Learning a Game by Paying the Agents*

Brian Hu Zhang[†] Tao Lin[‡] Yiling Chen[§] Tuomas Sandholm[¶]

May 24, 2026

Abstract

We study the problem of learning the utility functions of no-regret learning agents in a repeated normal-form game. Differing from most prior literature, we introduce a principal with the power to observe the agents playing the game, send agents signals, and give agents *payments* as a function of their actions. We show that the principal can, using a number of rounds polynomial in the size of the game, learn the utility functions of all agents to any desired precision $\varepsilon > 0$, for *any* no-regret learning algorithms of the agents. Our main technique is to formulate a zero-sum game between the principal and the agents, where the principal's strategy space is the set of all payment functions. Finally, we discuss implications for the problem of *steering* agents to a desired equilibrium: in particular, we introduce, using our utility-learning algorithm as a subroutine, the first algorithm for steering arbitrary no-regret learning agents without prior knowledge of their utilities.

1 Introduction

Most literature on game theory aims to understand the behavior of agents in a game given the preferences of the agents. That is, *knowing* what the agents want, how will they act? In this paper, we consider the inverse of this problem: if all we can observe is how agents act, can we infer what they want, that is, their utility functions? The problem of infering agents' utility functions from their behavior is known as "learning from revealed preferences" (e.g., Beigman and Vohra [2006], Zadimoghaddam and Roth [2012]) or "inverse game theory" (e.g., Kuleshov and Schrijvers [2015]). Despite a long history, these two lines of work have some limitations. First, they often assume that the observed behavior of the agents is *(Nash) equilibrium behavior*. This is arguably unrealistic, especially because Nash equilibria are PPAD-hard to compute [Daskalakis et al., 2006, Chen et al., 2009]. Second, they mostly focus on *passive* problems, aiming to learn agents' utility functions from a fixed set of behavioral data. Often, this creates trivial impossibility results, stemming from the fact that the behavioral data available is simply not enough to determine the agents' preferences.

In this paper, we consider an *active, non-equilibrium* inverse game problem. A principal observes the actions of agents playing an unknown repeated game, and seeks to learn the agents' utility functions from those observations. We do not assume the agents to play equilibria. Instead, they can use *any no-regret algorithms* to learn to choose actions. The principal can actively modify the unknown game by giving *payments* to the agents as a function of their play, as well as providing *signals*, in the spirit of correlated equilibria [Aumann, 1974]. The signaling scheme and payment scheme can change from round to round, depending on the agents' past actions. Under this setup, we ask:

Can a principal learn the utility functions of no-regret learning agents?

We will give a positive answer to this question, by designing an algorithm for the principal to learn the game via payments and signals.

*The full version of this work is published at ICLR 2026 [Zhang et al., 2026]. This is a 10-page version presented at the "Easy Peasy" Workshop at EC 2026.

[†]Massachusetts Institute of Technology, zhangbh@csail.mit.edu

[‡]Microsoft Research & The Chinese University of Hong Kong, Shenzhen, lintao@cuhk.edu.cn

[§]Harvard University.

[¶]Carnegie Mellon University & Strategy Robot, Inc., Strategic Machine, Inc., Optimized Markets, Inc.

1.1 Overview of our results

In our model, agents play a fixed normal-form game Γ repeatedly over T rounds, using arbitrary no-regret learning algorithms. A principal, initially knowing nothing about the agents' utility functions, can give non-negative *payments* to the agents, that get added to the agents' utilities, to influence the agents' behavior. The principal aims to learn the utility functions of all agents to a given target precision $\varepsilon > 0$, based on the actions taken by the agents.

Our main result is an efficient algorithm for the principal to learn the game by paying no-regret learning agents. Let n be the number of agents, m_i be the number of actions of each agent i, and $M = \prod_i m_i$ be the total number of action profiles. Assume that each agent's regret in T rounds is at most $\mathcal{O}(\sqrt{T})$ (satisfied by typical no-regret algorithms). Our algorithm learns the game in polynomially many rounds:

Theorem 1.1 (Informal version of Theorems 4.3 and 4.4)**.** *There exists a principal algorithm that learns a game to precision ε in $M^{\mathcal{O}(1)}/\varepsilon^2$ rounds, for **any** no-regret agents. This is tight up to the constant in $\mathcal{O}$.*

The main idea of our algorithm is surprisingly simple but powerful. In the single-agent case, we let the principal play a zero-sum game with the agent, where the agent chooses actions to maximize its reward, which is utility plus payment, while the principal chooses payments to minimize the agent's reward. This game admits a unique equilibrium where the principal's payment function is equal to the negation of the agent's utility function. By running no-regret algorithms against each other, the principal and agent can reach such an equilibrium, so the negative average payment function becomes an accurate estimate of the agent's utility function. In the multi-agent case, we use signals to separate the learning problems for different agents, reducing the problem to the single-agent case.

We then turn to a motivating application, which is the problem of *steering* no-regret learners to desirable outcomes, introduced by Zhang et al. [2024]. Departing from them, we do not require the principal to know the agents' utility functions. We define a solution concept called *correlated equilibrium with payments* (CEP), in which the principal has a utility function, and wishes to optimize its utility minus the amount of payment that it must give. We then show that the principal can learn to steer no-regret agents to achieve the CEP that is optimal for the principal, without prior knowledge of the agents' utility functions:

Theorem 1.2 (Informal version of Theorem 5.2)**.** *Let $F^*(\Gamma)$ be the principal's objective value in the optimal CEP in game Γ. Then, with no prior knowledge of the agents' utilities, there exists an algorithm for the principal to achieve time-averaged value at least $F^*(\Gamma) - \mathsf{poly}(M) \cdot T^{-1/4}$ against any no-regret agents.*

All our algorithms are implementable by the principal in $\mathsf{poly}(M)$ running time. To our knowledge, our Result 1 is the first positive result for utility-learning against arbitrary no-regret agents. Our Result 2 is the first in steering any no-regret agents without prior knowledge of their utilities, and the first exact characterization of the optimal value achievable by the principal in the steering problem.

1.2 Related research

Steering agents to achieve desirable outcomes is an important subject of study in economics, computer science, and control theory (e.g., Monderer and Tennenholtz [2003], Zhang et al. [2024], Canyakmaz et al. [2024], Yao et al. [2025]). In particular, Zhang et al. [2024] introduced the problem of steering no-regret learners via payments. Critically, most prior works on steering assume that the principal knows the agents' utility functions. We study how the agents' utility functions can be learned, the solution to which will serve for any downstream applications including steering.

