From Dynamic Pricing to Dynamic Principal-Agent Problems: Going Beyond the No Learning Theorem*

Minbiao Han

University of Chicago

minbiaohan@uchicago.edu

Abstract

This paper studies dynamic principal-agent problems, i.e., games in which a principal and an agent repeatedly interact, where the agent's type is unknown and the agent is non-myopic. A natural question to ask is, can the principal learn the optimal strategy against the unknown agent through these repeated interactions? The No Learning Theorem from dynamic pricing, a special class of dynamic principal-agent problems, would suggest that the principal cannot learn effectively from the agent. In contrast, we demonstrate, for the first time, that in general principal-agent problems, the principal can improve her utility through learning in repeated plays. We show that this dynamic policy continues to be nearly optimal even when allowing for the principal to have a larger static strategy space, specifically if we permit communication between the principal and agent. We also provide an algorithm based on a novel and compact mixed-integer linear program for finding the principal's optimal dynamic policy. In addition, we develop an algorithm to compute a Markovian policy for the principal that approximates the optimal dynamic policy while allowing for more efficient computation. Through simulations, we examine the efficiency, compared to static policies, and the runtime of the proposed algorithms. Lastly, we apply the generalized principal-agent framework to a specific contract design problem. We show that with the special structural properties of contract design, the optimal dynamic principal policy has a compact representation. Furthermore, in the case where the agent's type is known, dynamic principal policies enable full surplus extraction from the agent when the interaction with the principal extends over a sufficiently long time horizon.

1 Introduction

Optimal pricing given unknown demand is a well-studied problem that comes up in many settings [Aggarwal et al., 2006, Varian, 2009, Vickrey, 1961]. A natural extension of the problem is to assume that there is an opportunity for a seller to learn the demand through repeated interactions using *dynamic pricing*. However, even in the simplest setting, a single buyer with a fixed valuation drawn from a known distribution (i.e., the *fishmonger's problem*), it is impossible for a seller to exploit any information learned to improve her revenue. Formally, no dynamic pricing policy can outperform the fixed (or *static*) policy of simply offering the optimal single round price, i.e., the *Myerson price* [Myerson, 1981], at every round [Devanur et al., 2014, Vanunts and Drutsa, 2019], at least when the buyer is both strategic and patient. This negative result, the *No Learning Theorem*, is a function of the buyer behaving strategically in his purchase decisions in order to influence future prices. The effect on the literature of this strong negative result has been to adopt assumptions that weaken the strategic behavior of the buyer [Amin et al., 2013, 2014, Dawkins et al., 2021, 2022, Immorlica et al., 2017, Vanunts and Drutsa, 2019]. A natural question arises from this impossibility result, "Is a static

[*] Joint work with Quinlan Dawkins (nod.ai), Michael Albert (UVA), and Haifeng Xu (UChicago)

policy always optimal in a dynamic setting with strategic agents?" Or stated another way, "Can one both learn and exploit the information learned when facing strategic agents if we move beyond the fishmonger's problem?"

A natural generalization of the fishmonger's problem is a principal-agent problem with uncertainty over agent types. Specifically, principal-agent problems model a two-step sequential decision-making process between two players, a principal, and an agent. The principal moves first by committing to a randomized strategy. Then the agent acts after observing the principal's strategy; generally, it is assumed that the agent best responds to the observed principal strategy by playing the utility-maximizing action, i.e., the *Strong Stackelberg Equilibrium* (SSE). This canonical model for strategic principal-agent interactions has been adopted for many applications in the real world, such as contract design theory, security resource allocation, and optimal traffic routing [Bolton and Dewatripont, 2004, Paruchuri et al., 2008, Roth et al., 2016, Roughgarden, 2001, Yang et al., 2014]. The SSE can be easily solved if the principal has access to the agent's utility information [Conitzer and Sandholm, 2006]. However, in practice, this utility information is usually private and unknown to the principal, and instead, the principal may only have a distribution over possible agent types. Therefore, in a repeated setting, similar to the fishmonger's problem, there is an opportunity to learn from the behavior of the agent.

In this work, we demonstrate, in a general repeated principal-agent problem with an unknown agent type, that it is indeed possible to improve the principal's utility using a dynamic policy in a repeated principal-agent problem with unknown agent types relative to *any* static policy. This indicates that the No Learning Theorem is not fundamental to learning against strategic agents, and it is instead specific to the fishmonger's problem. We further demonstrate that this policy is provably nearly optimal even relative to stronger static strategy spaces for the principal that allow for such things as communication between the principal and agent. Through experiments on both several standard principal-agent problems and randomly generated bimatrix principal-agent games, we show that the dynamic policies meaningfully outperform the optimal static policy. However, the computation of the optimal dynamic policy for the principal is NP-Hard in general. Therefore, we additionally propose a novel Markovian principal policy, where the principal's strategy only depends on the agent's response at the previous round. We demonstrate experimentally that the Markovian principal policy both approximates well the optimal dynamic principal policy and is significantly more computationally efficient. Finally, we apply our generalized principal-agent problem to the well-studied contract design problem. We show the unique structural properties of contract design allow for a compact representation of the optimal dynamic principal policy. In addition, when the agent's type is known and the principal engages in a long-term interaction, this policy empowers the principal to fully extract the surplus from the agent, maximizing their gains.

1.1 Related Work

This work is closely related to the literature on learning in Stackelberg (security) games. When the agent's payoff information is unknown, [Balcan et al., 2014, Letchford et al., 2009, Peng et al., 2019, Roth et al., 2016] propose various policies that allow a principal to learn the agent's utility or optimal principal policy by observing the agent's response to a particular principal's strategy. These policies can learn the optimal principal strategy efficiently with a polynomial number of learning rounds. This literature assumes that the agent behaves *myopically*, i.e., the agent always best responds to the posted principal strategy without regard for future rounds. Recently, Haghtalab et al. [2022] studied learning in Stackelberg games with non-myopic agents and proposed a no-regret learning policy for the principal. However, their work relies on the assumption that the agent discounts the future utility at a greater rate than the principal. In a slightly different direction, [Deng et al., 2019] studies dynamic policy design in a repeated setting when the agent's payoff information is public but instead of best responding the agent follows a no-regret learning algorithm to interact with the principal.

Additionally, this work is related to the literature on learning optimal prices [Amin et al., 2013, 2014, Dawkins et al., 2021, 2022, Immorlica et al., 2017, Vanunts and Drutsa, 2019]. When the buyer discounts the future at a greater rate than the seller, there are positive results, e.g. there exists a no-regret learning policy to learn the optimal price [Amin et al., 2013]. If the buyer does not discount future utility or if the buyer is more patient than the seller, the optimal pricing policy is to post a constant price for every round [Devanur et al., 2014, Vanunts and Drutsa, 2019].

Our work is also related to the algorithmic contract design problem. Most of the previous works have focused on the computational issues of contract design [Alon et al., 2021, Castiglioni et al., 2021, 2022, Dütting et al., 2019]. In recent studies, the sample complexity of online contract design has been explored by researchers such as Ho et al. [2014], Zhu et al. [2022]. These studies investigate the number of samples required to effectively design contracts in an online setting. Cohen et al. [2022] further extend the analysis to a specific scenario where the agent's utility function exhibits bounded risk aversion.

Other related work considers dynamic policy design in extensive-form games Hart [1992], which is an extremely well-studied problem. One well-known solution concept for extensive-form games is the subgame perfect equilibria Fudenberg and Levine [1983], Moore and Repullo [1988]. We remark that our problem can be thought of as a stateless dynamic principal-agent problem with incomplete information when the principal has the ability to commit. This commitment assumption is very often used in dynamic principal-agent problems ranging from dynamic mechanism design Kakade et al. [2013], Pavan et al. [2014] to dynamic Stackelberg games Lauffer et al. [2022], Li and Sethi [2017]. When the principal has the ability to commit, we do not consider the concept of subgame perfect equilibria as in classic extensive-form games.

Recent work has also considered the computation of Stackelberg equilibria in stochastic games Shapley [1953]. The canonical solution concept for such games is the Markov perfect Stackelberg equilibria Maskin and Tirole [2001], which is a *stationary* equilibrium concept. One key difference between this line of work Bensoussan et al. [2015], Chang et al. [2015], Goktas et al. [2022], Vorobeychik and Singh [2012], Vu et al. [2022] and ours is that these all study Markov stationary policies. In contrast, our work studies non-stationary principal policies for non-stochastic games.

Finally, our setting is different from the traditional bandit setting in a very significant way. In a bandit problem, the arm does not behave strategically Slivkins et al. [2019]. In our setting, and this is the difficulty that drives results like the No Learning Theorem, the agent can behave strategically to throw off the principal. So, traditional no-regret learning results do not apply. The No Learning Theorem states that for at least some settings (since the Fishmonger's problem is a sub-problem of our setting) the regret is unbounded for any learning algorithm.

2 Preliminaries and Problem Setup

Principal-Agent Problems. The principal-agent problem is a game played by two players, who are referred to as the *principal (she)* and the *agent (he)*. The agent has a finite action set $[n] = \{1, \cdots, n\}$, while the principal's strategy space $\mathcal{X} \subseteq \mathbb{R}^m$ can be any closed convex set, which may have infinite size. This is because in practice, the principal has the privilege to move first by committing to a *randomized* strategy, while the agent responds with a pure action.[1] The utility functions of the principal and the agent are denoted as U and V, respectively, where $u/v : \mathcal{X} \times [n] \to \mathbb{R}$. Throughout this paper, we assume the agents' utility functions are linear with respect to the principal strategy $\boldsymbol{x} \in \mathcal{X}$ and agent action $j \in [n]$, which is correct in many specialized principal-agent problems. We explicitly write $U(\boldsymbol{x}, j) = \langle \boldsymbol{x}, \boldsymbol{u}_j \rangle + \alpha_j$ and $V(\boldsymbol{x}, j) = \langle \boldsymbol{x}, \boldsymbol{v}_j \rangle + \beta_j$ where $\boldsymbol{u}_j/\boldsymbol{v}_j \in \mathbb{R}^m, \alpha_j/\beta_j \in \mathbb{R}$ are utility parameters.

In a single round (i.e., static) principal-agent problem setting, the principal moves first by committing to a *randomized strategy* $\boldsymbol{x} \in \mathcal{X}$. After observing the principal strategy $\boldsymbol{x}$, the agent responds by playing some action j which leads to an expected agent utility $V(\boldsymbol{x}, j)$. A rational agent will pick the optimal action

[1] Since the agent moves after the principal, there is no need for the agent to randomize his strategy. If there are ties, we assume the agent breaks the tie in favor of the principal, which is without loss of generality [Von Stengel and Zamir, 2004].

$j^*(\boldsymbol{x}) = \arg\max_{j\in[n]} V(\boldsymbol{x}, j)$ to maximize his own utility. The corresponding principal utility is denoted as $U(\boldsymbol{x}, j)$. When the agent's utility function V is fully known to the principal, the principal can predict the agent's reaction $j^*(\boldsymbol{x})$ to any $\boldsymbol{x}$. The rational principal thus will commit to $\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}\in\mathcal{X}} U(\boldsymbol{x}, j^*(\boldsymbol{x}))$. This bi-level optimization problem can be solved in $\text{poly}(m, n)$ time via a linear programming approach [Conitzer and Sandholm, 2006], and the optimal $\boldsymbol{x}^*$ is the *Strong Stackelberg Equilibrium* (SSE).

Bayesian Principal-Agent Problems. In many settings of interest, the principal may not know the agent's utility function V. To capture the principal's uncertainty about agent payoffs, we follow the literature and model the uncertainty as a random *agent type* $\theta \in \Theta$ which is known privately to the agent while the principal only has a prior distribution $\boldsymbol{\mu} \in \Delta^{|\Theta|} = \{\boldsymbol{\mu} : \sum_{\theta\in\Theta} \mu(\theta) = 1\}$ over the types. We denote the utility function of a θ-type agent as V^θ, and the best response for agent type θ to any principal strategy $\boldsymbol{x}$ is $j^{*\theta}(\boldsymbol{x}) = \arg\max_{j\in[n]} V^\theta(\boldsymbol{x}, j)$. As a natural extension of SSE to this Bayesian setup, a principal with prior knowledge $\boldsymbol{\mu}$ will play an $\boldsymbol{x}^*$ to maximize her expected utility, formally, $\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}\in\mathcal{X}} \sum_{\theta\in\Theta} \mu(\theta) U^\theta(\boldsymbol{x}, j^{*\theta}(\boldsymbol{x}))$, known as the Bayesian Stackelberg Equilibrium (BSE). Unlike the SSE, computing a BSE is NP-hard [Conitzer and Sandholm, 2006]. For notation convenience, we drop the principal utility's dependence on the agent type θ and simply use U instead of U^θ. All results generalize trivially.

Dynamic Principal-Agent Problems. We consider dynamic principal-agent problems, which (1) generalize Bayesian principal-agent problems by allowing repeated principal-agent interactions; and (2) also generalize dynamic pricing by allowing general player payoff functions. Formally, a principal-agent game is played repeatedly for T rounds. The agent has a fixed private type θ drawn from $\boldsymbol{\mu}$. The principal plays a *Dynamic Stackelberg Policy*[2] (DSP) π which specifies a principal strategy $\boldsymbol{x}^t = \pi(\boldsymbol{j}_{t-1}) \in \mathcal{X}$ at each round t, where $\boldsymbol{j}_{t-1} = (j_1, \cdots, j_{t-1})$[3] is the agent's past responses. The principal commits to a DSP π before the game starts and the agent observes the policy in advance. This commitment assumption is common in the literature, e.g. the fishmonger's problem Devanur et al. [2014], Vanunts and Drutsa [2019], dynamic mechanism design [Amin et al., 2013, Mirrokni et al., 2020, Pavan et al., 2014] and Stackelberg security games [Sinha et al., 2018]. We call the optimal DSP the *Dynamic Stackelberg Equilibrium* (DSE).

Learning in Dynamic Principal-Agent Problems. The principal is not primarily concerned with identifying the agent type, as in the traditional learning paradigm. Instead, the principal seeks to maximize her utility over the rounds of the game. However, to do this, the principal may take advantage of the *revealed preferences* [Beigman and Vohra, 2006, Roth et al., 2016] of the agent, so that she can distinguish the agent type and tailor her strategies accordingly. As a byproduct of such optimization, the optimal dynamic policy π may learn and exploit the agent type. Formally, the principal learns *effectively* if there exists θ, θ' and t such that $\pi(\boldsymbol{j}^{*\theta}_{t-1}) \neq \pi(\boldsymbol{j}^{*\theta'}_{t-1})$. From this perspective, the DSE as the optimal dynamic policy is a concept that combines learning with utility optimization.

3 From Pricing to General Principal-Agent Problems

A key motivation for this work is the widely known folk theorem, commonly referred to as the *No Learning Theorem* [Devanur et al., 2014, Vanunts and Drutsa, 2019], for the dynamic pricing game with a single buyer with fixed valuation. Formally, a seller (she) repeatedly sells an item to the same buyer (he) for T rounds. The buyer has a private value $v \in \mathbb{R}_+$ for the item, which is drawn from some prior distribution $\boldsymbol{\mu}$ before the game starts and is fixed throughout the game. The seller knows the prior distribution $\boldsymbol{\mu}$ and can post a price $p_t \in \mathbb{R}_+$ at each round t; the buyer responds with $j_t \in \{0, 1\}$, indicating accepting the price and buying the item ($j_t = 1$) or not. The buyers value is $j_t(v - p_t)$ for round t. Before this game starts, the seller commits to a *dynamic pricing policy* π that maps any buyer's past responses to a price at the current round t, i.e., $p_t = \pi(j_1, \cdots, j_{t-1})$.

[2] Following conventions in sequential decision making, we use "policy" to denote a dynamic scheme that maps history to a strategy, whereas "strategy" is only used for one-round interactions.

[3] We use bold $\boldsymbol{j}_t$ to denote a vector throughout the paper, where the subscript t represents the vector's length.

One potential policy is to repeatedly post the Myerson price [Myerson, 1981], $p^* = \arg\max_p[p(1 - \Pr(v \leq p))]$, which maximizes the single-round revenue with respect to the seller's prior knowledge $\boldsymbol{\mu}$. This policy would seem to be highly sub-optimal – for example if the buyer rejects the item in the first round, he will reject it in all future rounds, leading to a revenue of zero. One might conjecture that the seller should be able to gradually learn the exact v from repeatedly observing the buyer's responses, and then set the price to v, extracting maximum revenue in all future rounds. However, the no learning theorem shows that, due to the buyer's strategic responses to seller's learning, it is *impossible* for the seller to achieve higher expected revenue using any other strategy [Vanunts and Drutsa, 2019]!

3.1 Learning in Dynamic Principal-Agent Problems

Dynamic pricing is a special case of general dynamic principal-agent problems with the seller as the principal and the buyer as the agent. Given the optimality of static pricing due to the No Learning Theorem, it becomes very natural to ask the following research question:

Is it possible to learn from a strategic agent in general dynamic principal-agent problems?

As the following example illustrates, it is indeed possible to learn from a strategic agent.

