T+1} \text{WELFARE} (c, F, r)
\]

\[
\text{OPT-MENU} (c', F', r') \leq \frac{T}{T+1} \text{OPT-MENU} (c, F, r) + \zeta
\]

\[
\text{OPT-SINGLE} (c', F', r') \leq \frac{T}{T+1} \text{OPT-SINGLE} (c, F, r) + \zeta
\]

\[
\text{OPT-LINEAR} (c', F', r') = \frac{T}{T+1} \text{OPT-LINEAR} (c, F, r)
\]

PROOF SKETCH. We use a parameter \(\epsilon \in (0, 1)\) to represent the probability mass dedicated to our two new outcomes, which is chosen in the full proof to keep probabilities valid. One of the outcomes serves to allow type-aware principals to extract the entire welfare as profit, while the other outcome balances probability mass. We introduce a new type to punish a principal unaware of the type who tries to use either of these new outcomes. Doing so is merely expensive for them, not impossible, and so we apply Corollary 12 to show that barely using these two outcomes is not very powerful.

We give the formal definition of the resulting principal-agent problem \((c', F', r')\) here, but defer proving it has the desired properties to the full version.

\[
i \in [n] \quad c'_i \triangleq c_i
\]

\[
\forall t \in [T], i \in [n], j \in [m + 2] \quad F_{i,j}^{(t)} \triangleq \begin{cases} 
(1 - \epsilon) F_{i,j}^{(t)} & \text{if } j \in [m] \\
\epsilon I (i = i_j^{(t)}) & \text{if } j = m + 1 \\
\epsilon I (i \neq i_j^{(t)}) & \text{if } j = m + 2
\end{cases}
\]

\[
\forall i \in [n], j \in [m + 2] \quad F_{i,j}^{(T+1)} \triangleq \begin{cases} 
0 & \text{if } j \in [m] \\
\frac{1}{2} & \text{otherwise}
\end{cases}
\]

\[
\forall j \in [m + 2] \quad r_j' \triangleq \begin{cases} 
\frac{r_j}{1 - \epsilon} & \text{if } j \in [m] \\
0 & \text{if } j \in \{m + 1, m + 2\}
\end{cases}
\]

Applying Theorem 13 to Corollary 8 yields the following corollary.

COROLLARY 14. For all \(n, T > 0\), there exists a principal-agent problem \((c, F, r)\) with \(T + 1\) types, \(n + 1\) actions, and four outcomes that satisfies

\[
\text{OPT-MENU} (c, F, r) \leq O \left( \frac{1}{n \log T} \right) \text{OPT-TYPEAWARE} (c, F, r).
\]

REFERENCES

[1] Moshe Babaioff, Michal Feldman, and Noam Nisan. 2006. Combinatorial agency. In Proceedings of the 7th ACM conference on Electronic commerce. 18–28.

[2] Moshe Babaioff, Michal Feldman, and Noam Nisan. 2006. Mixed strategies in combinatorial agency. In International Workshop on Internet and Network Economics. Springer, 353–364.

[3] Moshe Babaioff, Michal Feldman, and Noam Nisan. 2009. Free-riding and free-labor in combinatorial agency. In International Symposium on Algorithmic Game Theory. Springer, 109–121.

[4] Felipe Balmaceda, Santiago R Balseiro, José R Correa, and Nicolás E Stier-Moses. 2016. Bounds on the welfare loss from moral hazard with limited liability. Games and Economic Behavior 95 (2016), 137–155.

[5] Patrick Bolton, Mathias Dewatripont, et al. 2005. Contract theory. MIT press.
[6] Nobel Prize Committee et al. 2016. Oliver Hart and Bengt Holmström: Contract Theory. Scientific Background on the Sveriges Riksbank Prize in Economic Sciences in Memory of Alfred Nobel (2016).

[7] Paul Dütting, Tim Roughgarden, and Inbal-Talgam Cohen. 2020. The complexity of contracts. In Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms. SIAM, 2688–2707.

[8] Paul Dütting, Tim Roughgarden, and Inbal Talgam-Cohen. 2019. Simple versus optimal contracts. In Proceedings of the 2019 ACM Conference on Economics and Computation. 369–387.

[9] Daniel Gottlieb and Humberto Moreira. 2015. Simple contracts with adverse selection and moral hazard. The Wharton School Research Paper 78 (2015).

[10] Sanford J Grossman and Oliver D Hart. 1992. An analysis of the principal-agent problem. In Foundations of Insurance Economics. Springer, 302–340.

[11] Chien-Ju Ho, Aleksandrs Slivkins, and Jennifer Wortman Vaughan. 2016. Adaptive contract design for crowdsourcing markets: Bandit algorithms for repeated principal-agent problems. Journal of Artificial Intelligence Research 55 (2016), 317–359.

[12] Jon Kleinberg and Robert Kleinberg. 2018. Delegated search approximates efficient search. In Proceedings of the 2018 ACM Conference on Economics and Computation. 287–302.

[13] Jean-Jacques Laffont and David Martimort. 2009. The theory of incentives: the principal-agent model. Princeton university press.

[14] Jean-Jacques Laffont and Jean Tirole. 1986. Using cost observation to regulate firms. Journal of Political Economy 94, 3, Part 1 (1986), 614–641.

[15] Shenke Xiao, Zihe Wang, Mengjing Chen, Pingzhong Tang, and Xiwang Yang. 2019. Optimal common contract with heterogeneous agents. arXiv preprint arXiv:1911.04146 (2019).