Besides the aforementioned works, another literature about learning agents' utility functions from their behavior is *learning in Stackelberg games*, where the principal cannot pay the agents but can influence the agents' actions by changing its own strategy, and the agents respond myopically [Letchford et al., 2009, Peng et al., 2019] or by learning algorithms [Haghtalab et al., 2022]. Strong impossibility results are known for this problem: without regularity conditions or without knowing the details of the agents' learning algorithms, the principal cannot learn the agent's utility function [Zhang et al., 2025]. In contrast, we consider a setting where the principal gives payments but does not take actions. We show that the use of payment makes a significant difference: the principal can now learn the utility functions of any no-regret agents without knowing their behavioral details.

Even with payment, the problem of learning from learning agents is still challenging. As we will discuss more in Section 4, a key obstacle is the non-forgetfulness of agents' no-regret algorithms. The payment given to the agents in the past affect their future behavior. Non-forgetfulness is a known issue in multi-agent learning dynamics [Wu et al., 2022, Cai et al., 2024, Scheid et al., 2024]. We overcome this obstacle by designing a zero-sum-game-based learning algorithm for the principal and using signals to influence agents, without requiring the agents' algorithms to be forgetful.

Our use of signals to influence agents is inspired by *information design* (*e.g.*, Kamenica and Gentzkow [2011]). In fact, Feng et al. [2022], Bacchiocchi et al. [2024] studied how to learn agents' utility functions by providing signals to shape agents' beliefs and behaviors in information design problems. While they consider myopically best-responding agents, we allow any no-regret learning agents, a more challenging setting necessitating the combination of signals and payments.

The literature on *playing against no-regret agents* studies how the principal should play a game if they know the agents' utilities and some properties of the agents' algorithms (*e.g.*, Braverman et al. [2018], Deng et al. [2019], Mansour et al. [2022], D'Andrea [2023], Lin and Chen [2025], Arunachaleswaran et al. [2025]). For example, Deng et al. [2019] show that, for *mean-based* no-regret agents, the principal can gain *more* than the Stackelberg value in a Stackelberg game. Our algorithms and setting, on the other hand, consider *worst-case* no-regret agent behaviors. While most of the cited papers consider principal-agent problems with a single agent and with no payment, we consider arbitrary payment-argumented multi-agent games.

2 Preliminaries

Notations. The set of positive integers $\{1, \dots, n\}$ is denoted by $[n]$. The notation $f \lesssim g$ means $f \le \mathcal{O}(g)$, and $f \gtrsim g$ means $f \ge \Omega(g)$. $\tilde{\mathcal{O}}$ and $\tilde{\Omega}$ hide factors logarithmic in their argument. For a vector $\boldsymbol{v} \in \mathbb{R}^m$, its i-th component is denoted by $\boldsymbol{v}[i]$ or v_i. Vector of ones and zeros are $\mathbf{1} = (1, \dots, 1)$ and $\mathbf{0} = (0, \dots, 0)$. $\mathbb{I}\{\cdot\}$ is the indicator function, *i.e.*, for a statement p, $\mathbb{I}\{p\} = 1$ if p is true and 0 if p is false.

Normal-form games. A normal-form game $\Gamma = (n, A, U)$ consists of a set of n *agents*, or *players*, denoted by $[n]$. Each agent $i \in [n]$ has an action set A_i of size $m_i \ge 2$. Let $m := \max_i m_i$ and $M = \prod_i m_i$. A tuple $\boldsymbol{a} \in A := A_1 \times \cdots \times A_n$ is an *action profile*. Each agent has a *utility function* $U_i : A \to [0, 1]$, denoting the utility for agent i when the agents play action profile $\boldsymbol{a} \in A$. A *mixed strategy* of agent i is a distribution $\boldsymbol{x}_i \in \Delta(A_i)$. We will overload the utility function U_i to accept mixed strategies, so that $U_i(\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)$ is the expected utility for agent i when every agent $j \in [n]$ independently samples $a_j \sim \boldsymbol{x}_j$. As is standard in game theory, we use $-i$ to refer to the tuple of all agents except i. For instance, $U_i(a_i', \boldsymbol{a}_{-i})$ is the utility of agent i when agent i plays action $a_i' \in A_i$ and other agents play $\boldsymbol{a}_{-i} \in A_{-i} = \bigtimes_{j \ne i} A_j$.

No-regret learning. In *no-regret learning*, a learner has a convex compact strategy set $\mathcal{X} \subset \mathbb{R}^m$, and interacts with a possibly adversarial environment. On each timestep t, the learner selects a strategy $\boldsymbol{x}^t \in \mathcal{X}$. Simultaneously, the environment, possibly adversarially, selects a linear utility vector $\boldsymbol{u}^t \in \mathbb{R}^m$, which we assume to be bounded: $|\langle \boldsymbol{u}^t, \boldsymbol{x} \rangle| \lesssim 1$. The learner's *regret* after T timesteps is $R(T) = \max_{\boldsymbol{x} \in \mathcal{X}} \sum_{t=1}^T \langle \boldsymbol{u}^t, \boldsymbol{x} - \boldsymbol{x}^t \rangle$. We say that the learner is running a *no-regret algorithm* if its average regret $R(T)/T \to 0$ in the limit $T \to \infty$ for all possible sequences $(\boldsymbol{u}^1, \dots, \boldsymbol{u}^T)$.

A well-known no-regret algorithm for general strategy set $\mathcal{X}$ is *projected gradient descent/ascent* [Zinkevich, 2003]. Projected gradient ascent selects $\boldsymbol{x}^1 \in \mathcal{X}$ arbitrarily, and for every timestep $t > 1$ sets $\boldsymbol{x}^t = \Pi_{\mathcal{X}}[\boldsymbol{x}^{t-1} + \eta \boldsymbol{u}^{t-1}]$ where $\Pi_{\mathcal{X}}$ is the ℓ_2-projection operator into $\mathcal{X}$. With step size $\eta = B/(G\sqrt{T})$, where B is the ℓ_2-diameter of $\mathcal{X}$ and G bounds the ℓ_2-norm of $\boldsymbol{u}^t$, the algorithm achieves a regret of $R(T) \lesssim BG\sqrt{T}$.

To apply no-regret learning to games, each agent i runs a no-regret algorithm over their mixed strategy set $\mathcal{X} = \Delta(m_i)$. For this special case, the most common algorithm is the *multiplicative weights update* (MWU) algorithm (*e.g.*, Freund and Schapire [1999]), which sets $\boldsymbol{x}^t \propto \exp\left(\eta \sum_{\tau<t} \boldsymbol{u}^\tau\right)$, where $\eta = \sqrt{\log(m_i)/T}$ is an appropriately-chosen step size and $\exp(\cdot)$ is the element-wise exponential function. Multiplicative weights achieves regret bound $R(T) \lesssim \sqrt{T \log m_i}$.