U	j_0	j_1
i_0	5	2
i_1	5	7

V^0	j_0	j_1
i_0	5	2
i_1	4	2

V^1	j_0	j_1
i_0	5	7
i_1	4	3

Table 1: A bimatrix principal-agent game instance with two actions for the principal and agent. The principal's utility is defined by the first subtable U; the agent has two possible types, with utility matrix V^0, V^1 respectively, each having equal probability 0.5. The principal's strategy is a randomized strategy $\boldsymbol{x} \in \mathcal{X} = \Delta^2$, which denotes the probabilities of playing two leader actions i_0 and i_1. The leader's utility $U(\boldsymbol{x}, j) = \langle \boldsymbol{x}, U[:, j] \rangle$, where $U[:, j]$ represents the j'th column of the utility matrix. The agent's utility function if defined similarly.

Dynamic policies outperform the optimal static policy. Consider a Bayesian principal-agent problem with payoffs listed in Table 1. As can be computed, the optimal *static* policy for the principal in this game turns out to be the (BSE) principal strategy $\boldsymbol{x} = (\frac{1}{3}, \frac{2}{3})$, i.e., the principal commits to a randomized strategy which plays action i_0 with probability $\frac{1}{3}$ and i_1 with probability $\frac{2}{3}$.[4] This BSE strategy results in an expected principal utility of $5\frac{1}{6}$.

Consider the simplest possible dynamic setup with *two* rounds of principal-agent interactions. The optimal dynamic principal policy is to play $\boldsymbol{x}^1 = (0, 1)$ in the first round. If the agent responds with j_0 in the first round, the principal still plays $\boldsymbol{x}^2 = (0, 1)$ in the second round; otherwise, the principal plays $\boldsymbol{x}^2 = (1/2, 1/2)$. Agent V^0 best responds by playing j_0 in both rounds, i.e., $\boldsymbol{j}_2^{*0} = (j_0, j_0)$, while V^1 plays j_1 in both rounds, i.e., $\boldsymbol{j}_2^{*1} = (j_1, j_1)$. This implies that the principal plays different strategies against the two types in the second round, i.e $\pi(j_1^{*0}) \neq \pi(j_1^{*1})$; the optimal policy requires that the principal learns about the agent's type. Moreover, this dynamic policy results in a total principal utility of $10\frac{3}{4}$, averaged to $5\frac{3}{8}$ per round, which is strictly larger than the static optimal principal utility of $5\frac{1}{6}$.

The optimal dynamic policy also outperforms the principal's static strategy with complete knowledge. Specifically, suppose the principal can observe the agent's type before the game and can play the optimal static strategy against each type (i.e., the SSE), which is $\boldsymbol{x} = (1, 0)$ against V^0 and $\boldsymbol{x} = (\frac{1}{3}, \frac{2}{3})$ against V^1.

[4] The BSE is actually not the optimal static principal strategy in general. In this example, the BSE happens to be optimal, simplifying the discussion. We will discuss the optimal single-round equilibrium concept in Section 5.

In this case, the principal obtains expected utility 5 against agent type V^0 and $5\frac{1}{3}$ against V^1, both of which are less than the averaged dynamic utility $5\frac{3}{8}$. This is because dynamic strategies, with commitment, are intrinsically more powerful than static strategies.

This observation provides some intuition as to why learning is valuable in general principal-agent problems and not in the pricing game. A key insight in the proof of the no learning theorem is that the seller can implement, in a single round, a *mechanism*, i.e., a price and a probability that she gives (or *allocates*) the item to the buyer, that generates the average revenue of any dynamic policy. Specifically, the single round simulation of the dynamic pricing policy takes the average of the offered prices and the average of the accept decisions, i.e., the probability that the buyer accepts the item over a randomly chosen round. This is possible to implement with a single round mechanism because the seller can set an arbitrary price and, crucially, an arbitrary allocation probability. We can upper bound this single round mechanism with the well-known Myerson optimal revenue [Myerson, 1981], and we can then upper bound the performance of any dynamic policy. If the seller can achieve the upper bound with a static policy in the pricing game, which is true when she can set the Myerson price, then a static policy is optimal.

However, in the general principal-agent problem, there is not always a natural corresponding mechanism with something as simple as price and allocation given the broader action space of the agent. Moreover, even if there is an imaginable action that is somehow globally optimal, in the same way as the Myerson optimal price, the principal's action space may not contain the optimal action. In fact, as we show in Section 4.3, even a slight modification of the pricing game where we remove the Myerson optimal price from the principal's action set leads to the optimal pricing policy being a dynamic policy. In a general principal-agent problem, the principal must use dynamic policies to overcome the limitations of all static policies.

4 The Possibility of Learning in Dynamic Principal-Agent Problems

Next, we present a full discussion of how the no learning theorem fails for general dynamic principal-agent problems. To begin with, let us revisit the no learning theorem in dynamic pricing games.

4.1 The No Learning Theorem in Dynamic Pricing

The key technique for proving the optimality of static constant pricing is an elegant reduction from dynamic pricing interactions to a single-round feasible auction mechanism Myerson [1981][5]. Specifically, for any pricing policy π, we construct an auction mechanism M_π that works as follows. First, the seller asks the buyer to report his value v and simulates $\boldsymbol{j}^*$ based on the reported v, denoted as:

$$\boldsymbol{j}^*(v) = \underset{\boldsymbol{j}\in\{0,1\}^T}{\operatorname{argmax}} \sum_{t\in[T]} j_t \cdot \big(v - \pi(j_1, \cdots, j_{t-1})\big). \tag{1}$$

Then the seller randomly chooses a round $t \in [T]$ with probability $1/T$ and allocates the item to the buyer with price $\pi\big(j_1^*(v), \cdots, j_{t-1}^*(v)\big)$ if $j_t^*(v) = 1$.

Proposition 1 (Proposition A.1 Vanunts and Drutsa [2019])**.** *The auction mechanism M_π is feasible and its expected revenue is equivalent to the dynamic pricing policy π's average expected revenue over T.*

Proof. To begin with, we formally define the auction mechanism M_π's allocation probability Q_M and payment P_M with respect to a buyer report v.

$$Q_M(v) = \sum_{t=1}^{T} \frac{j_t^*(v)}{T} \quad \text{and} \quad P_M(v) = \sum_{t=1}^{T} \frac{j_t^*(v)\pi\big(j_1^*(v), \cdots, j_{t-1}^*(v)\big)}{T} \tag{2}$$

To prove M_π is a feasible auction mechanism, we prove it satisfies incentive compatibility, individual rationality, and $Q_M(v) \leq 1$.

[5] Details about feasible auction mechanisms can be found in Section 3 of Myerson [1981]

First, because $j_t^*(v) \in \{0, 1\}$, we have that $Q_M(v) \leq 1$.

Next, we prove the individual rationality of the mechanism, i.e., the buyer's expected utility $v \cdot Q_M(v) - P_M(v) \geq 0$. By definition,

$$v \cdot Q_M(v) - P_M(v) = \frac{1}{T} \sum_{t=1}^{T} j_t^*(v) \big(v - \pi\big(j_1^*(v), \cdots, j_{t-1}^*(v)\big)\big). \tag{3}$$

In other words, we need to prove $\sum_{t=1}^{T} j_t^*(v) \big(v - \pi\big(j_1^*(v), \cdots, j_{t-1}^*(v)\big)\big)$, which is exactly the buyer's optimal utility, is greater than or equal to 0. From the definition of $\boldsymbol{j}^*(v)$ in equation (1), this is trivially satisfied, since 0 is the buyer's utility when setting $j_t = 0$, $\forall t$.

Finally, we prove the incentive compatibility of the mechanism

$$v \cdot Q_M(v) - P_M(v) \geq v \cdot Q_M(u) - P_M(u). \tag{4}$$

It is equivalent to prove

$$\sum_{t=1}^{T} j_t^*(v) \big(v - \pi\big(j_1^*(v), \cdots, j_{t-1}^*(v)\big)\big) \geq \sum_{t=1}^{T} j_t^*(u) \big(v - \pi\big(j_1^*(u), \cdots, j_{t-1}^*(u)\big)\big), \tag{5}$$

which is correct by the definition of $\boldsymbol{j}^*(v)$ from (1). As a result, we have proved that the mechanism M_π is a feasible mechanism.

Last but not least, note that M_π's expected revenue is the expected payment,

$$P_M(v) = \frac{1}{T} \sum_{t=1}^{T} j_t^*(v) \pi\big(j_1^*(v), \cdots, j_{t-1}^*(v)\big) \tag{6}$$

where $\sum_{t=1}^{T} j_t^*(v) \pi\big(j_1^*(v), \cdots, j_{t-1}^*(v)\big)$ is exactly the seller's expected revenue when running dynamic policy π for T rounds, proving the proposition. □

As a result, we have M_π as a feasible mechanism for any dynamic pricing policy π and the seller's revenue must be bounded by the Myerson revenue Myerson [1981]:

$$\frac{1}{T} \sum_{t \in [T]} j_t^*(v) \pi\big(j_1^*(v), \cdots, j_{t-1}^*(v)\big) \leq MyersonRev, \forall \pi$$

Note that $\sum_{t \in [T]} j_t^*(v) \pi\big(j_1^*(v), \cdots, j_{t-1}^*(v)\big)$ is exactly the seller's total revenue in T rounds when she uses dynamic pricing policy π. Hence, we have shown that *any* dynamic pricing policy's revenue is upper bounded by the revenue of running constant Myerson price for T rounds.

4.2 Why the No Learning Theorem Ceases to Hold in Principal-Agent Problems?

A natural question to ask is why does this argument fail in a general principal-agent problem? Note the key step to proving the no learning theorem of the pricing game is the connection between the constructed feasible auction mechanism M_π's revenue and the dynamic pricing policy π's revenue (i.e., (6)). Specifically, for every round $t \in [T]$ that is sampled by the auction mechanism M_π with probability $\frac{1}{T}$, the seller get revenue $j_t^*(v)\pi\big(j_1^*(v), \cdots, j_{t-1}^*(v)\big)$ with allocation probability $j_t^*(v)$ and payment $\pi\big(j_1^*(v), \cdots, j_{t-1}^*(v)\big)$. In other words, the seller can *impose* the action $j_t^*(v)$ on the buyer with the allocation rule in the constructed auction mechanism, which results in exactly the same revenue the seller would get at the round t when interacting with the buyer using the dynamic policy π.

However, for a general principal-agent problem, an important distinction from traditional auction mechanisms is the feature known as moral hazard or hidden action [Grossman and Hart, 1992, Holmström, 1979]: the action *cannot* be imposed on the agent like the allocation rule in an auction. Specifically, in a general principal-agent problem, the principal can only play a strategy (like the payment rule) and recommend an action (instead of imposing an action like the allocation rule) to the agent. But whether or not the agent adopts the recommended action depends on the incentives induced by the principal's strategy. For example, consider the same reduction from dynamic principal-agent interactions using π to a single round mechanism M_π. Similarly, the principal can ask the agent to report a type θ and simulate the agent's response:

$$\boldsymbol{j}^*(\theta) = \operatorname*{argmax}_{\boldsymbol{j}\in[n]^T} \sum_{t\in[T]} V^\theta\big(\pi(j_1,\cdots,j_{t-1}), j_t\big)$$

Then M_π samples a round $t \in [T]$ with probability $\frac{1}{T}$ and plays principal strategy $\pi\big(j_1^*(\theta),\cdots,j_{t-1}^*(\theta)\big)$. The important distinction is that the principal can not impose action $j_t^*(\theta)$ on the agent as the allocation rule does in an auction mechanism, and therefore, the principal can not obtain the same utility (i.e., $U\big(\pi\big(j_1^*(\theta),\cdots,j_{t-1}^*(\theta)\big), j_t^*(\theta)\big)$) as the dynamic policy π's utility at round t. On the contrary, a rational agent would be incentivized to play

$$j_{M_\pi,t}^*(\theta) = \operatorname*{argmax}_{j\in[n]} V^\theta\big(\pi\big(j_1^*(\theta),\cdots,j_{t-1}^*(\theta)\big), j\big),$$

which is not necessarily equivalent to $j_t^*(\theta)$, i.e., agent's best response with respect to $\pi\big(j_1^*(\theta),\cdots,j_{t-1}^*(\theta)\big)$ in a T rounds repeated interactions. Because the agent also needs to consider his response's influence on the principal's strategy in future rounds in repeated interactions. As a result, the principal's expected utility of M_π would be

$$\frac{1}{T}\sum_{t=1}^{T} U\big(\pi\big(j_1^*(\theta),\cdots,j_{t-1}^*(\theta)\big), j_{M_\pi,t}^*(\theta)\big), \tag{7}$$

which is not comparable to the principal's utility of running a dynamic policy π

$$\sum_{t=1}^{T} U\big(\pi\big(j_1^*(\theta),\cdots,j_{t-1}^*(\theta)\big), j_t^*(\theta)\big), \tag{8}$$

since $j_{M_\pi,t}^*(\theta) \neq j_t^*(\theta)$. And we remark again the reason is that the principal can not impose action $j_t^*(\theta)$ on the agent like the allocation rule in an auction mechanism.

4.3 An Example for the Failure of No Learning

Note that a pricing game can be considered a specific principal-agent problem. For example, let us consider a pricing game where the seller sells a single item to a buyer, whose value v over the item is drawn uniformly over $v \sim V = \{8, 35, 96\}$ (three arbitrarily generated numbers). It is easy to compute that the optimal Myerson price in this setting is 96, which gives the seller an expected revenue of $\mathbf{32}$.

We can also write the utility matrices of the seller and the buyer in this example as follows. It can be verified with the approach we proposed in Section 6 that the optimal dynamic seller policy is to set the

U	j_0	j_1
i_0	0	8
i_1	0	35
i_2	0	96

V^0	j_0	j_1
i_0	0	0
i_1	0	-27
i_2	0	-88

V^1	j_0	j_1
i_0	0	27
i_1	0	0
i_2	0	-61

V^2	j_0	j_1
i_0	0	88
i_1	0	61
i_2	0	0

Table 2: A pricing game example where U represents the seller's utility matrix, and V^0, V^1, V^2 represents the buyer's utility matrices when his type is $v = 8$, $v = 35$, and $v = 96$ correspondingly. In addition, i_0, i_1, i_2 represents setting a price of $8, 35, 96$, while j_0/j_1 represents the buyer rejects/accepts the price.

Myerson price 96 at every round, which gives us the same revenue as the optimal static price and is consistent with the No Learning Theorem for the pricing game.

Notably, the reason Table 2 represents the pricing game example we introduced before is that it is without loss of generality to consider the seller has three available prices $\{i_0 = 8, i_1 = 35, i_2 = 96\}$ when $v \sim \{8, 35, 96\}$. However, we can consider this pricing game example (i.e., the seller sells a single item to a buyer, whose value v over the item is drawn uniformly over $v \sim V = \{8, 35, 96\}$) in a more traditional principal-agent problem perspective. In this case, the seller may have a finite action set $I = \{i_0, i_1, i_2\}$ which may not necessarily be the same as the buyer's value set V. For example, let us consider $\{i_0 = 22, i_1 = 40, i_2 = 61\}$ (again, some arbitrarily generated numbers). Similarly, we can write out the seller/principal's and the buyer/agent's utility matrices as in Table 3.

U	j_0	j_1
i_0	0	22
i_1	0	40
i_2	0	61

V^0	j_0	j_1
i_0	0	-14
i_1	0	-32
i_2	0	-53

V^1	j_0	j_1
i_0	0	13
i_1	0	-5
i_2	0	-26

V^2	j_0	j_1
i_0	0	74
i_1	0	56
i_2	0	35

Table 3: A pricing game example where U represents the principal's utility matrix, and V^0, V^1, V^2 represents the agent's utility matrices when his type is $v = 8$, $v = 35$, and $v = 96$ correspondingly. In addition, i_0, i_1, i_2 represents setting a price of *22*, *40*, *61* (note the difference from Table 2), while j_0/j_1 represents the agent rejects/accepts the price.

Interestingly, with this small modification to the pricing game, represented by Table 3, a dynamic policy indeed outperforms the optimal static policy. The optimal static principal policy can be computed as $(\frac{5}{18}, \frac{13}{18}, 0)$, which gives the principal an expected average utility of $\mathbf{23\frac{1}{3}}$. On the other hand, the optimal dynamic policy is to play a mixed strategy $(0, 0, 1)$ (i.e., set a price of 61) in the first round. If the price is rejected, the principal switches to the mixed strategy of $(\frac{2}{3}, 0, \frac{1}{3})$; the principal keeps the same price if the price is accepted in the first round. As a result, Agent V^0 with private value 8 rejects for two rounds; Agent V^2 with private value 96 accepts the price for two rounds; Agent V^1 rejects the price in the first round and then accepts in the second round. This dynamic policy gives the principal an expected average utility of $\mathbf{26\frac{1}{6}}$, showing that the optimal dynamic policy outperforms the optimal static policy.