3 Our Problem: Learning from Learning Agents

We study a setting where the principal does not initially know anything about the agents' utility functions U_i except for boundedness. The principal knows the number of agents n and their action sets A_i, oversees the agents playing the game repeatedly over T rounds, and can provide *payments* and *signals* to influence the agents' behavior, in order to learn the utility functions of the agents. More formally, in each round $t = 1, \dots T$, the following events happen in order:

1. The principal selects *payment function* $P_i^t : A_i \to [0, B]$ for each agent i, where B is a large constant.[1] The payment $P_i^t(a_i)$ is added to agent i's reward, creating a new game Γ^t with utility functions given by $U_i^t(a) := U_i(a) + P_i^t(a_i)$.
2. The principal sends a *signal* $s_i^t \in S_i$ to each agent i.
3. Observing s_i^t (but not P_i^t), each agent i selects a mixed strategy $\boldsymbol{x}_i^t \in \Delta(A_i)$.
4. The principal observes the joint mixed strategy $\boldsymbol{x}^t$. Each agent i gets reward $U_i^t(\boldsymbol{x}^t)$.

Agents' no-regret learning We assume that each agent selects $\boldsymbol{x}_i^t$ using a no-regret learning algorithm, or more precisely, a *contextual* no-regret learning algorithm where signals are contexts and agents achieve no regret given any signal/context. Formally, there is a (possibly game-dependent) constant $C \le \mathsf{poly}(M)$ such that, for every agent i, every signal $s_i \in S_i$, every time step $t \le T$,[2]

$$\hat{R}_i(t, s_i) \ := \ \max_{a_i \in A_i} \sum_{\tau \le t : s_i^\tau = s_i} \left[U_i^\tau(a_i, \boldsymbol{x}_{-i}^\tau) - U_i^\tau(\boldsymbol{x}^\tau) \right] \ \le \ C\sqrt{T}. \tag{1}$$

One way to achieve this guarantee is for each agent to run $|S_i|$ independent no-regret algorithms, one for each signal. However, we do not restrict to any specific algorithms: agents can run *any* algorithms satisfying the above no-regret property, such as projected gradient ascent, MWU, Exp3, and so on. In particular, agents can run full-feedback or bandit-feedback no-regret algorithms; our results are unaffected. Indeed, the agents' learning algorithms need not even be independent; they could choose their actions using a centralized algorithm. As typical no-regret algorithms do not require knowledge of the utility functions (they operate on the feedback received after each round), the agents can also be initially ignorant of their own utility functions U_i, just as the principal is.

Mixed strategies and agent randomization Our model stipulates that the principal observes the full joint mixed strategy $\boldsymbol{x}^t$, instead of a sampled pure strategy profile $\boldsymbol{a}^t \sim \boldsymbol{x}^t$. This stipulation, however, is not at all vital and could be removed with only minimal effect on the results. In particular, suppose that each agent i, instead of announcing a mixed strategy $\boldsymbol{x}_i^t$ in each round, samples an action $a_i^t \sim \boldsymbol{x}_i^t$ and announces the "mixed strategy" that assigns all mass to a_i^t. Then the principal only observes a_i^t. The difference now is that, since the agents are randomizing, the regret bound (1) cannot hold *deterministically*; there will be some stochastic approximation error term that can be bounded by Azuma-Hoeffding inequality, so (1) only holds *with high probability*. In that setting, one can think of the results of this paper as conditional on the high-probability event that (1) holds.

In particular, lower bounds that apply to the setting where the principal only observes the realized action profile also apply to the setting where the principal observes the mixed strategy profile. We will use this fact freely to prove lower bounds, in Theorem 4.4.

3.1 Our Goal

Our goal is to design algorithms for the principal to learn the agents' utility functions U_i by designing the payment functions P_i^t and sending signals s_i^t, for any no-regret learning agents. However, this goal as currently stated is impossible, because agents' regrets are only affected by the *utility differences* between

[1] All our positive results will only require $B = 2$, since the utility functions are bounded by 1.

[2] Such *anytime* no-regret is not a strong requirement. Our full paper shows that any algorithm with the usual no-regret guarantee under adversarial environments automatically satisfies anytime no-regret.

actions, namely, $U_i(a_i, \boldsymbol{a}_{-i}) - U_i(a_i', \boldsymbol{a}_{-i})$, and the behaviors of typical no-regret algorithms (such as MWU) only depend on those differences. In other words, if we create another game $\tilde{\Gamma}$ with $\tilde{U}(a_i, \boldsymbol{a}_{-i}) = U_i(a_i, \boldsymbol{a}_{-i}) + W_i(\boldsymbol{a}_{-i})$ for all $\boldsymbol{a} \in A$, where $W_i : A_{-i} \to \mathbb{R}$ is an arbitrary function not depending on agent i's action, there is no way to distinguish Γ from $\tilde{\Gamma}$ using only agents' behavioral data − that is, games Γ and $\tilde{\Gamma}$ are *strategically equivalent*. Thus, we can only determine utility functions *up to* strategic equivalence.

Our formal goal is thus the following. Given a game Γ and precision ε, we say that the principal's algorithm *ε-learns* the game Γ if it outputs utility functions $\tilde{U}_i : A \to \mathbb{R}$ such that there exist functions $W_i : A_{-i} \to \mathbb{R}$ satisfying

$$\left| U_i(\boldsymbol{a}) + W_i(\boldsymbol{a}_{-i}) - \tilde{U}_i(\boldsymbol{a}) \right| \le \varepsilon, \quad \forall i \in [n], \quad \forall \boldsymbol{a} \in A. \tag{2}$$

The goal of the principal is to ε-learn Γ in as few rounds as possible. Learning a game up to strategic equivalence is sufficient for almost all practical purposes because equilibrium notions such as Nash equilibria and correlated equilibria are invariant under strategically equivalent transformations of utility functions.

While our paper focuses on the problem of minimizing the number of *rounds* it takes to learn the game, an alternative goal might be to minimize the total payment to do so. The full version of our paper shows that, however, the minimal achievable payment is upper-bounded and lower-bounded by the minimal number of rounds up to constant factors, so these two problems are quantitatively similar.

4 Learning a Game by Paying No-Regret Learners

We design efficient algorithms for the principal to ε-learn a game by paying no-regret learning agents. We will start with the single-agent case to illustrate the main ideas of our algorithms, before proceeding to the multi-agent case. The single-agent algorithm turns out to be surprisingly simple.