This example also provides some intuition about one question some attentive readers might ask, "Given that the allocation rule gives the seller more power than the principal in a general principal-agent problem, why is the seller less able to learn than in the general principal-agent problem?" Notably, in the above example, the optimal dynamic utility is still worse than Myerson's revenue of $\mathbf{32}$ when the seller has an available price of 96. Therefore, we remark that even though the no learning theorem breaks down in the principal-agent problem (Table 3) in the sense that the optimal dynamic policy outperforms the optimal static policy, it actually does not outperform the Myerson revenue. In general principal-agent problems, there actually does not exist an exact analogy of the Myerson optimal price or the single-round mechanism with something as simple as price plus allocation from pricing games.

5 Efficacy of the `DSE`

In this section, we analyze the utility the `DSE` can secure for the principal — in particular, is it truly better than the optimal principal utility for all possible static policies? We provide an affirmative answer to this question, up to an $O(1/\sqrt{T})$ discrepancy. Before stating our main result of this section, we introduce the *optimal* static principal utility in a Bayesian principal-agent problem, played against an unknown agent drawn from a prior distribution.

Benchmark: the Theoretically Optimal Static Strategy. By invoking the *revelation principle* of Myerson [1982] for general principal-agent problems, the principal's optimal static strategy is to offer an *incentive compatible* (IC) *menu* of randomized strategies. Such a menu can be described by a pair $(\boldsymbol{p}, \boldsymbol{x}) = \{p_{\theta,j}, \boldsymbol{x}_{\theta,j}\}_{j\in[n],\theta\in\Theta}$. After committing to this menu, the principal asks the agent to report his type θ. She then draws $\boldsymbol{x}_{\theta,j}$ with probability $p_{\theta,j}$ and plays $\boldsymbol{x}_{\theta,j}$, which will incentivize agent type θ to best respond with action $j \in [n]$. The computation of the optimal menu, which we coin the Randomized Menu Equilibrium (RME), has been studied recently in general principal-agent problems Gan et al. [2022] and contract design Castiglioni et al. [2022]. Specifically, Gan et al. [2022] shows that the RME can be computed in polynomial time.

Note that this is no longer, strictly, a Bayesian Stackelberg equilibrium since it requires both communication between the principal and the agent and that the principal must randomize over a menu of (already randomized) mixed strategies. This implies the RME may not be applicable in many settings since: (1) principal-agent communication is not possible in many domains, such as security games [Sinha et al., 2018]; (2) the randomized menu over mixed strategies may not be plausible as a commitment due to the infeasibility of verifying the two layers of randomness. Therefore, we treat it mainly as a strong theoretic "optimality benchmark" for us to compare against. The main result of this section shows that the DSE is guaranteed to achieve nearly identical utility, up to a small $O(\frac{1}{\sqrt{T}})$ discrepancy. This illustrates the power of using dynamic policies. Note that we could, alternatively, consider the *dynamic* RME, i.e. the menu of dynamic policies, which would, by the revelation principle, be the strongest possible dynamic policy. However, we are concerned with identifying and comparing to the strongest *static* policy, so we restrict our attention to the static RME in this work.

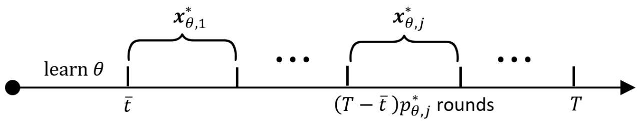
Let $U^{\texttt{DSE}}$ and $U^{\texttt{RME}}$ denote the principal's utility at DSE and RME. Our result on the comparison between $U^{\texttt{DSE}}$ and $U^{\texttt{RME}}$ hinges on a non-degeneracy assumption pertaining to a notion coined the *inducibility gap*, denoted by δ, a concept adopted in previous work [Gan et al., 2023, Wu et al., 2022].

Definition 1 (Inducibility Gap)**.** *The inducibility gap of a Bayesian principal-agent problem with agent types Θ is the largest δ such that there exists an IC randomized menu $\{\boldsymbol{x}_{\theta,j}, p_{\theta,j}\}_{\theta\in\Theta,j\in[n]}$ where for any agent type θ: $\sum_j p_{\theta,j} V^\theta(\boldsymbol{x}_{\theta,j}, j) \geq \sum_j p_{\theta',j} \max_{j'} V^\theta(\boldsymbol{x}_{\theta',j}, j') + \delta, \forall\, \theta' \neq \theta$.*

A positive inducibility gap simply means that every agent type $\theta \in \Theta$ can be *strictly* incentivized to report truthfully by some randomized menu. This is a non-degeneracy assumption since any Bayesian principal-agent problem trivially has $\delta \geq 0$ because playing any fixed mixed strategy $\boldsymbol{x}$, irrespective of agent types, already (weakly) incentivizes truthful reports from every agent type. A randomized menu with δ inducibility gap is called δ-strictly IC.

Theorem 1. *For any principal-agent problem with inducibility gap δ and $T \geq \Omega(\log_n^2 |\Theta|)$, we have $\frac{U^{\texttt{DSE}}}{T} \geq U^{\texttt{RME}} - O(\sqrt{\frac{\log|\Theta|}{T\delta^2}})$. Moreover, there exist instances where $\frac{U^{\texttt{DSE}}}{T} \leq U^{\texttt{RME}} - \Omega(\frac{1}{T})$.*

Proof Sketch. The high-level proof idea is as follows. Given any RME $\{p^*_{\theta,j}, \boldsymbol{x}^*_{\theta,j}\}_{j\in[n],\theta\in\Theta}$, we want to construct a dynamic policy (not necessarily the optimal DSE) that can "simulate" the given RME. Unfortunately, this construction has two main challenges that we have to overcome. First, the RME offers a menu of randomized strategies $\{p^*_{\theta,j}, \boldsymbol{x}^*_{\theta,j}\}_{j\in[n]}$ for each type θ. Because of incentive compatibility, the follower reports type θ truthfully and the leader plays a randomly selected strategy. However, the dynamic policy plays a single strategy at each round. As a result, the dynamic policy must use the initial rounds to learn the follower's type θ and then simulate the corresponding randomized strategy on the RME. Another challenge comes from the simulation of randomization over strategies with deterministically chosen strategies. An intuitive idea is to play $\boldsymbol{x}^*_{\theta,j}$ for the number of rounds that is proportional to $p^*_{\theta,j}$. Unfortunately, this requirement cannot be exactly fulfilled unless the number of rounds can be divided *evenly* for every j according to $p^*_{\theta,j}$; Otherwise,


$\boldsymbol{x}^*_{\theta,1}$
$\boldsymbol{x}^*_{\theta,j}$
learn θ
$\bar{t}$
$(T-\bar{t})p^*_{\theta,j}$ rounds
T

the follower's incentives may be distorted. A timeline of this high-level idea is sketched in the following figure. We show how to address these challenges in three steps.

Step 1: Eliciting follower type θ. This phase takes $\bar{t} = \lceil \log_n(|\Theta|) \rceil$ rounds. Construct function $H : \Theta \to [n]^{\bar{t}}$ such that $H(\theta)$ equals precisely the length-$\bar{t}$ *bit sequence* of the n-nary representation of θ (interpreting θ as an integer equaling at most $|\Theta|$). By definition, each θ has a unique $H(\theta)$ value. The leader strategies in these first $\bar{t}$ rounds can be arbitrary. The main challenge is deferred to the second phase below, during which we will carefully design the strategies to incentivize each follower type θ to respond exactly with action sequence $H(\theta)$ in this first phase.

Step 2: Constructing an approximately-optimal strictly-IC randomized menu. There are two reasons why we need strict incentive compatibility when simulating the randomized menu. The first is to incentivize every follower θ to respond with $H(\theta)$ in the above Step 1. To achieve this, we need to guarantee that, from θ's point of view, the strategy sequence in the remaining $T - \bar{t}$ rounds for him is strictly better — in fact at least additively $\bar{t}$ better — than the strategy sequence for any other θ'. The second reason is more intrinsic: we have to round all the probabilities in the randomized menu being played after $\bar{t}$ into multipliers of $1/(T - \bar{t})$, such that they can be *precisely* realized by deterministic strategies. Because of this, we also need a strictly IC menu to make up for the incentive distortion during the rounding of probabilities. In this step, we show that there always exists such a randomized menu that is $O(\sqrt{\frac{\log|\Theta|}{T\delta^2}})$ optimal and $O(\sqrt{\frac{\log|\Theta|}{T}})$-strictly IC (this is where the non-degenerate inducibility gap is needed). A high-level idea is to construct a new randomized menu that is both near-optimal and strictly IC by mixing the $\texttt{RME}$ with the δ-strictly IC menu.

Lemma 1. *For Stackelberg games with inducibility gap δ and $T \geq \Omega(\bar{t}^2)$, there always exists a randomized menu that is $O(\sqrt{\frac{\log|\Theta|}{T\delta^2}})$ optimal and $O(\sqrt{\frac{\log|\Theta|}{T}})$-strictly IC.*

Step 3: Existence of $(T - \bar{t})$-uniform, $O(\sqrt{\frac{\log|\Theta|}{T\delta^2}})$ optimal, and IC menu. From the previous step, we can construct a $O(\sqrt{\frac{\log|\Theta|}{T\delta^2}})$ optimal $O(\sqrt{\frac{\log|\Theta|}{T}})$-strictly IC randomized menu $\langle \boldsymbol{p}, \boldsymbol{x} \rangle$. But the $\boldsymbol{p}_\theta$ cannot be exactly fulfilled by deterministic strategies unless the number of rounds can be divided evenly for each action according to $\boldsymbol{p}_\theta$. Next, we introduce a useful lemma for rounding the probabilistic distribution $\boldsymbol{p}_\theta$ to be a k-uniform distribution $\overline{\boldsymbol{p}}_\theta$[6] with a bounded loss on the utility of all players.

Lemma 2 ([Althöfer, 1994])**.** *For any $\epsilon > 0$ and any $\{p_{\theta,j}, \boldsymbol{x}_{\theta,j}\}_{j \in [n]}$, there exists a k-uniform $\overline{\boldsymbol{p}}_\theta$ with $k = \lceil \frac{\log 2(|\Theta|+1)}{2\epsilon^2} \rceil$ such that*

$$|\sum_j p_{\theta,j} u_l(\boldsymbol{x}_{\theta,j}, j) - \sum_j \overline{p}_{\theta,j} u_l(\boldsymbol{x}_{\theta,j}, j)| \leq \epsilon, \textit{ and for all } \theta'$$
$$|\sum_j p_{\theta,j} \max_{j'} u_f^{\theta'}(\boldsymbol{x}_{\theta,j}, j') - \sum_j \overline{p}_{\theta,j} \max_{j'} u_f^{\theta'}(\boldsymbol{x}_{\theta,j}, j')| \leq \epsilon.$$

A complete proof of the last two steps can be found in appendix A.

Lastly, an instance with $\Omega(1/T)$ lower bound. We remark that the upper and lower bound of the utility comparison above is off by a factor of $O(1/\sqrt{T})$. Closing this gap is an interesting open question. To prove the $\Omega(1/T)$ bound, we consider the following bimatrix game instance,

[6] $\overline{\boldsymbol{p}}_\theta$ vector can be chosen such that $\overline{p}_{\theta,j} = k_j/k$ with natural numbers k_j for all $j = 1, \cdots, n$.

U	j_0	j_1
i_0	1	0
i_1	0	1

V^0	j_0	j_1
i_0	1	0
i_1	1	0

V^1	j_0	j_1
i_0	0	1
i_1	0	1

in which the principal has a prior distribution $(0.5, 0.5)$ over the two agent types (V^0, V^1). It is straightforward that the RME is to offer $\boldsymbol{x}^{V^0} = (1, 0)$ with probability 1 for type V^0; $\boldsymbol{x}^{V^1} = (0, 1)$ with probability 1 for type V^1. As a result, $u_l^{\texttt{RME}} = 1$. If the principal cannot offer a menu of strategies but applies a dynamic policy instead, the optimal policy would be to use the first round to learn the agent's type based on his response (either j_0 or j_1) and play pure strategy i_0 or i_1 accordingly. The optimal expected utility at the first round, however, is at most 0.5. In this case, we have $U^{\texttt{DSE}} = 0.5 + (T - 1)$ while playing RME gives the principal a utility of T, if the principal and the agent interact for T rounds. Therefore, we get $\frac{u_l^{\texttt{DSE}}}{T}$ =RME$-\frac{1}{2T}$, proving the theorem. □

Remark. *Theorem 1 shows that DSE always outperform RME, up to a $O(\sqrt{\frac{1}{T}})$ gap. It turns out that one can easily find examples in which RME will be significantly worse — specifically, $\Omega(1)$ worse — than the average DSE utility, even when there is only a single agent type.*

Consider the following bimatrix game example:

L	j_0	j_1
i_0	0.5	1
i_1	0	0

F	j_0	j_1
i_0	1	0.5
i_1	0	0

The optimal static principal strategy is $(1, 0)$, i.e. playing the pure action i_0, leading to a principal utility of 0.5 per round or $0.5T$ in total. On the other hand, we propose the following dynamic policy for the principal:

$$\boldsymbol{x}^1 = (1, 0); \quad \boldsymbol{x}^t = \begin{cases} (0, 1) \text{ if } \exists\, t' < t \text{ and } j_{t'} = j_0 \\ (1, 0) \text{ otherwise} \end{cases} \quad \text{for } t > 1. \tag{9}$$

The agent's optimal responses turn out to be always responding with j_1 for the $T - 1$ interactions and responding with j_0 at the final T'th round, leading to a total principal utility of $T - 0.5$. As a result, there is a gap of $0.5(T - 1)$ between the total utility under the optimal static strategy and the dynamic policy (9).

6 Computing the DSE Both Approximately and Exactly

We remark that when $T = 1$, the DSE is the BSE, which is known to be NP-hard [Conitzer and Sandholm, 2006]. With $T > 1$, even writing down a dynamic policy takes space exponential in T as it has to specify a strategy for each possible sequence of agent actions. Thus, our computational study focuses on developing practical algorithms for computing the DSE. First, we explore the computation of the optimal *Markovian* principal policy, i.e. a policy where the principal's strategy in any round is a function of the agent's response in the previous round only. We demonstrate that the optimal Markovian principal policy can be computed through a novel Mixed Integer Linear Program (MILP) formulation using only a *polynomial* number of decision variables. Additionally, by drawing on techniques similar to those used in constructing the MILP for the Markovian principal policies, we construct a MILP for the optimal (non-Markovian) dynamic principal policy. While the second program does involve an exponential number of decision variables, this is to be expected given that the size of the dynamic policy description is $\Omega(n^T)$. However, we demonstrate, using industry-standard optimization solvers such as Cplex [2009], Gurobi Optimization, LLC [2022], that even the optimal dynamic policy can be practically computed for Bayesian principal-agent problems. In Section 8, we implement both programs and demonstrate that the optimal Markovian policy can serve as a strong approximation to the principal utility realized under the optimal policy, while significantly improving computational time.

6.1 Efficient Approximation of DSE via Markovian Policies

In this subsection, we present a heuristic approach to improve the scalability. The approach is inspired by the Markov Decision Process (MDP), where an environment interacts with an agent and the transition of the environment from one state to another depends only on the current state and the action taken by the agent. Similarly, we consider the Markovian principal policy, where principal strategy $\boldsymbol{x}^t$ only depends on the time step t and the agent response j_{t-1} (i.e. Markovian policy $\boldsymbol{x}^t_{j_{t-1}}$[7] $= \pi(j_{t-1}) : [n] \times [T] \to \Delta^m$). Specifically, we can formally define the optimal Markovian principal policy as the solution to the following program (10).

$$\begin{aligned}
\max \quad & \textstyle\sum_{t\in[T]} \sum_{\theta\in\Theta} \left[\mu(\theta)\, U(\boldsymbol{x}^t_{j^\theta_{t-1}}, j^\theta_t)\right] \\
\text{s.t.} \quad & \textstyle\sum_{t\in[T]} V^\theta(\boldsymbol{x}^t_{j^\theta_{t-1}}, j^\theta_t) \geq \sum_{t\in[T]} V^\theta(\boldsymbol{x}^t_{\widehat{j}_{t-1}}, \widehat{j}_t), \forall \theta \in \Theta, \widehat{\boldsymbol{j}}_T \in [n]^T \\
& \boldsymbol{j}^\theta_T \in [n]^T, \qquad\qquad\qquad\qquad \forall \theta \in \Theta
\end{aligned} \tag{10}$$

Note that in Program (10), the decision variables j^θ_{t-1} appear in the *indices* of other decision variables, $\boldsymbol{x}^t_{j_{t-1}}$, which cannot be solved by any known optimization software. Towards that end, the main result of this section is a Mixed Integer Linear Program (MILP) formulation for computing the optimal Markovian policy. Deriving such a MILP is nontrivial since a naive formulation of the problem is not solvable. Indeed, our development of the MILP formulation is divided into three technical steps, as shown in the following theorem.

Theorem 2. *The optimal dynamic Markovian policy can be computed by a* MILP *with a* $O(T|\Theta|mn^2)$ *number of continuous variables and* $T|\Theta|n$ *integer variables.*

Proof. At every round t, a dynamic Markovian principal policy determines the principal strategy according to the agent response at the previous round, denoted as $\boldsymbol{x}^t_{j_{t-1}}$. The principal's cumulative utility throughout the interaction can be written as $\sum_{t\in[T]} \sum_{\theta\in\Theta}[\mu(\theta)\, U(\boldsymbol{x}^t_{j^\theta_{t-1}}, j^\theta_t)]$. We represent the agent's response sequence as $\boldsymbol{y}^\theta \in \{0,1\}^{T\times n}$, where $y^\theta_{t,j} = 1$ if the agent type θ plays j at round t. Thus, the principal's expected cumulative utility can be written as

$$\textstyle\sum_\theta \mu(\theta) \sum_{t,j_{t-1},j_t} \left(\langle \boldsymbol{x}^t_{j_{t-1}}, \boldsymbol{u}_{j_t}\rangle + \alpha_{j_t}\right) y^\theta_{t-1,j_{t-1}}\, y^\theta_{t,j_t}, \tag{11}$$

where $y^\theta_{t-1,j_{t-1}}\, y^\theta_{t,j_t}$ is non-zero when both of them are 1. We can characterize the agent's optimal response history with $\boldsymbol{y}$ using the following constraint:

$$0 \leq a_\theta - \left(\textstyle\sum_t \langle \boldsymbol{x}^t_{j_{t-1}}, \boldsymbol{v}^\theta_{j_t}\rangle + \beta_{j_t}\right) \leq M\left(T - \sum_{t=1}^T y^\theta_{t,j_t}\right), \; \forall \boldsymbol{j}_T, \theta\ ^{8} \tag{12}$$

where M is a very large constant and $\boldsymbol{a} \in \mathbb{R}^{|\Theta|}$ is a set of newly introduced *decision variables*. Formally, Program (10) can be transformed into a Mixed Integer Program (MIP) as shown by the following lemma.

Lemma 3. *Program (10) can be written as a* MIP.

[7] At the starting round $t = 1$, there does not exist a historical agent response, the principal plays initialization strategy. We denote it $\boldsymbol{x}^1_{j_0}$ throughout the paper for the notation consistency, where j_0 can be thought as NULL.

[8] We omit the feasible regions of variables (e.g. $t \in [T], i \in [m], \boldsymbol{j}_T \in [n]^T, \theta \in \Theta$) in some programs for the sake of space when they are clear from the context.

Proof. Program (10) is equivalent to the following `MIP` (13).

$$\begin{aligned}
\max \quad & \textstyle\sum_\theta \mu(\theta) \sum_{t,j_{t-1},j_t} \left(\langle \boldsymbol{x}^t_{j_{t-1}}, \boldsymbol{u}_{j_t}\rangle + \alpha_{j_t}\right) y^\theta_{t-1,j_{t-1}}\, y^\theta_{t,j_t} \\
\text{s.t.} \quad & \textstyle\sum_i x^t_{j_{t-1},i} = 1; \quad \sum_j y^\theta_{t,j} = 1, \qquad\qquad\qquad\qquad \forall j_{t-1}, t, \theta \\
& 0 \le a_\theta - (\textstyle\sum_t \langle \boldsymbol{x}^t_{j_{t-1}}, \boldsymbol{v}^\theta_{j_t}\rangle + \beta_{j_t}) \ \le M(T - \sum_{t\in[T]} y^\theta_{t,j_t}), \forall \boldsymbol{j}_T, \theta \\
& 0 \le \boldsymbol{x} \le 1; \quad \boldsymbol{y} \in \{0,1\}.
\end{aligned} \tag{13}$$

Reformulating agent's incentive constraints in Program (10). We argue that under our new variable $\boldsymbol{y}^\theta \in \{0,1\}^{T\times n}$, the following constraint is equivalent to the first set of constraints in Program (10) and thus correctly captures the agent's dynamic optimal responses:

$$0 \le a_\theta - (\sum_t \langle \boldsymbol{x}^t_{j_{t-1}}, \boldsymbol{v}^\theta_{j_t}\rangle + \beta_{j_t}) \le M(T - \sum_{t=1}^{T} y^\theta_{t,j_t}), \; \forall \boldsymbol{j}_T, \theta, \tag{14}$$

where M is a very large constant, $\boldsymbol{a} \in \mathbb{R}^k$ is a set of newly introduced *decision variables*.

To see how Inequality (14) characterizes the optimality condition of the agent's best responses, consider any variables $\boldsymbol{x}, \boldsymbol{y}$ and $\boldsymbol{a}$ feasible to Inequality (14). Recall that $\boldsymbol{y}^\theta_t \in \{0,1\}^n$ is a one-hot vector of length n for any $\theta \in \Theta$ and $t \in [T]$. We denote $j^{*\theta}_t \in [n]$ as the index of the unique 1-valued entry in $\boldsymbol{y}^\theta_t$, and vector $\boldsymbol{j}^{*\theta}_T = (j^{*\theta}_1, \cdots, j^{*\theta}_T)$. We argue that if $\boldsymbol{x}, \boldsymbol{y}$ and $\boldsymbol{a}$ satisfy Inequality (14), then $\boldsymbol{j}^{*\theta}_T$ must be an optimal response sequence for type θ.

By construction, we have $\sum_t y^\theta_{t,j^{*\theta}_t} = T$, for all θ and $\boldsymbol{j}^{*\theta}_T$ by definition. Plugging this into Inequality (14), we obtain $M(T - \sum_t y^\theta_{t,j^{*\theta}_t}) = 0$. Thus Inequality (14) becomes $0 \le a_\theta - (\sum_{t\in[T]} \langle \boldsymbol{x}^t_{j^{*\theta}_{t-1}}, \boldsymbol{v}^\theta_{j^{*\theta}_t}\rangle + \beta_{j^{*\theta}_t}) \le 0$, which implies

$$a_\theta = \sum_{t\in[T]} \langle \boldsymbol{x}^t_{j^{*\theta}_{t-1}}, \boldsymbol{v}^\theta_{j^{*\theta}_t}\rangle + \beta_{j^{*\theta}_t}, \quad \forall \theta. \tag{15}$$

That is, a_θ is precisely agent type θ's total utility from response $\boldsymbol{y}^\theta$.

Now let us consider any other possible response sequence $\boldsymbol{j}_T \in [n]^T \neq \boldsymbol{j}^{*\theta}_T$. Since $\boldsymbol{y}^\theta_t$ is a one-hot vector and $y^\theta_{t,j_t} = 1$ if and only if $j_t = j^{*\theta}_t$. Therefore, if $\boldsymbol{j}_T \neq \boldsymbol{j}^{*\theta}_T$, there must exists some t such that $y^\theta_{t,j_t} = 0$, hence,

$$\sum_t y^\theta_{t,j_t} < T, \quad \forall \theta, \boldsymbol{j}_T \neq \boldsymbol{j}^{*\theta}_T.$$

Plugging this $\boldsymbol{j}_T$ into (14), we must have $M(T - \sum_t y^\theta_{t,j_t}) \ge M$ as a very large constant, which makes the right-hand-side inequality void for any $\boldsymbol{j}_T \in [n]^T \neq \boldsymbol{j}^{*\theta}_T$. Thus, the only useful constraint is $0 \le a_\theta - (\sum_{t\in[T]} \langle \boldsymbol{x}^t_{j_{t-1}}, \boldsymbol{v}^\theta_{j_t}\rangle + \beta_{j_t})$ for any $\boldsymbol{j}_T \neq \boldsymbol{j}^{*\theta}_T$, which implies

$$\sum_{t\in[T]} \langle \boldsymbol{x}^t_{j_{t-1}} \boldsymbol{v}^\theta_{j_t}\rangle + \beta_{j_t} \le a_\theta, \quad \forall \theta, \boldsymbol{j}_T \neq \boldsymbol{j}^{*\theta}_T. \tag{16}$$

It is now clear to see that $\boldsymbol{j}^{*\theta}_T$ is the optimal response for type θ by combining (16) and (15).

Reformulating the objective function in Program (10). We claim that for any feasible variable $\boldsymbol{x}, \boldsymbol{y}$ and $\boldsymbol{a}$, the objective of Program (10) equals to the following expression:

$$\sum_\theta \mu(\theta) \sum_{t,j_{t-1},j_t} \left(\langle \boldsymbol{x}^t_{j_{t-1}}, \boldsymbol{u}_{j_t}\rangle + \alpha_{j_t}\right) y^\theta_{t-1,j_{t-1}}\, y^\theta_{t,j_t} \tag{17}$$

The objective above enumerates all possible path histories $j_{t-1}, j_t \in [n]$ for any $t \in [T]$. However, the product $y^\theta_{t-1,j_{t-1}}\, y^\theta_{t,j_t}$ is only non-zero when both $y^\theta_{t-1,j_{t-1}}$ and y^θ_{t,j_t} are non-zero, i.e. when $y^\theta_{t-1,j_{t-1}}$ and y^θ_{t,j_t} is on the optimal agent response path for agent type θ. This guarantees that Objective (17) only counts the j_{t-1} and j_t on the optimal path $\boldsymbol{j}^{*\theta}_T$. Therefore, it correctly calculates the expected principal utility when each agent type θ follows their optimal response paths.

Showing the equivalence. Consider $\boldsymbol{x}$ and $\boldsymbol{j}$ as a feasible solution of (10). We will show that $\boldsymbol{x}$, $y^{\theta}_{t,j} = \begin{cases} 1 \text{ if } j = j^{\theta}_t \\ 0 \text{ otherwise} \end{cases}$, and $a_\theta = \sum_{t\in[T]} V^{\theta}(\boldsymbol{x}^t_{j^{\theta}_{t-1}}, j^{\theta}_t)$ is a feasible solution of (13) of the same objective function value. The last four constraints of (13) are satisfied by construction. To see how the first constraint is satisfied, note for any $\boldsymbol{j}_T \neq \boldsymbol{j}^{\theta}_T$ returned by (10) of type θ, we have $\sum_{t\in[T]} y^{\theta}_{t,j_t} < T$ by construction, and its corresponding agent utility $\sum_t \langle \boldsymbol{x}^t_{j_{t-1}}, \boldsymbol{v}^{\theta}_{j_t}\rangle + \beta_{j_t} \le a_\theta$ by the definition that $\boldsymbol{j}_T$ is not the best response. When $\boldsymbol{j}_T = \boldsymbol{j}^{\theta}_T$, we have $M(T - \sum_{t\in[T]} y^{\theta}_{t,j_t}) = 0$ and $a_\theta = \sum_t \langle \boldsymbol{x}^t_{j_{t-1}}, \boldsymbol{v}^{\theta}_{j_t}\rangle + \beta_{j_t}$ by construction. Hence, the first constraint of (13) is also satisfied. The fact that $y^{\theta}_{t,j} = 0$ guarantees that $y^{\theta}_{t-1,j_{t-1}}\, y^{\theta}_{t,j_t} = 0$ if j_{t-1} or $j_t \notin \boldsymbol{j}^{\theta}_T$. Then we have $\sum_{t,j_{t-1},j_t} (\langle \boldsymbol{x}^t_{j_{t-1}}, \boldsymbol{u}_{j_t}\rangle + \alpha_{j_t})\, y^{\theta}_{t-1,j_{t-1}}\, y^{\theta}_{t,j_t} = \sum_t \langle \boldsymbol{x}^t_{j^{\theta}_{t-1}}, \boldsymbol{u}_{j^{\theta}_t}\rangle + \alpha_{j^{\theta}_t} = \sum_t U(\boldsymbol{x}^t_{j^{\theta}_{t-1}}, j^{\theta}_t)$, which shows the constructed feasible solution of (13) has the same objective value as (10).

Let us now consider $\boldsymbol{x}$, $\boldsymbol{y}$, and $\boldsymbol{a}$ feasible for (13). We construct $\boldsymbol{j}^{\theta}_T$ such that $j^{\theta}_t \in [n]$ and $y^{\theta}_{t,j^{\theta}_t} = 1$. We will show that $\boldsymbol{x}$ and $\boldsymbol{y}$ are feasible for (10) with the same objective value. Recall the discussion from equations (15)– (16), we have that $\boldsymbol{j}^{\theta}_T$ captures the agent's optimal behavior and satisfies $\sum_{t\in[T]} V^{\theta}(\boldsymbol{x}^t_{j^{\theta}_{t-1}}, j^{\theta}_t) \ge \sum_{t\in[T]} V^{\theta}(\boldsymbol{x}^t_{\widehat{j}_{t-1}}, \widehat{j}_t)$. What is more, by the same argument as in the previous direction, we have $\sum_\theta \mu(\theta) \sum_{t,j_{t-1},j_t} (\langle \boldsymbol{x}^t_{j_{t-1}}, \boldsymbol{u}_{j_t}\rangle + \alpha_{j_t})\, y^{\theta}_{t-1,j_{t-1}}\, y^{\theta}_{t,j_t} = \sum_\theta \mu(\theta) \sum_t U(\boldsymbol{x}^t_{j^{\theta}_{t-1}}, j^{\theta}_t)$, which shows the equivalence between the objective values of these two programs. □

Finally, we can linearize the production of decision variables $x^t_{j_{t-1},i}\, y^{\theta}_{t-1,j_{t-1}}\, y^{\theta}_{t,j_t}$ in the `MIP` of the previous Lemma.

$$z^{t,\theta}_{j_{t-1},j_t} = y^{\theta}_{t-1,j_{t-1}}\, y^{\theta}_{t,j_t} \text{ and } w^{t,\theta}_{j_{t-1},j_t,i} = x^t_{j_{t-1},i}\, y^{\theta}_{t-1,j_{t-1}}\, y^{\theta}_{t,j_t}. \tag{18}$$

The key challenge for deriving our re-formulation is to set up the right set of constraints for these new variables so that they will exactly enforce the feasibility of the original variables.

$$\begin{aligned}
\max \quad & \textstyle\sum_\theta \mu(\theta) \sum_{t,j_{t-1},j_t} \langle \boldsymbol{w}^{t,\theta}_{j_{t-1},j_t}, \boldsymbol{u}_{j_t}\rangle + \alpha_{j_t} z^{t,\theta}_{j_{t-1},j_t} && && \text{(19a)}\\
\text{s.t.} \quad & z^{t,\theta}_{j_{t-1},jt} \le y^{\theta}_{t,j_t};\ z^{t,\theta}_{j_{t-1},jt} \le y^{\theta}_{t-1,j_{t-1}};\ z^{t,\theta}_{j_{t-1},jt} \ge y^{\theta}_{t,j_t} + y^{\theta}_{t-1,j_{t-1}} - 1; && \forall j_{t-1}, j_t, t, \theta && \text{(19b)}\\
& w^{t,\theta}_{j_{t-1},j_t,i} \le x^t_{j_{t-1},i};\ w^{t,\theta}_{j_{t-1},j_t,i} \le z^{t,\theta}_{j_{t-1},j_t}; && \forall j_{t-1}, j_t, t, \theta, i && \text{(19c)}\\
& w^{t,\theta}_{j_{t-1},j_t,i} \ge x^t_{j_{t-1},i} - M(1 - z^{t,\theta}_{j_{t-1},j_t}); && \forall j_{t-1}, j_t, t, \theta, i && \text{(19d)}\\
& \textstyle\sum_i x^t_{j_{t-1},i} = 1; \sum_j y^{\theta}_{t,j} = 1; && \forall t, j_{t-1}, \theta && \text{(19e)}\\
& 0 \le a_\theta - (\textstyle\sum_t \langle \boldsymbol{x}^t_{j_{t-1}}, \boldsymbol{v}^{\theta}_{j_t}\rangle + \beta_{j_t}) \le M(T - \sum_t y^{\theta}_{t,j_t}); && \forall \boldsymbol{j}_T, \theta && \text{(19f)}\\
& 0 \le \boldsymbol{z} \le 1; \quad 0 \le \boldsymbol{w} \le 1; \quad 0 \le \boldsymbol{x} \le 1; \quad \boldsymbol{y} \in \{0,1\}. && && \text{(19g)}
\end{aligned}$$

Lemma 4. *Program* (10) *can be written as the* `MILP` (19).

The proof of Lemma 4 has two steps. We first show that any optimal solution for `MIP` (13) must correspond to a feasible solution of `MILP` (19) with the same objective value, and then show its reverse direction.

Step 1: $OPT(19) \ge OPT(13)$**.** Consider any optimal solution $\boldsymbol{x}$, $\boldsymbol{y}$, and $\boldsymbol{a}$ for MIP (13). We argue that $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{a}$, together with constructed $\boldsymbol{z}, \boldsymbol{w}$ via Equations (18), forms a feasible solution of (19). The equivalence of the objective function is immediately satisfied by construction. It is also obvious that $0 \le \boldsymbol{z} \le 1$ and $0 \le \boldsymbol{w} \le 1$ by construction. Next, we show that constraints (19b) – (19d) are satisfied. Recall that each entry of $\boldsymbol{y}$ is binary. For any θ, t and $(j_1, \cdots, j_t)$, we consider the following two possible cases:

- If either $y^{\theta}_{t-1,j_{t-1}} = 0$ or $y^{\theta}_{t,j_t} = 0$, then we have $z^{t,\theta}_{j_{t-1},j_t} = 0$ and $w^{t,\theta}_{j_{t-1},j_t,i} = 0$ by construction, As a result, it is obvious that the first two constraints of (19b) and (19c) are satisfied. To see why the last constraint of (19b) holds, note that as long as either $y^{\theta}_{t-1,j_{t-1}} = 0$ or $y^{\theta}_{t,j_t} = 0$, then we have $y^{\theta}_{t,j_t} + y^{\theta}_{t-1,j_{t-1}} - 1 \leq 0 = z^{t,\theta}_{j_{t-1},j_t}$. Finally, when $z^{t,\theta}_{j_{t-1},j_t} = 0$, we have the right-hand side of (19d) be a large negative number, so (19d) is satisfied as well.