4.1 A Simple Zero-Sum-Game Based Algorithm in the Single-Agent Case

In the single-case case ($n = 1$), it is more convenient to view the single agent's utility function as a vector $\boldsymbol{u} \in [0, 1]^m$, and similarly the payment $P^t : [m] \to \mathbb{R}$ as vector $\boldsymbol{p}^t \in \mathbb{R}^m$ and total utility $\boldsymbol{u}^t := \boldsymbol{u} + \boldsymbol{p}^t$. To simplify notations, we subtract the average utility of all actions from the utility of each action: $\boldsymbol{u} \leftarrow \boldsymbol{u} - \frac{\langle \mathbf{1}, \boldsymbol{u} \rangle}{m} \mathbf{1}$, so that $\boldsymbol{u} \in [-1, 1]^m$ and $\langle \mathbf{1}, \boldsymbol{u} \rangle = 0$. By the discussion before (2), this does not change the principal's learning problem. Our algorithm in the single-agent case will not need signaling, so it will be enough to set $|S_i| = 1$.

The main challenge of learning a game from a no-regret agent is the history-dependency of the agent's behavior. To fix ideas, suppose the agents has two actions A and B with unknown utility gap $u_A - u_B = \Delta > 0$. A straightforward attempt to learn the gap Δ is to try different payments for action B (while paying 0 for action A) in a binary-search manner: the payment at which the agent just starts to play action B should reveal the value of Δ. However, that approach does not work for a no-regret learning agent because no-regret algorithms may not respond instantaneously to changes to the payment function. Historical payments affect the agent's future behavior. Moreover, the agent may incur *negative* regret over time, making it difficult to learn anything from the agent's behavior on subsequent rounds. For example, if an agent has regret $-K$ for all actions, then one cannot say anything at all about how the agent will behave in the next K rounds.

The key idea of our algorithm is to *imagine the principal and the agent as playing a zero-sum game* where the principal selects the payment function $\boldsymbol{p}$ from some set $\mathcal{P}$ to be specified later, the agent selects mixed strategy $\boldsymbol{x} \in \Delta(m)$, the agent's utility is given by $\langle \boldsymbol{u} + \boldsymbol{p}, \boldsymbol{x} \rangle$, and the principal's utility is $-\langle \boldsymbol{u} + \boldsymbol{p}, \boldsymbol{x} \rangle$. Call this game Γ_0. In particular, by setting $\mathcal{P} = \{\boldsymbol{p} \in [0, 2]^m : \langle \mathbf{1}, \boldsymbol{p} \rangle = m\}$, the zero-sum game Γ_0 has the following property:

Lemma 4.1. *In the zero-sum game Γ_0, every ε-Nash equilibrium strategy for the principal has the form $\boldsymbol{p} = \mathbf{1} - \boldsymbol{u} + \boldsymbol{z}$, where $\|\boldsymbol{z}\|_1 \le 4m\varepsilon$.*

Proof. Setting $\boldsymbol{p} = \mathbf{1} - \boldsymbol{u}$ guarantees $\langle \boldsymbol{u} + \boldsymbol{p}, \boldsymbol{x} \rangle = \langle \mathbf{1}, \boldsymbol{x} \rangle = 1$ for every $\boldsymbol{x} \in \Delta(m)$. Thus, in every ε-Nash equilibrium, the agent's utility is at most $1 + \varepsilon$. Now suppose for contradiction that $(\boldsymbol{p}, \boldsymbol{x})$ is an ε-Nash equilibrium with $\|\boldsymbol{p} + \boldsymbol{u} - \mathbf{1}\|_1 > 4m\varepsilon$. Then since $\langle \boldsymbol{p} + \boldsymbol{u} - \mathbf{1}, \mathbf{1} \rangle = 0$ by construction, there must be an action a for which $(\boldsymbol{p} + \boldsymbol{u} - \mathbf{1})[a] > 2\varepsilon$, *i.e.*, $(\boldsymbol{u} + \boldsymbol{p})[a] > 1 + 2\varepsilon$. But then the agent has an ε-profitable deviation to action a. □

It is well known that no-regret learning algorithms converge on average to Nash equilibria in zero-sum games. In particular, if both principal and agent run no-regret algorithms, and R_0 is the principal's regret after T timesteps, then the average principal strategy $\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{p}^t$ is an ε-Nash equilibrium for $\varepsilon \lesssim (R_0+C\sqrt{T})/T$. Here, we use the projected gradient descent algorithm for the principal. Note that, although the principal's utility function $\boldsymbol{p} \mapsto -\langle \boldsymbol{u}+\boldsymbol{p}, \boldsymbol{x}\rangle$ depends on $\boldsymbol{u}$ (which the principal does not know), the gradient of the principal's utility function is $-\boldsymbol{x}$, which does not depend on $\boldsymbol{u}$. Thus, the principal can run projected gradient descent without knowing $\boldsymbol{u}$. The resulting algorithm is formalized in Algorithm 1.

Algorithm 1 Principal's learning algorithm for a single no-regret agent

```
1: p^1 ← 1
2: for each time t = 1, ..., T do
3:     principal selects payment vector p^t ∈ P, observes strategy x^t played by the agent
4:     principal sets p^{t+1} ← Π_P[p^t − ηx^t]     ▷ η = √(m/T) is the step size
5: return −(1/T) Σ_{t=1}^T p^t
```

Theorem 4.2 (main result). *Algorithm 1 ε-learns any single-agent game Γ in $T = \mathcal{O}(\frac{m^3+C^2m^2}{\varepsilon^2})$ rounds, for any agent with regret $\leq C\sqrt{T}$.*

Proof. Let $\bar{\boldsymbol{p}} = \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{p}^t$ be the average payment. From the preliminaries, the regret bound of the principal is given by $R_0 \leq BG\sqrt{T}$ where $B \lesssim \sqrt{m}$ and $G = 1$. By the previous paragraph, the average play between the principal and the agent is an $\varepsilon \lesssim (R_0 + C\sqrt{T})/T$-Nash equilibrium. Then, by Lemma 4.1, we have

$$\varepsilon = \|\bar{\boldsymbol{p}} + \boldsymbol{u} - \mathbf{1}\|_\infty \leq \|\bar{\boldsymbol{p}} + \boldsymbol{u} - \mathbf{1}\|_1 \lesssim 4m\frac{R_0 + C\sqrt{T}}{T} \lesssim \frac{m}{\sqrt{T}}(C + \sqrt{m}).$$

Solving for T yields the desired result. □

The zero-sum-game idea of Algorithm 1 is surprisingly simple and powerful. The principal moves the payment vector in the opposite direction of the agent's mixed strategy every round, and the negative average payment vector ultimately becomes an accurate estimate of the agent's utility vector. This idea works for any no-regret learning algorithm of the agent.