- If $y^{\theta}_{i,j_i} = 1$ for all $i = 1, \cdots, t$, then we have $z^{t,\theta}_{j_{t-1},j_t} = 1$ and $w^{t,\theta}_{j_{t-1},j_t,i} = x^{t}_{j_{t-1},i}$ by construction. It is also straightforward to see that constraint the first two constraints of (19b), (19c), and (19d) are satisfied. Moreover, we have $\sum_{i\in[t]} y^{\theta}_{i,j_i} = t$ in this case, and thus $\sum_{i\in[t]} y^{\theta}_{i,j_i} - (t-1) = 1 = z^{t,\theta}_{j_{t-1},j_t}$ so the last constraint of (19b) is satisfied as well.

Therefore, we have shown the constructed $\boldsymbol{z}, \boldsymbol{w}$ together with $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{a}$ form a feasible solution to MILP (19) with the same optimal objective value as (13). This implies $OPT(19) \geq OPT(13)$, where OPT denotes the optimal solution value of the program.

Step 2: $OPT(19) \leq OPT(13)$. We now prove the reverse direction, which is the more interesting and non-trivial step. Specifically, suppose that we are given an optimal solution $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{a}, \boldsymbol{z}, \boldsymbol{w}$ of MILP (19). We show that these $\boldsymbol{x}, \boldsymbol{y}$ and $\boldsymbol{a}$ form a feasible solution of (13) with the same objective function. It is obvious to see that $\boldsymbol{x}, \boldsymbol{y}$ and $\boldsymbol{a}$ are still feasible to (13) since the constraints of (13) is a subset of constraints of (19). The key step is to show these $\boldsymbol{x}, \boldsymbol{y}$ and $\boldsymbol{a}$ achieve the same objective value in (13). To show this, we prove that for any optimal solution $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{a}, \boldsymbol{z}, \boldsymbol{w}$ of (19), we must have $w^{t,\theta}_{j_{t-1},j_t,i} = x^{t}_{j_{t-1},i}\, y^{\theta}_{t-1,j_{t-1}}\, y^{\theta}_{t,j_t}$ and $z^{t,\theta}_{j_{t-1},j_t} = y^{\theta}_{t-1,j_{t-1}}\, y^{\theta}_{t,j_t}$ for any $\theta, t,$ and $\boldsymbol{j}_t$. We prove this is correct through a careful case analysis:

(1) If either $y^{\theta}_{t-1,j_{t-1}} = 0$ or $y^{\theta}_{t,j_t} = 0$, then we have $z^{t,\theta}_{j_{t-1},j_t} = 0$ from (19b) and $w^{t,\theta}_{j_{t-1},j_t,i} = 0$ from (19c), proving the equivalence.

(2) If $y^{\theta}_{t-1,j_{t-1}} = y^{\theta}_{t,j_t} = 1$, we have $z^{t,\theta}_{j_{t-1},j_t} = 1 = y^{\theta}_{t-1,j_{t-1}}\, y^{\theta}_{t,j_t}$ from (19b). Next, we claim $w^{t,\theta}_{j_{t-1},j_t,i} = x^{t}_{j_{t-1},i}\, y^{\theta}_{t-1,j_{t-1}}\, y^{\theta}_{t,j_t}$. Note that we must have $w^{t,\theta}_{j_{t-1},j_t,i} \leq x^{t}_{j_{t-1},i}$ and $w^{t,\theta}_{j_{t-1},j_t,i} \leq z^{t,\theta}_{j_{t-1},j_t}$ by constraint (19c). Note that when $y^{\theta}_{t-1,j_{t-1}} = y^{\theta}_{t,j_t} = 1$, we have $z^{t,\theta}_{j_{t-1},j_t} \leq 1$ by the first two constraints of (19b), as well as $z^{t,\theta}_{j_{t-1},j_t} \geq 1$ by the last constraint of (19b). Hence, $z^{t,\theta}_{j_{t-1},j_t} = 1$, and the second constraint of (19c) will not add additional constraint on $w^{t,\theta}_{j_{t-1},j_t,i}$. In addition, we also have $w^{t,\theta}_{j_{t-1},j_t,i} \geq x^{t}_{j_{t-1},i}$ from constraint (19d) when $z^{t,\theta}_{j_{t-1},j_t} = 1$, leading to $w^{t,\theta}_{j_{t-1},j_t,i} = x^{t}_{j_{t-1},i}$. As a result, we have shown that for any optimal solution $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{a}, \boldsymbol{z}, \boldsymbol{w}$ of MILP (19), we must have $w^{t,\theta}_{j_{t-1},j_t,i} = x^{t}_{j_{t-1},i}\, y^{\theta}_{t-1,j_{t-1}}\, y^{\theta}_{t,j_t}$. Thus, $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{a}$ form a feasible solution to MIP (13) with the same objective value, implying $OPT(19) \leq OPT(13)$.

We remark that the MILP (19) has $nT|\Theta|$ integer variables, and the sizes of continuous variables are mnT for $\boldsymbol{x}$, the size $|\Theta|Tn^2$ for $\boldsymbol{z}$, and the size $|\Theta|Tmn^2$ for $\boldsymbol{w}$. Therefore, the total number of decision variables of (19) is $O(|\Theta|Tmn^2)$, proving the theorem. □

6.2 Computing the Exact DSE

In this subsection, we show that we can also develop a corresponding MILP for computing the exact optimal dynamic principal policy. Even though this MILP has an exponential number of decision variables, it provides a practical solution for the previously unexplored NP-Hard problem.

Proposition 2. *The exact DSE can be computed by a MILP with a $O(T|\Theta|mn^T)$ number of continuous variables and $T|\Theta|n$ integer variables.*

Proof. Given a sequence of interaction history $\{(\boldsymbol{x}^t, j_t)\}_{t\in[T]}$, the principal's total utility is computed as $\sum_{t=1}^{T} U(\boldsymbol{x}^t, j_t)$ whereas the total utility of any agent type θ is $\sum_{t=1}^{T} V^{\theta}(\boldsymbol{x}^t, j_t)$. This leads to an initial optimization formulation to compute the DSE with integer variables $\boldsymbol{j}^{\theta}_{T} \in [n]^T$ denoting the response sequence of agent type θ, and continuous variables $\boldsymbol{x}^{t}_{\boldsymbol{j}_{t-1}} = \pi(\boldsymbol{j}_{t-1}) \in \Delta^m$ denoting the principal's mixed strategy at round t given any agent response sequence $\boldsymbol{j}_{t-1} \in [n]^{t-1}$. The program maximizes the principal's total utility subject to the constraint that $\boldsymbol{j}_T^\theta$ is the optimal response sequence for agent type θ:

$$\begin{aligned}
\max \quad & \sum_{t\in[T]} \sum_{\theta\in\Theta} \left[\mu(\theta)\, U(\boldsymbol{x}^t_{\boldsymbol{j}^\theta_{t-1}}, j^\theta_t)\right] \\
\text{s.t.} \quad & \sum_{t\in[T]} V^\theta(\boldsymbol{x}^t_{\boldsymbol{j}^\theta_{t-1}}, j^\theta_t) \geq \sum_{t\in[T]} V^\theta(\boldsymbol{x}^t_{\widehat{\boldsymbol{j}}_{t-1}}, \widehat{j}_t), \forall\theta\in\Theta, \widehat{\boldsymbol{j}}_T \in [n]^T \\
& \boldsymbol{j}^\theta_T \in [n]^T, \qquad\qquad\qquad\qquad \forall\theta\in\Theta
\end{aligned} \tag{20}$$

At every round t, a dynamic principal policy determines the principal strategy according to the historical agent responses, denoted as $\boldsymbol{x}^t_{\boldsymbol{j}_{t-1}}$. The principal's cumulative utility throughout the interaction can be written as $\sum_{t\in[T]}\sum_{\theta\in\Theta}[\mu(\theta)\, U(\boldsymbol{x}^t_{\boldsymbol{j}^\theta_{t-1}}, j^\theta_t)]$. Similar to our formulation in the previous subsection, we represent the agent's response sequence as $\boldsymbol{y}^\theta \in \{0,1\}^{T\times n}$, where $y^\theta_{t,j} = 1$ if the agent type θ plays j at round t. Thus, the principal's expected cumulative utility can be written as

$$\textstyle\sum_\theta \mu(\theta) \sum_{t,\boldsymbol{j}_t} (\langle \boldsymbol{x}^t_{\boldsymbol{j}_{t-1}}, \boldsymbol{u}_{j_t}\rangle + \alpha_{j_t}) \prod_{t'=1}^t y^\theta_{t',j_{t'}}, \tag{21}$$

where $\prod_{t'=1}^t y^\theta_{t',j_{t'}}$ is non-zero when all of them are 1. We can characterize the agent's optimal response history with $\boldsymbol{y}$ using a similar constraint as (12), which gives us the following MIP to solve for the optimal dynamic policy.

$$\begin{aligned}
\max \quad & \textstyle\sum_\theta \mu(\theta) \sum_{t,\boldsymbol{j}_t} (\langle \boldsymbol{x}^t_{\boldsymbol{j}_{t-1}}, \boldsymbol{u}_{j_t}\rangle + \alpha_{j_t}) \prod_{t'=1}^t y^\theta_{t',j_{t'}} \\
\text{s.t.} \quad & 0 \leq a_\theta - (\textstyle\sum_t \langle \boldsymbol{x}^t_{\boldsymbol{j}_{t-1}} \boldsymbol{v}^\theta_{j_t}\rangle + \beta_{j_t}) \ \leq M(T - \textstyle\sum_{t\in[T]} y^\theta_{t,j_t}), \forall \boldsymbol{j}_T \in [n]^T, \theta\in\Theta \\
& \textstyle\sum_i x^t_{\boldsymbol{j}_{t-1},i} = 1, && \forall \boldsymbol{j}_{t-1} \in [n]^{t-1}, t\in[T] \\
& x^t_{\boldsymbol{j}_{t-1},i} \in [0,1], && \forall i\in[m], \boldsymbol{j}_{t-1}\in[n]^{t-1}, t\in[T] \\
& \textstyle\sum_j y^\theta_{t,j} = 1, && \forall t\in[T], \theta\in\Theta \\
& y^\theta_{t,j} \in \{0,1\}, && \forall j\in[n], t\in[T], \theta\in\Theta
\end{aligned} \tag{22}$$

Finally, we can linearize the production of decision variables $x^t_{\boldsymbol{j}_{t-1},i} \prod_{t'=1}^t y^\theta_{t',j_{t'}}$ with $z^{t,\theta}_{\boldsymbol{j}_t} = \prod_{t'=1}^t y^\theta_{t',j_{t'}}$ and $w^{t,\theta}_{\boldsymbol{j}_t,i} = x^t_{\boldsymbol{j}_{t-1},i} \cdot \prod_{t'=1}^t y^\theta_{t',j_{t'}}$. As a result, we can rewrite the above program (22) as the MILP (23).

$$\begin{aligned}
\max \quad & \textstyle\sum_\theta \mu(\theta) \sum_{t,\boldsymbol{j}_t} \langle w^{t,\theta}_{\boldsymbol{j}_t}, \boldsymbol{u}_{j_t}\rangle + \alpha_{j_t} z^{t,\theta}_{\boldsymbol{j}_t} \\
\text{s.t.} \quad & z^{t,\theta}_{\boldsymbol{j}_t} \leq y^\theta_{t,j_t};\ z^{t,\theta}_{\boldsymbol{j}_t} \leq z^{t-1,\theta}_{\boldsymbol{j}_{t-1}};\ z^{t,\theta}_{\boldsymbol{j}_t} \geq y^\theta_{t,j_t} + z^{t-1,\theta}_{\boldsymbol{j}_{t-1}} - 1; && \forall \boldsymbol{j}_t\in[n]^t, t\in[T], \theta\in\Theta \\
& w^{t,\theta}_{\boldsymbol{j}_t,i} \leq x^t_{\boldsymbol{j}_{t-1},i};\ w^{t,\theta}_{\boldsymbol{j}_t,i} \leq z^{t,\theta}_{\boldsymbol{j}_t}; && \forall \boldsymbol{j}_t\in[n]^t, t\in[T], \theta\in\Theta, i\in[m] \\
& w^{t,\theta}_{\boldsymbol{j}_t,i} \geq x^t_{\boldsymbol{j}_{t-1},i} - M(1 - z^{t,\theta}_{\boldsymbol{j}_t}); && \forall \boldsymbol{j}_t\in[n]^t, t\in[T], \theta\in\Theta, i\in[m] \\
& \textstyle\sum_i x^t_{\boldsymbol{j}_{t-1},i} = 1; && \forall t\in[T], \boldsymbol{j}_{t-1}\in[n]^{t-1} \\
& \textstyle\sum_j y^\theta_{t,j} = 1; && \forall t\in[T], \theta\in\Theta \\
& 0 \leq a_\theta - (\textstyle\sum_t \langle \boldsymbol{x}^t_{\boldsymbol{j}_{t-1}} \boldsymbol{v}^\theta_{j_t}\rangle + \beta_{j_t}) \leq M(T - \textstyle\sum_t y^\theta_{t,j_t});\ \forall \boldsymbol{j}_T \in [n]^T, \theta\in\Theta \\
& 0 \leq \boldsymbol{z} \leq 1; \quad 0 \leq \boldsymbol{w} \leq 1; \quad 0 \leq \boldsymbol{x} \leq 1; \quad \boldsymbol{y} \in \{0,1\}.
\end{aligned} \tag{23}$$

□

We note that the `MILP` for Proposition 2 takes time exponential in $T, |\Theta|$ to solve. The exponential time dependence on $T, |\Theta|$ is as expected since the size of the policy description is already $\Omega(n^T)$ and the exponential-in-$|\Theta|$ time is due to the NP-hardness even when $T = 1$. We remark that our `MILP`'s running time does *not* depend on n exponentially. This is because any row of $\boldsymbol{y}^\theta$ (i.e. $\boldsymbol{y}^\theta_t$) has only a single non-zero entry, thus there are in total $O(n^{T|\Theta|})$ many feasible $\boldsymbol{y}$. Given a feasible $\boldsymbol{y}$, the program will be an LP with the size of variables bounded by a polynomial in m and n (given constants T, $|\Theta|$). This argument leads to the following corollary of Proposition 2.

Corollary 1. *The* DSE *can be computed in* $\text{poly}(m, n)$ *time with constant numbers of time steps and agent types.*

7 Application to Dynamic Contract Design

In this section, we discuss a specific principal-agent problem known as the contract design problem, in order to illustrate the implications of our previous results. What is more, by utilizing certain structural properties of the contract design game, we are able to show more intriguing outcomes regarding the characteristics of dynamic principal policies.

The agent has $[n] = \{1, \cdots, n\}$ actions to choose from. Each action $j \in [n]$ results in a probabilistic distribution $\boldsymbol{p}_j \in \Delta^m$ over $[m] = \{1, \cdots, m\}$ outcomes and has a corresponding cost $c_j \in \mathbb{R}_+$. Therefore, every agent's type θ is determined by the matrix $P \in \mathbb{R}^{n \times m}$ together with the cost vector $\boldsymbol{c} \in \mathbb{R}^n$. On the other hand, the principal has a reward $r_i \in \mathbb{R}_+$ for every outcome $i \in [m]$. The principal designs contract $\boldsymbol{x} \in \mathbb{R}_+^m$ which specifies payment x_i to the agent if outcome i is realized. As a result, the principal's utility function can be computed as follows.

$$U(\boldsymbol{x}, j) = \sum_i (r_i - x_i) p_{j,i} = R_j - \langle \boldsymbol{x}, \boldsymbol{p}_j \rangle, \tag{24}$$

where $R_j = \sum_i r_i p_{j,i}$ denotes the expected reward for action $j \in [n]$. The agent's utility function is denoted as follows.

$$V^\theta(\boldsymbol{x}, j) = \langle \boldsymbol{x}, \boldsymbol{p}_j \rangle - c_j. \tag{25}$$

As standard assumptions in the contract design model [Alon et al., 2021, Castiglioni et al., 2021, Dütting et al., 2019], we assume the first outcome has zero rewards for the principal and this outcome only occurs if and only if the agent takes the first action, which has zero cost as well. In addition, there are no actions that can be considered "dominated", meaning that for any two actions j and j', their expected rewards R_j and $R_{j'}$ are distinct and the action with a higher expected outcome ($R_j > R_{j'}$) also incurs a higher cost ($c_j > c_{j'}$). There is a unique action j^* that maximizes the welfare, i.e., $j^* = \text{argmax}_j\, R_j - c_j$. Next, we show that there exists a compact representation of the optimal dynamic policy to the leader utilizing the structural assumption on the contra design problem.