Another possible method to overcome history-dependency is to use signals: whenever the payment vector is changed, send a new signal to the agent to disentangle from the history. Signaling allows us to implement a binary-search algorithm to learn the game. However, that would require $m\log(1/\varepsilon)$ signals, one for each step of the binary search, and the total number of rounds would be at least $C^2m/\varepsilon^2 \cdot \log(1/\varepsilon)$, whereas our Algorithm 1 achieves the better dependence of $1/\varepsilon^2$, saving a logarithmic term, without using signals.

4.2 Extension to the Multi-Agent Case

We then consider the multi-agent case. Our algorithm for the multi-agent case will combine the single-agent algorithm with signaling. Intuitively, our algorithm uses signals to induce the action profile $\boldsymbol{a}_{-i}$ among other agents without increasing their regret by too much. More precisely, we set the signal set as $S_i := A_i \cup \{\perp\}$ where $\perp$ is a special signal indicating that i's utility is the one being learned at the moment. For every action profile $\boldsymbol{a}_{-i} \in A_{-i}$, we send signal $\perp$ to agent i and the desired action a_j for each agent $j \neq i$ to learn $U_i(\cdot, \boldsymbol{a}_{-i})$. This idea is formalized in Algorithm 2.

Theorem 4.3. *For some choice of parameter L, Algorithm 2 ε-learns any game in $\frac{\mathsf{poly}(M,C)}{\varepsilon^2}$ rounds.*

Proof Sketch. Since the principal always gives a large reward to agent who obey signals other than $\perp$, agent will almost always obey such signals. Thus, agents other than agent i will almost always play profile $\boldsymbol{a}_{-i}$. This allows the principal to learn $U_i(\cdot, \bar{\boldsymbol{a}}_{-i})$ using the one-player algorithm from Theorem 4.2. The formal proof is deferred to Appendix A. □

Algorithm 2 Principal's learning algorithm for multiple no-regret agents

$t \leftarrow 1$
for every agent $i = 1, \dots, n$ **do**
  for every action profile $\bar{\boldsymbol{a}}_{-i} \in A_{-i}$ **do**
    $\boldsymbol{p}^1 \leftarrow \mathbf{1} \in \mathbb{R}^{A_i}$
    for timestep $\ell = 1, \dots, L$ **do**
      principal sets $P_i^t(\cdot) = \boldsymbol{p}^\ell[\cdot]$ and $P_j^t(a_j) = 2\mathbb{I}\{a_j = \bar{a}_j\}$ for every $j \neq i$
      principal sends signals $s_i^t = \perp$ and $s_j^t = \bar{a}_j$ for every $j \neq i$
      principal observes profile $\boldsymbol{x}^t$ played by agents
      principal sets $\boldsymbol{p}^{\ell+1} \leftarrow \Pi_{\mathcal{P}}\left[\boldsymbol{p}^\ell - \eta \boldsymbol{x}_i^t\right]$ ▷ $\eta = \sqrt{m_i/L}$ is the step size
      $t \leftarrow t + 1$
    $\tilde{U}_i(\cdot, \bar{\boldsymbol{a}}_{-i}) \leftarrow -\frac{1}{L}\sum_{\ell=1}^{L} \boldsymbol{p}^\ell$
return $\tilde{U}$

Signals are vital to this analysis. Without them, it would be possible for players to incur large *negative* regret, which harms the learning process because it allows the players to "delay" the learning until their regrets once again become non-negative. For example, if we were to execute our algorithm without signals, then by the time $\underline{T}_n(0)$ at which the outer loop reaches agent n, agent n could have $\Omega(\overline{T}_n(0))$ regret for *every* action, making it impossible to say anything about how agent n will act for the next $\Omega(\overline{T}_n(0))$ rounds. Using signals allows us to separate out the regret of agent n in previous rounds from the regret of agent n when its own utility function is being learned.

4.3 Lower bound

We then prove a lower bound that matches Theorem 4.3 up to the exponent on M.

Theorem 4.4. *Against no-regret agents, any algorithm that ε-learns a game takes at least* $\max\{\Omega(nM) \cdot \log \frac{1}{\varepsilon},\ \frac{C^2}{4\varepsilon^2}\}$ *rounds.*

Proof. We prove the two terms in the max separately. For the first term, suppose that $U_i(1, \cdot) = 0$ for all agents i, and $U_i(a_i, a_{-i}) \sim \{0, 2\varepsilon, 4\varepsilon, \dots, 1\}$ i.i.d. for $2 \leq a \leq m_i$ and $a_{-i} \in A_{-i}$. Thus the utility U is uniformly sampled from a set of $\Omega(1/\varepsilon)^K$ possible utilities, where

$$K = \sum\nolimits_{i=1}^{n} \left((m_i - 1)\prod\nolimits_{j \neq i} m_j\right) \geq \frac{nM}{2}.$$

Each utility function differs by 2ε, it follows that ε-learning a game sampled from this family entails exactly outputting the utility U. Suppose that, as discussed in Section 3, the no-regret algorithms always output pure strategies $a_i^t \in A_i$. Then on each round, the principal only observes a single action profile $a \in A$, which only conveys $\log M$ bits of information. Therefore, ε-learning the game takes at least

$$\frac{K \log(\Omega(1/\varepsilon))}{\log M} \gtrsim \frac{nM \log(1/\varepsilon)}{\log M}$$

rounds, as desired.

For the second term, suppose $\varepsilon \leq \frac{C}{2\sqrt{T}}$. Let $Z_i : A \to [0, \varepsilon]$ be any function, and suppose that every agent plays according to utility function $U_i + Z_i$ instead of U_i using an algorithm with $\frac{C}{2}\sqrt{T}$ regret. Such an agent incurs at most $\frac{C}{2}\sqrt{T} + \varepsilon T \leq C\sqrt{T}$ regret with respect to U_i. Such an agent is completely indistinguishable from an agent who has true utility $U_i + Z_i$ and runs an algorithm with regret $C\sqrt{T}$, and therefore the principal can never distinguish between these two possibilities. Since this is true for any Z_i, this means that the principal cannot learn U_i to accuracy better than $\varepsilon \leq \frac{C}{2\sqrt{T}}$. Thus, T must be at least $\frac{C^2}{4\varepsilon^2}$ for the principal to ε-learn the game. □

Theorem 4.4 shows that it is impossible to exponentially improve the dependence on any of the parameters in Theorem 4.2. For example, it implies that there can be no algorithm taking $C^2/\varepsilon^2 \cdot M^{1-\Omega(1)}$ rounds, because that would contradict the lower bound for constant ε and suffficiently large M. We leave it as an interesting open question to close the polynomial gaps between the lower and upper bounds presented here.