Theorem 3. *Computing the optimal dynamic contract is a problem in NP. That is, there always exists an optimal dynamic contract (as the certifier for the decision version of the problem) that has description size poly*$(m, n, T, |\Theta|)$.

Proof. For any optimal dynamic contract (i.e. the DSE, $\boldsymbol{x}^*$ and $\boldsymbol{j}^*$, which forms an optimal solution to (20)), there exists a best response path $\boldsymbol{j}_T^{*\theta}$ for every type $\theta \in \Theta$. Also, note that the description size of $\boldsymbol{x}^*$ is $O(n^T m)$ since $\boldsymbol{x}^*$ needs to specify the principal contract for any possible agent action history. Next, we propose to construct a dynamic $\widehat{\boldsymbol{x}}$ that only needs polynomial description size but achieves the same expected principal revenue as $\boldsymbol{x}^*$. Specifically, for any round $t \in [T]$ and for any possible agent action history $\boldsymbol{j}_{t-1}$, we construct $\widehat{\boldsymbol{x}}_{\boldsymbol{j}_{t-1}}^t$ as follows.

1. If there exists $\theta \in \Theta$ such that $\boldsymbol{j}_{t-1} = \boldsymbol{j}_{t-1}^{*\theta}$, then we set $\widehat{\boldsymbol{x}}_{\boldsymbol{j}_{t-1}}^t = \boldsymbol{x}_{\boldsymbol{j}_{t-1}}^{*t}$.

2. Otherwise, set $\widehat{\boldsymbol{x}}_{\boldsymbol{j}_{t-1}}^t = \boldsymbol{0}$, i.e. does not pay the agent anything.

Note that the dynamic contract $\widehat{\boldsymbol{x}}$ only specifies contract for $|\Theta|$ types of action paths (denoted as J_T), where each path has a length of T. As a result, the above dynamic contract only has a description size of $O(mT|\Theta|)$. Next, we show that $\widehat{\boldsymbol{x}}$ achieves the same expected revenue for the principal, which is equivalent to showing that every agent type θ follows the same response path under $\widehat{\boldsymbol{x}}$ and $\boldsymbol{x}^*$. Given a dynamic contract $\boldsymbol{x}$ and follower response history $\boldsymbol{j}_T$, we denote the follower's cumulative utility as

$$V^\theta(\boldsymbol{x}, \boldsymbol{j}_T) = \sum_{t\in[T]} \sum_{i\in[m]} x^t_{j_{t-1},i}\, p^\theta_{j_t,i} - c^\theta_{j_t}$$

Next, we consider the following two cases for any follower type θ.

For any path $\boldsymbol{j}_T \in J_T$ and $\boldsymbol{j}_T \neq \boldsymbol{j}_T^{*\theta}$, we have

$$V^\theta(\boldsymbol{x}^*, \boldsymbol{j}_T^{*\theta}) \geq V^\theta(\boldsymbol{x}^*, \boldsymbol{j}_T) = V^\theta(\widehat{\boldsymbol{x}}, \boldsymbol{j}_T) \tag{26}$$

by the incentive compatibility of $\boldsymbol{x}^*$ and the definition of $\widehat{\boldsymbol{x}}$.

For any path $\boldsymbol{j}_T \notin J_T$, there must exist a time step $\widehat{t} \leq T$ such that $j_t = 0$ for all $\widehat{t} \leq t \leq T$, and $\widehat{t}$ represents the time step when the leader discovers the $\boldsymbol{j}_T \notin J_T$ and does not pay the agent (i.e. $\widehat{\boldsymbol{x}}^t = \boldsymbol{0}$ for all $\widehat{t} \leq t \leq T$). First of all, we still have $V^\theta(\boldsymbol{x}^*, \boldsymbol{j}_T^{*\theta}) \geq V^\theta(\boldsymbol{x}^*, \boldsymbol{j}_T)$ by the incentive compatibility of $\boldsymbol{x}^*$. In addition, we have

$$V^\theta(\widehat{\boldsymbol{x}}, \boldsymbol{j}_T) = \sum_{t=1}^{\widehat{t}-1} \sum_{i=1}^{m} \widehat{x}^t_{j_{t-1},i}\, p^\theta_{j_t,i} - c^\theta_{j_t} = \sum_{t=1}^{\widehat{t}-1} \sum_{i=1}^{m} x^{*t}_{j_{t-1},i}\, p^\theta_{j_t,i} - c^\theta_{j_t}$$

by the definition of $\widehat{\boldsymbol{x}}$. Next, we show $V^\theta(\boldsymbol{x}^*, \boldsymbol{j}_T^{*\theta}) \geq V^\theta(\widehat{\boldsymbol{x}}, \boldsymbol{j}_T)$.

$$\begin{aligned} V^\theta(\boldsymbol{x}^*, \boldsymbol{j}_T^{*\theta}) &= \sum_{t=1}^{T} \sum_{i=1}^{m} x^{*t}_{j^{*\theta}_{t-1},i}\, p^\theta_{j^{*\theta}_t,i} - c^\theta_{j^{*\theta}_t} \overset{(i)}{\geq} \sum_{t=1}^{\widehat{t}-1} \sum_{i=1}^{m} x^{*t}_{j_{t-1},i}\, p^\theta_{j_t,i} - c^\theta_{j_t} \\ &= \sum_{t=1}^{\widehat{t}-1} \sum_{i=1}^{m} \widehat{x}^t_{j_{t-1},i}\, p^\theta_{j_t,i} - c^\theta_{j_t} = V^\theta(\widehat{\boldsymbol{x}}, \boldsymbol{j}_T) \end{aligned} \tag{27}$$

Note the inequality (i) is by the fact that $\boldsymbol{j}_T^{*\theta}$ is optimal for the follower. Suppose (i) does not hold, then we can come up with a follower response sequence, which plays $j_1, \cdots, j_{t-1}$ for the first $\widehat{t}-1$ rounds followed by non-effort actions (i.e. the action with 0 utility for the agent) afterward, contradicting the optimality $\boldsymbol{j}_T^{*\theta}$.

Combing (26) and (27), we show that it is still optimal for the follower type θ to follow the path $\boldsymbol{j}_T^{*\theta}$ under the contract $\widehat{\boldsymbol{x}}$, proving that $\widehat{\boldsymbol{x}}$ achieves the same expected principal utility as $\boldsymbol{x}^*$. □

In line with our previous outcomes concerning generalized principal-agent problems, we present practical algorithms to address the optimal dynamic contracts and Markovian contracts utilizing `MILP`.

Corollary 2. *The optimal dynamic Markovian contract can be computed by a* `MILP` *with a $O(T|\Theta|mn^2)$ number of continuous variables and $T|\Theta|n$ integer variables; The exact* `DSE` *can be computed by a* `MILP` *with a $O(T|\Theta|mn^T)$ number of continuous variables and $T|\Theta|n$ integer variables.*

This result is directly implied by Theorem 2 and Proposition 2. For example, We can substitute the previous utility functions of the principal and the agent with equations (24) and (25) and get the following `MILP` for computing the `DSE` for dynamic contract design games.

$$\max \quad \sum_{t\in[T]} \sum_{\theta\in\Theta} \mu(\theta) \sum_{\boldsymbol{j}_t\in[n]^t} \sum_{i\in[m]} p^\theta_{j_t,i}(r_i z^{t,\theta}_{\boldsymbol{j}_t} - w^{t,\theta}_{\boldsymbol{j}_t,i})$$

$$
\begin{aligned}
\text{s.t.} \quad & \textstyle\sum_{t'\in[t]} y^{\theta}_{t',j_{t'}} - t + 1 \le z^{t,\theta}_{\boldsymbol{j}_t} \le y^{\theta}_{t',j_{t'}}, && \forall \boldsymbol{j}_t \in [n]^t, t' \in [t], t \in [T], \theta \in \Theta \\
& w^{t,\theta}_{\boldsymbol{j}_t,i} \le x^t_{\boldsymbol{j}_{t-1},i};\ w^{t,\theta}_{\boldsymbol{j}_t,i} \le z^{t,\theta}_{\boldsymbol{j}_t};\ w^{t,\theta}_{\boldsymbol{j}_t,i} \ge x^t_{\boldsymbol{j}_{t-1},i} - (1 - z^{t,\theta}_{\boldsymbol{j}_t})M, && \forall \boldsymbol{j}_t \in [n]^t, t \in [T], \theta \in \Theta, i \in [m] \\
& 0 \le a_\theta - \big(\textstyle\sum_{t,i} x^t_{\boldsymbol{j}_{t-1},i} p^{\theta}_{j_t,i} - c^{\theta}_{j_t}\big) \le M(T - \textstyle\sum_t y^{\theta}_{t,j_t}), && \forall \boldsymbol{j}_T \in [n]^T, \theta \in \Theta \\
& \textstyle\sum_j y^{\theta}_{t,j} = 1, && \forall t \in [T], \theta \in \Theta \qquad (28) \\
& 0 \le \boldsymbol{w}; \quad 0 \le \boldsymbol{x}; \quad 0 \le \boldsymbol{z} \le 1; \quad \boldsymbol{y} \in \{0,1\}
\end{aligned}
$$

Similar to Theorem 1, we can show that the dynamic contract is provably nearly optimal even relative to stronger static strategy spaces for the principal that allow for such things as communication between the principal and the agent. We also have a specific contract design example, which shows RME will be $\Omega(1)$ worse than the average DSE utility, even when there is only a single agent type.

Consider the following contract design example.

P	o_1	o_2
a_0	0	0
a_1	0.5	0.5
a_2	0	1

Table 6: An example contract design game, where P denotes each agent's action a_j's ($j \in \{0, 1, 2\}$) probabilistic distribution over outcomes $\{o_1, o_2\}$. Each action a_i has a corresponding cost vector c_i and the agent's cost vector is $\boldsymbol{c} = [0, 0.1, 0.3]$. In addition, every outcome o_1/o_2 has a corresponding reward $0.5/1$ for the principal.

It can be computed that the optimal single-round contract for the principal is to design a contract as $\boldsymbol{x} = (0, 0.2)$, i.e. pay the agent with payment 0.2 is outcome o_2 is realized, resulting in a utility of 0.65 for the principal where the agent responds by playing action a_1.

On the other hand, for any $T > 1$, the principal can design the contract as follows:

$$
\boldsymbol{x}^t = (0, 0),\ \forall t = 1, \cdots, T-1; \quad \boldsymbol{x}^T = \begin{cases} (0, 0.3T) & \text{if } j_t = a_2\ \forall t < T \\ (0, 0) & \text{otherwise} \end{cases}. \qquad (29)
$$

As a result, the agent will always play action a_2, resulting in a total utility of $0.7T$. There is a gap $0.05T$ between the total principal utility under the optimal static contract and the dynamic contract (29), equivalently $\Omega(1)$ gap between the average DSE utility and RME utility.

Note that in the example of Table 6, the principal can extract the maximal full surplus from the agent (i.e., pay the agent the exact amount of cost he spent such that the agent gets zero utility). It turns out that this is not a coincidence. Next, we show that the principal is always able to extract the maximal full surplus from the agent utilizing the special structural property of the contract design game.

Definition 2 (Payment-dominance Region and Ratio)**.** *Let $j^* = \operatorname{argmax}_j R_j - c_j$ be the agent action that achieves the maximal welfare. The payment dominance region X^* is defined as the set of all contracts whose expected pay on j^* is strictly more than that of any other action, or formally,*

$$
X^* = \{\boldsymbol{x} : \textstyle\sum_i x_i(P_{j^*,i} - P_{j,i}) > 0, \forall j \neq j^*\};
$$

The payment dominance ratio $\mathbf{q}(\boldsymbol{x})$ for any $\boldsymbol{x} \in X^*$ is defined as

$$
q_j(\boldsymbol{x}) = \frac{\sum_i x_i P_{j^*,i}}{\sum_i x_i(P_{j^*,i} - P_{j,i})} \quad \forall j \neq j^*.
$$

Non-emptiness of the payment dominance region X^* guarantees the existence of a contract such that its expected payment of playing the action j^*, which maximizes the social welfare, is greater than the expected payment of other actions. With this definition, we are ready to show that the principal is always able to extract the maximal social welfare (i.e., $\max_j R_j - c_j$) with a dynamic contract when T is large enough and the principal knows the agent's utility function.

Proposition 3. *For any contract design instance with non-empty payment-dominance region X^*, the optimal dynamic contract extracts full surplus within any interaction period $T \geq T_0$ where*

$$T_0 = \left\lceil \min_{\boldsymbol{x} \in X^*} \max_{j \in \{j : c_j < c_{j^*}\}} q_j(\boldsymbol{x})(1 - c_j / c_{j^*}) \right\rceil$$

is an instance-dependent parameter that depends on the payment-dominance ratio.

Note that the set $\{j : c_j < c_{j^*}\}$ is always non-empty, since $\operatorname{argmin}_j c_j$ cannot be j^* due to its social welfare being 0 as defined. Next, we will demonstrate the construction of such a dynamic contract.

Proof. Denote $j^* = \operatorname{argmax}_{j \in [n]} \sum_i P_{j,i} r_i - c_j$, i.e. the agent action that achieves the maximal welfare. We propose the following contract to extract the first-best surplus from the agent.

$$\boldsymbol{x}^t = \boldsymbol{0}, \forall t = 1, \cdots, T-1; \quad \boldsymbol{x}^T = \begin{cases} \boldsymbol{x}^* & \text{if } j_t = j^* \ \forall t < T \\ \boldsymbol{0} & \text{otherwise} \end{cases} . \tag{30}$$

where $\boldsymbol{x}^*$ satisfies the following linear system:

$$\begin{aligned} &\sum_i P_{j^*,i} x_i^* = T c_{j^*}; \\ &\sum_i P_{j,i} x_i^* \leq (T-1) c_{j^*} + c_j, \quad \forall j \neq j^*. \end{aligned} \tag{31}$$

or equivalently,

$$\begin{aligned} &\sum_i P_{j^*,i} x_i^* = T c_{j^*}; \\ &\sum_i x_i^* (P_{j^*,i} - P_{j,i}) \geq c_{j^*} - c_j, \quad \forall j \neq j^*. \end{aligned} \tag{32}$$

By definition 2 of X^*, for any $\widehat{\boldsymbol{x}} \in X^*$, it satisfies the following property:

$$\sum_i \widehat{x}_i (P_{j^*,i} - P_{j,i}) > 0, \ \forall j \neq j^*.$$

Next, we prove there always exists a $\boldsymbol{x}^*$ that satisfies (32) by construction. First of all, we must have $\sum_i \widehat{x}_i (P_{j^*,i} - P_{j,i}) \geq c_{j^*} - c_j$ if $c_{j^*} \leq c_j$. On the other hand, for the actions $j \in \{j' : c_{j^*} > c_{j'}\}$ let $\alpha = \min_{j \in \{j' : c_{j^*} > c_{j'}\}} \frac{\sum_i \widehat{x}_i (P_{j^*,i} - P_{j,i})}{c_{j^*} - c_j}$ and we define $\overline{\boldsymbol{x}} = \widehat{\boldsymbol{x}} / \alpha$. Thus we have

$$\sum_i \overline{x}_i (P_{j^*,i} - P_{j,i}) = \frac{\sum_i \widehat{x}_i (P_{j^*,i} - P_{j,i})}{\alpha} \geq \sum_i \widehat{x}_i (P_{j^*,i} - P_{j,i}) \frac{c_{j^*} - c_j}{\sum_i \widehat{x}_i (P_{j^*,i} - P_{j,i})} = c_{j^*} - c_j,$$

where the inequality is by definition of α. As a result, we have $\overline{\boldsymbol{x}}$ satisfies the *second* constraint of (32).

Finally, we construct $\boldsymbol{x}^*$, which satisfies *both* constraints of (32), based on $\overline{\boldsymbol{x}}$. If $\sum_i \overline{x}_i P_{j^*,i} = Tc_{j^*}$ for some natural number $T \in \mathbb{N} = \{1, 2, 3, \cdots\}$, we already have the existence of $\boldsymbol{x}^* = \overline{\boldsymbol{x}}$ that is feasible to (32). On the other hand, if $\sum_i \overline{x}_i P_{j^*,i} \neq Tc_{j^*}$, we let $\boldsymbol{x}^* = T\frac{c_{j^*}}{\sum_i \overline{x}_i P_{j^*,i}}\overline{\boldsymbol{x}}$ where $T \in \mathbb{N}$. Thus, we have

$$\sum_i P_{j^*,i} x_i^* = Tc_{j^*} \frac{\sum_i \overline{x}_i P_{j^*,i}}{\sum_i \overline{x}_i P_{j^*,i}} = Tc_{j^*}.$$

To make sure $\boldsymbol{x}^*$ still satisfies the second constraint of (32), we must have T is large enough such that $T \geq \frac{\sum_i \overline{x}_i P_{j^*,i}}{c_{j^*}}$ (i.e., $T\frac{c_{j^*}}{\sum_i \overline{x}_i P_{j^*,i}} \geq 1$), which guarantees the left-hand side values (positive for all $j \neq j^*$) of the second constraint of (32) increase. As a result, we have shown the existence of $\boldsymbol{x}^*$ when

$$T \geq \frac{\frac{1}{\alpha}\sum_i \widehat{x}_i P_{j^*,i}}{c_{j^*}} = \max_{j \in \{j' : c_{j^*} > c_{j'}\}} \frac{\sum_i \widehat{x}_i P_{j^*,i}}{\sum_i \widehat{x}_i (P_{j^*,i} - P_{j,i})}(1 - c_j/c_{j^*}) = \max_{j \in \{j' : c_{j^*} > c_{j'}\}} q_j(\widehat{\boldsymbol{x}})(1 - c_j/c_{j^*}),$$

where $\widehat{\boldsymbol{x}}$ is any feasible strategy in X^*. Minimizing over $\widehat{\boldsymbol{x}} \in X^*$ completes the proof of the proposition. □

8 Experiments

To examine the efficacy of the DSE and the efficiency of our proposed algorithm, we perform several experiments using Gurobi 9.5.1 solver on a machine with Ubuntu 20.04.5 LTS operating system, 2 × 18 cores 3.0 GHz processors, and 256GB RAM.

The Dynamic Pricing Game. First, we test our MILP to compute the exact DSE from Section 6 on the dynamic pricing game as described in Section 3 to experimentally verify the no learning theorem. We examine a game with finite buyer value set $V = \{0.4, 0.5, 0.6\}$ and a uniform value distribution. It is easy to compute that the optimal single-round mechanism is to post the Myerson price of 0.4, inducing expected revenue of 0.4. This game can be written as a principal-agent game, whose payoff matrices can be written as follows.

U	j_0	j_1
i_0	0	0.4
i_1	0	0.5
i_2	0	0.6

V^0	j_0	j_1
i_0	0	0
i_1	0	-0.1
i_2	0	-0.2

V^1	j_0	j_1
i_0	0	0.1
i_1	0	0
i_2	0	-0.1

V^2	j_0	j_1
i_0	0	0.2
i_1	0	0.1
i_2	0	0

Here, U represents the seller's utility matrix; V^0, V^1, V^2 represent the buyer's utility matrix when $v = 0.4, v = 0.5, v = 0.6$ accordingly; i_0, i_1, i_2 represent the seller sets a posted price $0.4, 0.5, 0.6$; j_0, j_1 means the buyer rejects or accepts the item.

We run experiments for varying $T = 1, \cdots, 10$, and always observe $u_l^{\mathtt{DSE}} = 0.4T$. That is, the average optimal dynamic pricing utility is indeed the same as the optimal static revenue.

Battle of the Sexes. In this game, two players receive higher utilities when they take the same action, though one of them enjoys this action more than the other. We consider a variant of this game with a principal-agent structure and two uniformly distributed agent types V^0, V^1 as follows:

R	🕺	⛹️‍♂️
💃	1	0
⛹️‍♀️	0	0.5

C^0	🕺	⛹️‍♂️
💃	0.5	0
⛹️‍♀️	0	1

C^1	🕺	⛹️‍♂️
💃	0	0.5
⛹️‍♀️	0	1

	$u_l^{\mathtt{DSE}}/T$	$u_l^{\mathtt{RME}}$
T = 1	0.5	**0.625**
T = 2	**0.75**	0.625
T = 3	**0.75**	0.625
T = 4	**0.75**	0.625
T = 5	**0.75**	0.625

where V^0 is a type as in a standard battle of the sexes game whereas V^1 is an agent type who "stubbornly" insists on the *Basketball* action though they would enjoy it more if the principal also picks this action. The right-most subtable shows the principal's expected average utility in the dynamic setup for different numbers of interaction rounds.

Stackelberg Security Game (SSG). We consider a specific SSG, where there is a defender (principal) trying to protect 2 targets with 1 resource from the attack of an attacker (agent). Specifically, there are two uniformly distributed agent types whose utility information is as follows:

R	t_0	t_1
t_0	1	0
t_1	0	1

C^0	t_0	t_1
t_0	0	0.5
t_1	1	0

C^1	t_0	t_1
t_0	0	1
t_1	0.5	0

	u_l^{DSE}/T	u_l^{RME}
$\mathrm{T}=1$	0.5	0.5
$\mathrm{T}=2$	**0.562**	0.5
$\mathrm{T}=3$	**0.571**	0.5
$\mathrm{T}=4$	**0.583**	0.5
$\mathrm{T}=5$	**0.588**	0.5

where each row represents the principal's action (i.e. protecting the target) and each column represents the agent's action (i.e. attacking the target). Follower type V^0 prefers target t_0 while V^1 prefers target t_1. The right-most subtable shows the principal's expected average utility in the dynamic setup for different numbers of interaction rounds.

From the experimental results, we can see that the utility improvement of DSE compared to RME is quite evident in both games. Note that when $T = 1$, the optimal DSE is exactly the BSE, which is always less than or equal to RME. We also want to highlight the increasing property of average u_l^{DSE} in the presented results. As shown by the results, though the u_l^{DSE} might be lower than u_l^{RME} in the beginning rounds, it catches up with u_l^{RME} as T increases, consistent with Theorem 1. Another interesting observation is that u_l^{DSE} converged within two rounds in the battle of sexes game. In the SSG, u_l^{DSE} did not converge within five rounds. We remark that the convergence of u_l^{DSE} is an interesting open question for future research. Finally, in appendix B, we also increase the size of some games (e.g. the number of types in the battle of sexes game and the number of targets in the SSG) and present additional experimental results on the Game of Chicken, another classical structured game.

Experimental Results on Randomized Games. Finally, we conduct experiments on random game instances in which each agent's utility is uniformly drawn from $[0, 1]$. As in the previous experiments, we compare the average principal utility between DSE and the optimal static policy. In addition, to demonstrate the performance of our Markovian policy, we also compare the runtime and average principal utility of solving the optimal Markovian policy, versus solving the optimal dynamic policy. The results from T=$1, 2, 3$ have been averaged over 50 random instances and the standard deviation is reported in the table. As for $T = 4, 5$, the runtime is high and the advantage of Markovian policy is significant (note the runtime of DSE when $T = 4$ is already much higher than the Markovian method when $T = 5$), so we only test the first 10 random instances out of the 50 randome instances, which explains why the RME performs differently when $T = 4, 5$. The advantage of DSE over RME on the average principal utility is also consistent with our observation in structured games. Last but not least, we also highlight that Markovian policies achieve nearly optimal utility while being much more computationally efficient compared to the optimal dynamic policies.

9 Conclusion

In this paper, we demonstrate, in surprising contrast to the No Learning Theorem in classic pricing games, the possibility of learning from a strategic agent in general repeated principal-agent problems. Moreover, we develop a novel methodology for finding and approximating the optimal dynamic policy for the principal, and we give a strong lower bound on the performance of the optimal dynamic policy. We also have demonstrated experimentally that the optimal dynamic policy often significantly exceeds the theoretical lower bound. Our results open the possibilities for many other interesting questions. For example, given the NP-hardness of

	Runtime		Average Utility		
	Markovian	DSE	Markovian	DSE	RME
T = 1	**0.073** ± 0.05	0.094 ± 0.06	0.880 ± 0.04	0.880 ± 0.04	**0.935** ± 0.02
T = 2	1.466 ± 0.51	**1.403** ± 0.66	0.933 ± 0.02	0.933 ± 0.02	**0.935** ± 0.02
T = 3	**5.678** ± 2.26	54.34 ± 5.46	0.944 ± 0.02	**0.951** ± 0.01	0.935 ± 0.02
T = 4	**43.70** ± 30.7	2435.7 ± 484	0.949 ± 0.01	**0.956** ± 0.01	0.942 ± 0.02
T = 5	658.8 ± 778	N/A	0.951 ± 0.02	N/A	0.942 ± 0.02

Table 8: Running time (columns 2-3 with the unit: second) and average utility (columns 4-6) for random game instances with $m = 10, n = 5, |\Theta| = 2$, where "N/A" means the algorithm can not return a solution within 3 hours.

computing the BSE, is computing the DSE NP-hard or maybe in PSPACE-hard? Other interesting questions include studying the convergence property of $u_l^{\texttt{DSE}}$ and applying the dynamic solution concept to randomized menu policies.

Acknowledgment

This work was supported in part by ARO through W911NF-23-1-0030 and NSF through CCF-2303372.

References

Gagan Aggarwal, Ashish Goel, and Rajeev Motwani. Truthful auctions for pricing search keywords. In *Proceedings of the 7th ACM Conference on Electronic Commerce*, pages 1–7, 2006. 1

Tal Alon, Paul Dütting, and Inbal Talgam-Cohen. Contracts with private cost per unit-of-effort. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pages 52–69, 2021. 1.1, 7

Ingo Althöfer. On sparse approximations to randomized strategies and convex combinations. *Linear Algebra and its Applications*, 199:339–355, 1994. 2, 2

Kareem Amin, Afshin Rostamizadeh, and Umar Syed. Learning prices for repeated auctions with strategic buyers. In *Advances in Neural Information Processing Systems*, pages 1169–1177, 2013. 1, 1.1, 2

Kareem Amin, Afshin Rostamizadeh, and Umar Syed. Repeated contextual auctions with strategic buyers. In *Advances in Neural Information Processing Systems*, pages 622–630, 2014. 1, 1.1

Maria-Florina Balcan, Amit Daniely, Ruta Mehta, Ruth Urner, and Vijay V Vazirani. Learning economic parameters from revealed preferences. In *International Conference on Web and Internet Economics*, pages 338–353. Springer, 2014. 1.1

Eyal Beigman and Rakesh Vohra. Learning from revealed preference. In *Proceedings of the 7th ACM Conference on Electronic Commerce*, pages 36–42, 2006. 2

Alain Bensoussan, Shaokuan Chen, and Suresh P Sethi. The maximum principle for global solutions of stochastic stackelberg differential games. *SIAM Journal on Control and Optimization*, 53(4):1956–1981, 2015. 1.1

Patrick Bolton and Mathias Dewatripont. *Contract theory*. MIT press, 2004. 1

Matteo Castiglioni, Alberto Marchesi, and Nicola Gatti. Bayesian agency: Linear versus tractable contracts. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pages 285–286, 2021. 1.1, 7

Matteo Castiglioni, Alberto Marchesi, and Nicola Gatti. Designing menus of contracts efficiently: The power of randomization. In *EC '22: The 23rd ACM Conference on Economics and Computation*, pages 705–735. ACM, 2022. doi: 10.1145/3490486.3538270. 1.1, 5

Yanling Chang, Alan L Erera, and Chelsea C White. A leader–follower partially observed, multiobjective markov game. *Annals of Operations Research*, 235(1):103–128, 2015. 1.1

Alon Cohen, Argyrios Deligkas, and Moran Koren. Learning approximately optimal contracts. In *Algorithmic Game Theory: 15th International Symposium, SAGT 2022, Colchester, UK, September 12–15, 2022, Proceedings*, pages 331–346. Springer, 2022. 1.1

Vincent Conitzer and Tuomas Sandholm. Computing the optimal strategy to commit to. In *Proceedings of the 7th ACM conference on Electronic commerce*, pages 82–90, 2006. 1, 2, 6

IBM ILOG Cplex. V12. 1: User's manual for cplex. *International Business Machines Corporation*, 46(53): 157, 2009. 6

Quinlan Dawkins, Minbiao Han, and Haifeng Xu. The limits of optimal pricing in the dark. *Advances in Neural Information Processing Systems*, 34:26649–26660, 2021. 1, 1.1

Quinlan Dawkins, Minbiao Han, and Haifeng Xu. First-order convex fitting and its application to economics and optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 6480–6487, 2022. 1, 1.1

Yuan Deng, Jon Schneider, and Balasubramanian Sivan. Strategizing against no-regret learners. In *Advances in Neural Information Processing Systems*, pages 1577–1585, 2019. 1.1

Nikhil R Devanur, Yuval Peres, and Balasubramanian Sivan. Perfect bayesian equilibria in repeated sales. In *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*, pages 983–1002. SIAM, 2014. 1, 1.1, 2, 3

Paul Dütting, Tim Roughgarden, and Inbal Talgam-Cohen. Simple versus optimal contracts. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 369–387, 2019. 1.1, 7

Drew Fudenberg and David Levine. Subgame-perfect equilibria of finite-and infinite-horizon games. *Journal of Economic Theory*, 31(2):251–268, 1983. 1.1

Jiarui Gan, Minbiao Han, Jibang Wu, and Haifeng Xu. Optimal coordination in generalized principal-agent problems: A revisit and extensions. *arXiv preprint arXiv:2209.01146*, 2022. 5

Jiarui Gan, Minbiao Han, Jibang Wu, and Haifeng Xu. Robust stackelberg equilibria. In *Proceedings of the 24th ACM Conference on Economics and Computation*, 2023. 5

Denizalp Goktas, Jiayi Zhao, and Amy Greenwald. Zero-sum stochastic stackelberg games. *arXiv preprint arXiv:2211.13847*, 2022. 1.1

Sanford J Grossman and Oliver D Hart. An analysis of the principal-agent problem. In *Foundations of insurance economics*, pages 302–340. Springer, 1992. 4.2

Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2022. URL https://www.gurobi.com. 6

Nika Haghtalab, Thodoris Lykouris, Sloan Nietert, and Alexander Wei. Learning in stackelberg games with non-myopic agents. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, pages 917–918, 2022. 1.1

Sergiu Hart. Games in extensive and strategic forms. *Handbook of game theory with economic applications*, 1:19–40, 1992. 1.1

Chien-Ju Ho, Aleksandrs Slivkins, and Jennifer Wortman Vaughan. Adaptive contract design for crowdsourcing markets: Bandit algorithms for repeated principal-agent problems. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pages 359–376, 2014. 1.1

Bengt Holmström. Moral hazard and observability. *The Bell journal of economics*, pages 74–91, 1979. 4.2

Nicole Immorlica, Brendan Lucier, Emmanouil Pountourakis, and Samuel Taggart. Repeated sales with multiple strategic buyers. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, EC '17, page 167–168, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450345279. doi: 10.1145/3033274.3085130. URL https://doi.org/10.1145/3033274.3085130. 1, 1.1

Sham M Kakade, Ilan Lobel, and Hamid Nazerzadeh. Optimal dynamic mechanism design and the virtual-pivot mechanism. *Operations Research*, 61(4):837–854, 2013. 1.1

Niklas Lauffer, Mahsa Ghasemi, Abolfazl Hashemi, Yagiz Savas, and Ufuk Topcu. No-regret learning in dynamic stackelberg games. *arXiv preprint arXiv:2202.04786*, 2022. 1.1

Joshua Letchford, Vincent Conitzer, and Kamesh Munagala. Learning and approximating the optimal strategy to commit to. In *International symposium on algorithmic game theory*, pages 250–262. Springer, 2009. 1.1

Tao Li and Suresh P Sethi. A review of dynamic stackelberg game models. *Discrete & Continuous Dynamical Systems-B*, 22(1):125, 2017. 1.1

Eric Maskin and Jean Tirole. Markov perfect equilibrium: I. observable actions. *Journal of Economic Theory*, 100(2):191–219, 2001. 1.1

Vahab Mirrokni, Renato Paes Leme, Pingzhong Tang, and Song Zuo. Non-clairvoyant dynamic mechanism design. *Econometrica*, 88(5):1939–1963, 2020. 2

John Moore and Rafael Repullo. Subgame perfect implementation. *Econometrica: Journal of the Econometric Society*, pages 1191–1220, 1988. 1.1

Roger B Myerson. Optimal auction design. *Mathematics of operations research*, 6(1):58–73, 1981. 1, 3, 3.1, 4.1, 5, 4.1

Roger B Myerson. Optimal coordination mechanisms in generalized principal–agent problems. *Journal of mathematical economics*, 10(1):67–81, 1982. 5

Praveen Paruchuri, Jonathan P Pearce, Janusz Marecki, Milind Tambe, Fernando Ordonez, and Sarit Kraus. Playing games for security: An efficient exact algorithm for solving bayesian stackelberg games. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 2*, pages 895–902, 2008. 1

Alessandro Pavan, Ilya Segal, and Juuso Toikka. Dynamic mechanism design: A myersonian approach. *Econometrica*, 82(2):601–653, 2014. 1.1, 2

Binghui Peng, Weiran Shen, Pingzhong Tang, and Song Zuo. Learning optimal strategies to commit to. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 2149–2156, 2019. 1.1

Aaron Roth, Jonathan Ullman, and Zhiwei Steven Wu. Watch and learn: Optimizing from revealed preferences feedback. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 949–962, 2016. 1, 1.1, 2

Tim Roughgarden. Stackelberg scheduling strategies. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pages 104–113, 2001. 1

Lloyd S Shapley. Stochastic games. *Proceedings of the national academy of sciences*, 39(10):1095–1100, 1953. 1.1

Arunesh Sinha, Fei Fang, Bo An, Christopher Kiekintveld, and Milind Tambe. Stackelberg security games: Looking beyond a decade of success. IJCAI, 2018. 2, 5

Aleksandrs Slivkins et al. Introduction to multi-armed bandits. *Foundations and Trends® in Machine Learning*, 12(1-2):1–286, 2019. 1.1

Arsenii Vanunts and Alexey Drutsa. Optimal pricing in repeated posted-price auctions with different patience of the seller and the buyer. *Advances in Neural Information Processing Systems*, 32, 2019. 1, 1.1, 2, 3, 1

Hal R Varian. Online ad auctions. *American Economic Review*, 99(2):430–34, 2009. 1

William Vickrey. Counterspeculation, auctions, and competitive sealed tenders. *The Journal of finance*, 16 (1):8–37, 1961. 1

Bernhard Von Stengel and Shmuel Zamir. Leadership with commitment to mixed strategies. Technical report, Citeseer, 2004. 1

Yevgeniy Vorobeychik and Satinder Singh. Computing stackelberg equilibria in discounted stochastic games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, pages 1478–1484, 2012. 1.1

Quoc-Liem Vu, Zane Alumbaugh, Ryan Ching, Quanchen Ding, Arnav Mahajan, Benjamin Chasnov, Sam Burden, and Lillian J Ratliff. Stackelberg policy gradient: Evaluating the performance of leaders and followers. In *ICLR 2022 Workshop on Gamification and Multiagent Solutions*, 2022. 1.1

Jibang Wu, Weiran Shen, Fei Fang, and Haifeng Xu. Inverse game theory for stackelberg games: the blessing of bounded rationality. In *Advances in Neural Information Processing Systems*, 2022. 5

Rong Yang, Benjamin J Ford, Milind Tambe, and Andrew Lemieux. Adaptive resource allocation for wildlife protection against illegal poachers. In *Aamas*, pages 453–460, 2014. 1

Banghua Zhu, Stephen Bates, Zhuoran Yang, Yixin Wang, Jiantao Jiao, and Michael I Jordan. The sample complexity of online contract design. *arXiv preprint arXiv:2211.05732*, 2022. 1.1

A Omitted Details in the Proof of Theorem 1

Missing Proof. Here we present the omitted proof of lemma 1 and details of Step 3 to show that there always exists such an IC menu that is $O(\sqrt{\frac{\log|\Theta|}{T\delta^2}})$ optimal and can offset the distorted incentives.