5 Applicaiton: Learning to Steer No-Regret Learners

A main motivating application of our result is the problem of *steering* no-regret learners to desired outcomes, introduced by Zhang et al. [2024] who assume that the principal knows the game. In this section, we explore the steering problem with unknown agent utilities.

The steering problem. In the steering problem, the principal uses payments (and signals) to incentivize no-regret learning agents to achieve certain outcomes. We focus on the outcome that is optimal for the principal. Formally, assume that the principal has a utility function $U_0 : A \to \mathbb{R}$. We aim to maximize the principal's utility minus payments over T rounds:

$$F(T) := \tfrac{1}{T} \sum_{t=1}^{T} \mathbb{E}\left[U_0(\boldsymbol{a}^t) - \sum_{i=1}^{n} P_i^t(\boldsymbol{a}^t)\right].$$

To analyze the above objective, we introduce a solution concept called *correlated equilibrium with payments* (CEP). A CEP is a correlated distribution of signals and payment functions that satisfies the usual incentive compatibility constraints. Formally, we have the following definition.

Definition 5.1. A *correlated profile with payments* is a pair $(\mu, P) \in \Delta(A) \times [0, B]^{[n]\times A\times A}$, where

- $\mu \in \Delta(A)$ is a distribution over joint action profiles $\boldsymbol{a} \in A$, recommending each a_i to agent i privately;
- P consists of n payment functions $P_i : A \times A \to [0, B]$, where $P_i(\boldsymbol{s}, \boldsymbol{a})$ is the payment to agent i given joint signal $\boldsymbol{s} \in A$ and joint action $\boldsymbol{a} \in A$.

Given (μ, P), the *objective value* for the principal is defined as

$$F(\mu, P) := \mathop{\mathbb{E}}_{\boldsymbol{a}\sim\mu}\left[U_0(\boldsymbol{a}) - \sum_{i=1}^{n} P_i(\boldsymbol{a}, \boldsymbol{a})\right],$$

An ε-*correlated equilibrium with payments* (ε-CEP) is a pair (μ, P) satisfying the ε-incentive compatibility (IC) constraints: for every agent $i \in [n]$ and deviation function $\phi_i : A_i \to A_i$,

$$\mathop{\mathbb{E}}_{\boldsymbol{a}\sim\mu}\left[U_i^P(\boldsymbol{a}, \phi_i(a_i), \boldsymbol{a}_{-i}) - U_i^P(\boldsymbol{a}, \boldsymbol{a})\right] \le \varepsilon,$$

where $U_i^P(\boldsymbol{s}, \boldsymbol{a}) := U_i(\boldsymbol{a}) + P_i(\boldsymbol{s}, \boldsymbol{a})$. An 0-CEP is called a CEP.

Let $F^*(\Gamma)$ be the principal's objective value under an *optimal* CEP of game Γ:

$$F^*(\Gamma) \;=\; \max_{(\mu,P)\text{: a CEP of game }\Gamma} F(\mu, P).$$

Steering to optimal CEP We now show that the principal can learn to steer agents to achieve the optimal CEP objective $F^*(\Gamma)$ in the limit $T \to \infty$. The algorithm (Algorithm 3) works in two stages. In the first stage, the principal uses Algorithm 2 to learn the utility functions of the agents. Then, the principal computes an optimal CEP and steers the agents to it. Notably, since the principal learns the game up to an error $\varepsilon > 0$, it must give extra payments of at least 2ε to ensure that agents do not deviate. Theorem 5.2 shows that the principal can learn to steer agents to achieve the optimal objective $F^*(\Gamma)$ at a rate of $\mathsf{poly}(M, C)/T^{1/4}$. In addition, our full paper also shows that no steering algorithm can achieve a better objective than $F^*(\Gamma)$.

Theorem 5.2. *For appropriate choices of parameters L (from Algorithm 2) and ρ, Algorithm 3 guarantees principal objective $F(T) \ge F^*(\Gamma) - \mathsf{poly}(M, C)/T^{1/4}$ on average in T rounds.*

Algorithm 3 Principal's algorithm for steering without prior knowledge of utilities

1: using Algorithm 2, estimate the utility functions to precision ε
2: compute an optimal CEP $(\tilde{\mu}^*, \tilde{P}^*)$ of the estimated game $\tilde{\Gamma}$
3: **for** remaining rounds **do**
4: set $\mu^t = \mu^*$ and $P_i^t(\boldsymbol{s}, \boldsymbol{a}) = \begin{cases} \tilde{P}_i^*(\boldsymbol{a}, \boldsymbol{a}) + 2\varepsilon + \rho & \text{if } \quad \boldsymbol{s} = \boldsymbol{a} \\ 2 & \text{if } \quad s_i = a_i, \boldsymbol{s}_{-i} \neq \boldsymbol{a}_{-i} \\ 0 & \text{otherwise} \end{cases}$.

Proof. From the analysis of Theorem 4.3, Algorithm 2 learns a game to precision $\varepsilon = \mathsf{poly}(M, C)/\sqrt{L}$. (Note that we do not express ε as a function of T because T is now the total number of rounds across both learning and steering stages.)

Since $\tilde{U}$ and U differ by only ε (up to agent-independent terms), every CEP of $\tilde{\Gamma}$ is a 2ε-CEP of Γ. The payment function P_i^t for the steering stage then ensures that, when given signal s_i, it is *dominant* for agent i to play $a_i = s_i$. Formally, regardless of how other agents act, we have

$$U_i^t(\boldsymbol{s}, a_i, \boldsymbol{a}_{-i}) - U_i^t(\boldsymbol{s}, a_i', \boldsymbol{a}_{-i}) \geq \rho, \quad \forall \boldsymbol{s} \in A, a_i = s_i, \forall a_i' \neq s_i, \forall \boldsymbol{a}_{-i} \in A_{-i}.$$

Further, from the analysis of Theorem 4.3, agent i's regret against following signals $s_i \neq \perp$ is always nonnegative. Therefore, by agent i's regret bound, there are at most $C\sqrt{T}/\rho$ rounds on which agent i fails to obey recommendation s_i in the steering stage. By a union bound, there are therefore $mnC\sqrt{T}/\rho$ rounds in the steering stage on which $\boldsymbol{a}^t \neq \boldsymbol{s}^t$. Thus, the principal's suboptimality is bounded by

$$F^*(\Gamma) - F(T) \leq \underbrace{F^*(\Gamma) - F^*(\tilde{\Gamma})}_{(1)} + \underbrace{\tfrac{(2n+1)nML}{T}}_{(2)} + \underbrace{n(2\varepsilon + \rho)}_{(3)} + \underbrace{\tfrac{(2n+1)mnC}{\rho\sqrt{T}}}_{(4)} \leq \mathsf{poly}(M) \cdot \left(\tfrac{L}{T} + \tfrac{1}{\sqrt{L}} + \rho + \tfrac{1}{\rho\sqrt{T}}\right)$$

where the four terms are:

(1) The difference between the optimal objectives on games Γ and $\tilde{\Gamma}$. It is at most $2n\varepsilon$ because $F^*(\Gamma) = F(\mu^*, P^*) \leq F(\mu^*, P^* + 2\varepsilon) + 2n\varepsilon \leq F(\tilde{\mu}^*, \tilde{P}^*) + 2n\varepsilon = F^*(\tilde{\Gamma}) + 2n\varepsilon$.

(2) The suboptimality and payments in the utility learning stage,

(3) The bonus payments to ensure strict incentive compatibility in the steering stage, and

(4) The suboptimality and payments in rounds on which $a^t \neq s^t$.

Setting $\rho = T^{-1/4}$ and $L = T^{2/3}$ then completes the proof. $\square$

6 Conclusions

We showed that a principal can efficiently learn the utility functions of agents in games through payments and signals, and applied our algorithms to achieve optimal steering without prior knowledge of the game. Our results apply to arbitrary no-regret agents. The zero-sum-game based algorithm for the single-agent case is surprisingly simple.

A Proof of Theorem 4.3

As in Theorem 4.2, we will assume without loss of generality that $\sum_{a_i \in A_i} U_i(a_i, \boldsymbol{a}_{-i}) = 0$ for all agents i and opponent profiles $\boldsymbol{a}_{-i}$.

We first claim that, for any agent i and any given signal $s_i^t = a_i \in A_i$, the total probability mass that i plays actions *other than* a_i is bounded by $C\sqrt{T}$. To see this, note that whenever the principal sends signal a_i, the payment is always set such that $U_i^t(a_i, \boldsymbol{a}_{-i}) \geq 1 + U_i^t(a_i', \boldsymbol{a}_{-i})$. Thus, the number of times i does not play $a_i^t = a_i$ quantity lower-bounds the regret $\hat{R}(T, a_i)$. The claim follows from the regret guarantee $\hat{R}(T, a_i) \le C\sqrt{T}$.

We will refer to the iterations of the inner loop over action profiles $\boldsymbol{a}_{-i}$ as *phases*. Fix an agent i, and number the phases for that agent using integers $k \in \{1, \cdots, M_i = \prod_{j \neq i} m_j\}$, corresponding to strategy profiles $\bar{\boldsymbol{a}}_{-i}^1, \ldots, \bar{\boldsymbol{a}}_{-i}^{M_i} \in A_{-i}$. Let $\mathcal{T}_i(k) = \{\underline{T}_i(k), \ldots, \overline{T}_i(k)\}$ be the set of timesteps in agent i's kth phase. The length of each phase is $|\mathcal{T}_i(k)| = L$. Let B_K be the total probability mass placed by all agents $j \neq i$ on strategy profiles other than $\bar{\boldsymbol{a}}_{-i}^k$ throughout phases $1, \ldots, K$. By the previous claim, we have

$$B_K := \sum_{k \le K, t \in \mathcal{T}_i(k)} \Big(1 - \prod_{j \neq i} \boldsymbol{x}_j^t(\bar{a}_j^k)\Big) \le \sum_{k \le K, t \in \mathcal{T}_i(k), j \neq i} \big(1 - \boldsymbol{x}_j^t(\bar{a}_j^k)\big) \le nmC\sqrt{T}.$$

By the principal's regret bound in each phase, we must have

$$\begin{aligned}
\sum_{t \in \mathcal{T}_i(k)} U_i^t(\boldsymbol{x}_i^t, \bar{\boldsymbol{a}}_{-i}^k) &= \sum_{t \in \mathcal{T}_i(k)} U_i(\boldsymbol{x}_i^t, \bar{\boldsymbol{a}}_{-i}^k) + \sum_{t \in \mathcal{T}_i(k)} P_i^t(\boldsymbol{x}_i^t) \\
&\le \sum_{t \in \mathcal{T}_i(k)} U_i(\boldsymbol{x}_i^t, \bar{\boldsymbol{a}}_{-i}^k) + \sum_{t \in \mathcal{T}_i(k)} \big[1 - U_i(\boldsymbol{x}_i^t, \bar{\boldsymbol{a}}_{-i}^k)\big] + R_0 \\
&= L + R_0
\end{aligned}$$

where the inequality follows from the facts that 1) the principal's regret is bounded, 2) $P_i^t(\cdot) = 1 - U_i(\cdot, \bar{\boldsymbol{a}}_{-i}^k)$ is a valid unilateral deviation for the principal.

Fix some $K \le M_i$ and $a_i \in A_i$. By the anytime regret bound of agent i under signal $\perp$, we have

$$\begin{aligned}
\sum_{k \le K, t \in \mathcal{T}_i(k)} U_i^t(a_i, \boldsymbol{x}_{-i}^t) &\le \sum_{k \le K, t \in \mathcal{T}_i(k)} U_i^t(\boldsymbol{x}_i^t, \boldsymbol{x}_{-i}^t) + \hat{R}_i(T_i(k), \perp) \\
&\le 2B_K + \sum_{k \le K, t \in \mathcal{T}_i(k)} U_i^t(\boldsymbol{x}_i^t, \bar{\boldsymbol{a}}_{-i}^k) + \hat{R}_i(T_i(k), \perp) \\
&\le 2nmC\sqrt{T} + K(L + R_0) + nC\sqrt{T}.
\end{aligned}$$

Moving KL to the left and writing $U_i^t(a_i, \boldsymbol{x}_{-i}^t)$ as $U_i(a_i, \boldsymbol{x}_{-i}^t) + P_i^t(a_i)$,

$$\sum_{k=1}^{K} \underbrace{\frac{1}{L} \sum_{t \in \mathcal{T}_i(k)} [U_i(a_i, \boldsymbol{x}_{-i}^t) + P_i^t(a_i) - 1]}_{\text{denoted by } \varepsilon_i(k, a_i)} \le \frac{1}{L}\Big(R_0 K + 3nmC\sqrt{T}\Big).$$

The error we need to bound is $\|\varepsilon_i(k, \cdot)\|_\infty$. Since the above inequality holds for any a_i, and $\sum_{a_i} \varepsilon_i(k, \cdot) = 0$ by definition, it follows that

$$\left\| \sum_{k=1}^{K} \varepsilon_i(k, \cdot) \right\|_\infty = \max_{a_i \in A_i} \left| \sum_{k=1}^{K} \frac{1}{L} \sum_{t \in \mathcal{T}_i(k)} [U_i(a_i, \boldsymbol{x}_{-i}^t) + P_i^t(a_i) - 1] \right| \le \frac{m}{L}\Big(R_0 K + 3nmC\sqrt{T}\Big).$$