Lemma 1. *For Stackelberg games with inducibility gap δ and $T \geq \Omega(\bar{t}^2)$, there always exists a randomized menu that is $O(\sqrt{\frac{\log|\Theta|}{T\delta^2}})$ optimal and $O(\sqrt{\frac{\log|\Theta|}{T}})$-strictly IC.*

Proof. We prove the lemma by explicitly constructing a randomized menu $\langle \boldsymbol{p}, \boldsymbol{x}\rangle$ and show that $\langle \boldsymbol{p}, \boldsymbol{x}\rangle$ is $O(\sqrt{\frac{\log|\Theta|}{T}})$-strictly IC and achieves $(1-\sqrt{\frac{\log|\Theta|}{T\delta^2}})u_l^{\mathtt{RME}}$. Specifically, denote the $\mathtt{RME}$ as $\langle \boldsymbol{p}^*, \boldsymbol{x}^*\rangle$ and the δ-strictly IC randomized menu as $\langle \boldsymbol{p}^\delta, \boldsymbol{x}^\delta\rangle$, we construct the new randomized menu $\langle \boldsymbol{p}, \boldsymbol{x}\rangle$ as follows

$$\begin{aligned}
p_{\theta,j} &= (1-\sqrt{\frac{\log|\Theta|}{T\delta^2}})p^*_{\theta,j} + (\sqrt{\frac{\log|\Theta|}{T\delta^2}})p^\delta_{\theta,j} \quad \text{and} \\
\boldsymbol{x}_{\theta,j} &= \frac{(1-\sqrt{\frac{\log|\Theta|}{T\delta^2}})p^*_{\theta,j}}{p_{\theta,j}}\boldsymbol{x}^*_{\theta,j} + \frac{(\sqrt{\frac{\log|\Theta|}{T\delta^2}})p^\delta_{\theta,j}}{p_{\theta,j}}\boldsymbol{x}^\delta_{\theta,j} \quad \forall \theta, j
\end{aligned} \tag{33}$$

For every follower type θ, by definition

$$\begin{aligned}
&\sum_j p^\delta_{\theta,j} u^\theta_f(\boldsymbol{x}^\delta_{\theta,j}, j) \geq \sum_j p^\delta_{\theta',j} \max_{j'} u^\theta_f(\boldsymbol{x}^\delta_{\theta',j}, j') + \delta, \text{ and} \\
&\sum_j p^*_{\theta,j} u^\theta_f(\boldsymbol{x}^*_{\theta,j}, j) \geq \sum_j p^*_{\theta',j} \max_{j'} u^\theta_f(\boldsymbol{x}^*_{\theta',j}, j'), \forall\, \theta' \neq \theta.
\end{aligned} \tag{34}$$

As a result, we have

$$\begin{aligned}
&\sum_j p_{\theta,j} u^\theta_f(\boldsymbol{x}_{\theta,j}, j) = \\
&\sum_j (1-\sqrt{\frac{\log|\Theta|}{T\delta^2}})p^*_{\theta,j} u^\theta_f(\boldsymbol{x}^*_{\theta,j}, j) + (\sqrt{\frac{\log|\Theta|}{T\delta^2}})p^\delta_{\theta,j} u^\theta_f(\boldsymbol{x}^\delta_{\theta,j}, j) \\
&\geq \sum_j \max_{j'} (1-\sqrt{\frac{\log|\Theta|}{T\delta^2}})p^*_{\theta',j} u^\theta_f(\boldsymbol{x}^*_{\theta',j}, j') + \\
&\sum_j \max_{j'} \sqrt{\frac{\log|\Theta|}{T\delta^2}} p^\delta_{\theta',j} u^\theta_f(\boldsymbol{x}^\delta_{\theta',j}, j') + \delta\sqrt{\frac{\log|\Theta|}{T\delta^2}} \\
&\geq \sum_j \max_{j'} \big((1-\sqrt{\frac{\log|\Theta|}{T\delta^2}})p^*_{\theta',j} u^\theta_f(\boldsymbol{x}^*_{\theta',j}, j') + \\
&\sqrt{\frac{\log|\Theta|}{T\delta^2}} p^\delta_{\theta',j} u^\theta_f(\boldsymbol{x}^\delta_{\theta',j}, j')\big) + \delta\sqrt{\frac{\log|\Theta|}{T\delta^2}} \\
&= \sum_j \max_{j'} p_{\theta',j} u^\theta_f(\boldsymbol{x}_{\theta',j}, j') + \sqrt{\frac{\log|\Theta|}{T}}, \quad \forall \theta' \neq \theta.
\end{aligned}$$

where the first inequality is by (34) and the second inequality is by merging two $\max$'s into one $\max$, proving the constructed mechanism $\langle \boldsymbol{p}, \boldsymbol{x}\rangle$ is $O(\sqrt{\frac{\log|\Theta|}{T}})$-strictly IC.

Next, we write out the leader's expected utility for every type θ

$$\begin{aligned}&\sum_j p_{\theta,j} u_l(\boldsymbol{x}_{\theta,j}, j) = \\ &\sum_j (1-\sqrt{\frac{\log|\Theta|}{T\delta^2}}) p^*_{\theta,j} u_l(\boldsymbol{x}^*_{\theta,j}, j) + (\sqrt{\frac{\log|\Theta|}{T\delta^2}}) p^\delta_{\theta,j} u_l(\boldsymbol{x}^\delta_{\theta,j}, j) \\ &\geq \sum_j (1-\sqrt{\frac{\log|\Theta|}{T\delta^2}}) p^*_{\theta,j} u_l(\boldsymbol{x}^*_{\theta,j}, j)\end{aligned} \tag{35}$$

As a result, we have the expected utility of $\langle \boldsymbol{p}, \boldsymbol{x}\rangle$ as $u_l^{\langle \boldsymbol{p}, \boldsymbol{x}\rangle} \geq (1-\sqrt{\frac{\log|\Theta|}{T\delta^2}}) u_l^{\text{RME}}$, proving the lemma. □

Step 3: Existence of $(T-\bar{t})$-uniform, $O(\sqrt{\frac{\log|\Theta|}{T\delta^2}})$ optimal, and IC menu.

Lemma 2 ([Althöfer, 1994]). *For any $\epsilon > 0$ and any $\{p_{\theta,j}, \boldsymbol{x}_{\theta,j}\}_{j\in[n]}$, there exists a k-uniform $\overline{\boldsymbol{p}}_\theta$ with $k = \lceil \frac{\log 2(|\Theta|+1)}{2\epsilon^2} \rceil$ such that*

$$\begin{aligned}&|\sum_j p_{\theta,j} u_l(\boldsymbol{x}_{\theta,j}, j) - \sum_j \overline{p}_{\theta,j} u_l(\boldsymbol{x}_{\theta,j}, j)| \leq \epsilon, \textit{ and for all } \theta' \\ &|\sum_j p_{\theta,j} \max_{j'} u_f^{\theta'}(\boldsymbol{x}_{\theta,j}, j') - \sum_j \overline{p}_{\theta,j} \max_{j'} u_f^{\theta'}(\boldsymbol{x}_{\theta,j}, j')| \leq \epsilon.\end{aligned}$$

By Lemma 2, there always exists an approximation of $\boldsymbol{p}_\theta$ denoted as $\overline{\boldsymbol{p}}_\theta$ for all θ, which is a $(T-\bar{t})$-uniform distribution and can be precisely fulfilled in the $T-\bar{t}$ rounds. Next, we show that the new randomized menu $\langle \overline{\boldsymbol{p}}, \boldsymbol{x}\rangle$ is IC and $O(\sqrt{\frac{\log|\Theta|}{T\delta^2}})$ optimal.

Since the original $\langle \boldsymbol{p}, \boldsymbol{x}\rangle$ is $O(\sqrt{\frac{\log|\Theta|}{T}})$-strictly IC randomized menu, we have

$$\sum_j p_{\theta,j} u_f^\theta(\boldsymbol{x}_{\theta,j}, j) \geq \sum_j p_{\theta',j} \max_{j'} u_f^\theta(\boldsymbol{x}_{\theta',j}, j') + O(\sqrt{\frac{\log|\Theta|}{T}}), \tag{36}$$

for all θ and $\theta' \neq \theta$. When we approximate $\boldsymbol{p}_\theta$ with a $(T-\bar{t})$-uniform distribution $\overline{\boldsymbol{p}}_\theta$, there exists an approximate error $\epsilon = \sqrt{\frac{\log 2(|\Theta|+1)}{2(T-\bar{t})}} = O(\sqrt{\frac{\log|\Theta|}{T}})$ for all players' utilities under the randomized strategy $\{\overline{p}_{\theta,j}, \boldsymbol{x}_{\theta,j}\}_{j\in[n]}$. As a result, for the $(T-\bar{t})$-uniform distribution $\overline{\boldsymbol{p}}_\theta$, we have following observations by Lemma 2 for all θ and $\theta' \neq \theta$:

$$\begin{aligned}&\sum_j \overline{p}_{\theta,j} u_f^\theta(\boldsymbol{x}_{\theta,j}, j) \geq \sum_j p_{\theta,j} u_f^\theta(\boldsymbol{x}_{\theta,j}, j) - \epsilon \quad \text{and} \\ &\sum_j \overline{p}_{\theta',j} \max_{j'} u_f^\theta(\boldsymbol{x}_{\theta',j}, j') \leq \sum_j p_{\theta',j} \max_{j'} u_f^\theta(\boldsymbol{x}_{\theta',j}, j') + \epsilon\end{aligned} \tag{37}$$

Therefore, we can combine (36) and (37), where $\epsilon = O(\sqrt{\frac{\log|\Theta|}{T}})$, to get

$$\begin{aligned}&\sum_j \overline{p}_{\theta,j} u_f^\theta(\boldsymbol{x}_{\theta,j}, j) \geq \sum_j p_{\theta,j} u_f^\theta(\boldsymbol{x}_{\theta,j}, j) - O(\sqrt{\frac{\log|\Theta|}{T}}) \quad \text{and} \\ &\sum_j \overline{p}_{\theta',j} \max_{j'} u_f^\theta(\boldsymbol{x}_{\theta',j}, j') \leq \sum_j p_{\theta,j} u_f^\theta(\boldsymbol{x}_{\theta,j}, j) - O(\sqrt{\frac{\log|\Theta|}{T}})\end{aligned} \tag{38}$$

and equivalently,

$$\sum_j \overline{p}_{\theta,j} u_f^\theta(\boldsymbol{x}_{\theta,j}, j) \geq \sum_j \overline{p}_{\theta',j} \max_{j'} u_f^\theta(\boldsymbol{x}_{\theta',j}, j'), \forall\theta, \tag{39}$$

proving the new randomized menu $\langle \overline{\boldsymbol{p}}, \boldsymbol{x} \rangle$ is IC.

Finally, the leader's utility of the new randomized menu $\langle \overline{\boldsymbol{p}}, \boldsymbol{x} \rangle$ is also bounded by Lemma 2 as follows

$$\begin{aligned} &\sum_j \overline{p}_{\theta,j} u_l(\boldsymbol{x}_{\theta,j}, j) \geq \sum_j p_{\theta,j} u_l(\boldsymbol{x}_{\theta,j}, j) - \epsilon, \\ &= \sum_j p_{\theta,j} u_l(\boldsymbol{x}_{\theta,j}, j) - O(\sqrt{\frac{\log|\Theta|}{T}}) \quad \forall\theta. \end{aligned} \tag{40}$$

Since by construction, the original randomized menu $\langle \boldsymbol{p}, \boldsymbol{x} \rangle$ is $O(\sqrt{\frac{\log|\Theta|}{T\delta^2}})$ optimal, we have

$$\sum_j p_{\theta,j} u_l(\boldsymbol{x}_{\theta,j}, j) \geq \sum_j p^*_{\theta,j} u_l(\boldsymbol{x}^*_{\theta,j}, j) - O(\sqrt{\frac{\log|\Theta|}{T\delta^2}}), \forall\theta. \tag{41}$$

Combining (40) and (41), we also have the new randomized menu $\langle \overline{\boldsymbol{p}}, \boldsymbol{x} \rangle$ is $O(\sqrt{\frac{\log|\Theta|}{T\delta^2}})$ optimal, proving Step 3. Since a randomized menu with $(T - \overline{t})$-uniform distribution can be precisely simulated by a dynamic policy whereas u_l^{DSE} denotes the leader utility under the optimal dynamic policy, we have $\frac{u_l^{\mathrm{DSE}}}{T} \geq u_l^{\langle \overline{\boldsymbol{p}}, \boldsymbol{x} \rangle}$ and proved $\frac{u_l^{\mathrm{DSE}}}{T} \geq u_l^{\mathrm{RME}} - O(\sqrt{\frac{\log|\Theta|}{T\delta^2}})$ in the Theorem. □

B Additional Experimental Results

B.1 Additional Results on Structured Games

Game of Chicken. The *game of chicken* models the situation in which two players use a shared resource and a collision happens when both players use the resource. We consider a variant of this game with a leader-follower structure and two uniformly distributed follower types C^0, C^1: where the first row/column action represents "giving up the resource" whereas the second row/column action presents "using the resource".

R	🛑	🟢
🛑	0.5	0
🟢	1	-1

C^0	🛑	🟢
🛑	0.5	1
🟢	0	-1

C^1	🛑	🟢
🛑	0.5	1
🟢	-1	0

Type C^0 is the type as in the classic game of chicken whereas C^1 is an "aggressive" type who strongly prefers using the resource. Table 9 shows the leader's expected average utility in the dynamic setup for different numbers of interaction rounds.

	u_l^{DSE}/T	u_l^{RME}
T = 1	1/6	1/6
T = 2	**0.5**	1/6
T = 3	**0.574**	1/6
T = 4	**0.597**	1/6
T = 5	**0.611**	1/6

Table 9: average leader utility per round for the game of chicken.

Stackelberg security game. Additionally, we consider the case where the leader has one more target that needs to be protected (i.e. more actions for both the leader and the follower). Specifically, consider the following utility matrices for both agents.

R	t_0	t_1	t_2
t_0	1	0	0
t_1	0	1	0
t_2	0	0	1

C^0	t_0	t_1	t_2
t_0	0	0.5	0.5
t_1	1	0	0.5
t_2	1	0.5	0

C^1	t_0	t_1	t_2
t_0	0	1	0.5
t_1	0.5	0	0.5
t_2	0.5	1	0

C^2	t_0	t_1	t_2
t_0	0	0.5	1
t_1	0.5	0	1
t_2	0.5	0.5	0

The leader has 1 unit of resource to protect three targets from the attacker whose type is drawn uniformly from $\{C^0, C^1, C^2\}$. Each follower type $C^i, i \in \{0, 1, 2\}$ prefers target t_i. Table 12 shows the leader's expected average utility in the dynamic setup for different numbers of interaction rounds. Note the leader's expected average utility dropped overall compared to the result in the main paper, which is reasonable since the leader has to use the same resource to protect more targets.

	$u_l^{\texttt{DSE}}/T$	$u_l^{\texttt{RME}}$
T = 1	1/3	1/3
T = 2	**0.444**	1/3
T = 3	**0.467**	1/3
T = 4	**0.479**	1/3
T = 5	**0.493**	1/3

Table 12: average leader utility per round for the SSG.