By triangle inequality, we have

$$\|\varepsilon_i(k, \cdot)\|_\infty = \left\| \sum_{k'=1}^{k} \varepsilon_i(k', \cdot) - \sum_{k'=1}^{k-1} \varepsilon_i(k', \cdot) \right\|_\infty \le \frac{2m}{L}\Big(R_0 M + 3nmC\sqrt{T}\Big).$$

Finally, substituting $R_0 \lesssim \sqrt{mL}$ and $T \le nML$, we arrive at

$$\|\varepsilon_i(k, \cdot)\|_\infty \lesssim \frac{1}{\sqrt{L}}\Big(m^{3/2} M + n^{3/2} m^2 C M^{1/2}\Big).$$

Taking $L = \mathcal{O}(\frac{m^3 M^2 + n^3 m^4 M C^2}{\varepsilon^2})$ completes the proof, with $T \le nML = \mathcal{O}(\frac{nm^3 M^3 + n^4 m^4 M^2 C^2}{\varepsilon^2})$.

References

Eshwar Ram Arunachaleswaran, Natalie Collina, and Jon Schneider. Learning to Play Against Unknown Opponents. In *Proceedings of the 26th ACM Conference on Economics and Computation*, pages 478–504, Stanford University Stanford CA USA, July 2025. ACM.

Robert Aumann. Subjectivity and correlation in randomized strategies. *Journal of Mathematical Economics*, 1(1):67–96, 1974.

Francesco Bacchiocchi, Matteo Bollini, Matteo Castiglioni, Alberto Marchesi, and Nicola Gatti. Online bayesian persuasion without a clue. In *Neural Information Processing Systems (NeurIPS)*, 2024.

Eyal Beigman and Rakesh Vohra. Learning from revealed preference. In *Proceedings of the 7th ACM conference on Electronic commerce*, pages 36–42, Ann Arbor Michigan USA, June 2006. ACM.

Mark Braverman, Jieming Mao, Jon Schneider, and Matt Weinberg. Selling to a No-Regret Buyer. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pages 523–538, Ithaca NY USA, June 2018. ACM. ISBN 978-1-4503-5829-3.

Yang Cai, Gabriele Farina, Julien Grand-Clément, Christian Kroer, Chung-Wei Lee, Haipeng Luo, and Weiqiang Zheng. Fast last-iterate convergence of learning in games requires forgetful algorithms. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

Ilayda Canyakmaz, Iosif Sakos, Wayne Lin, Antonios Varvitsiotis, and Georgios Piliouras. Learning and steering game dynamics towards desirable outcomes. *arXiv preprint arXiv:2404.01066*, 2024.

Xi Chen, Xiaotie Deng, and Shang-Hua Teng. Settling the complexity of computing two-player Nash equilibria. *Journal of the ACM*, 56(3):14, 2009.

Maurizio D'Andrea. Playing against no-regret players. *Operations Research Letters*, 51(2):142–145, March 2023.

Constantinos Daskalakis, Paul Goldberg, and Christos Papadimitriou. The complexity of computing a Nash equilibrium. In *Symposium on Theory of Computing (STOC)*, 2006.

Yuan Deng, Jon Schneider, and Balasubramanian Sivan. Strategizing against no-regret learners. In *Neural Information Processing Systems (NeurIPS)*, 2019.

Yiding Feng, Wei Tang, and Haifeng Xu. Online Bayesian Recommendation with No Regret. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, pages 818–819, Boulder CO USA, July 2022. ACM.

Yoav Freund and Robert Schapire. Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, 29:79–103, 1999.

Nika Haghtalab, Thodoris Lykouris, Sloan Nietert, and Alexander Wei. Learning in stackelberg games with non-myopic agents. In *ACM Conference on Economics and Computation (EC)*, pages 917–918, 2022.

Emir Kamenica and Matthew Gentzkow. Bayesian Persuasion. *American Economic Review*, 101(6):2590–2615, October 2011.

Volodymyr Kuleshov and Okke Schrijvers. Inverse game theory: Learning utilities in succinct games. In *International Workshop On Internet And Network Economics (WINE)*, pages 413–427. Springer, 2015.

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.

Tao Lin and Yiling Chen. Generalized Principal-Agent Problem with a Learning Agent. In *The Thirteenth International Conference on Learning Representations*, 2025.

Yishay Mansour, Mehryar Mohri, Jon Schneider, and Balasubramanian Sivan. Strategizing against learners in bayesian games. In *Conference on Learning Theory (COLT)*, 2022.

Dov Monderer and Moshe Tennenholtz. k-Implementation. In *ACM Conference on Electronic Commerce (ACM-EC)*, pages 19–28, San Diego, CA, 2003.

Binghui Peng, Weiran Shen, Pingzhong Tang, and Song Zuo. Learning Optimal Strategies to Commit To. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):2149–2156, July 2019.

Antoine Scheid, Aymeric Capitaine, Etienne Boursier, Eric Moulines, Michael Jordan, and Alain Durmus. Learning to Mitigate Externalities: the Coase Theorem with Hindsight Rationality. In *Advances in Neural Information Processing Systems*, volume 37, pages 15149–15183. Curran Associates, Inc., 2024.

Jibang Wu, Haifeng Xu, and Fan Yao. Multi-Agent Learning for Iterative Dominance Elimination: Formal Barriers and New Algorithms. In *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, pages 543–543. PMLR, July 2022.

Fan Yao, Yuwei Cheng, Ermin Wei, and Haifeng Xu. Single-agent Poisoning Attacks Suffice to Ruin Multi-Agent Learning. In *The Thirteenth International Conference on Learning Representations*, 2025.

Morteza Zadimoghaddam and Aaron Roth. Efficiently Learning from Revealed Preference. In *Internet and Network Economics*, volume 7695, pages 114–127. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. Series Title: Lecture Notes in Computer Science.

Brian Hu Zhang, Gabriele Farina, Ioannis Anagnostides, Federico Cacciamani, Stephen Marcus McAleer, Andreas Alexander Haupt, Andrea Celli, Nicola Gatti, Vincent Conitzer, and Tuomas Sandholm. Steering no-regret learners to a desired equilibrium. In *ACM Conference on Economics and Computation (EC)*, 2024.

Brian Hu Zhang, Tao Lin, Yiling Chen, and Tuomas Sandholm. Learning a Game by Paying the Agents. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=8yRtP2n8OK.

Yizhou Zhang, Yian Ma, and Eric Mazumdar. Learning to steer learners in games. In *Forty-second International Conference on Machine Learning*, 2025.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 928–936, 2003.