Generalized Bandit Regret Minimizer Framework in Imperfect Information Extensive-Form Game

Linjian Meng
Department of Computer Science
Nanjing University
Nanjing, 210000
menglinjian@smail.nju.edu.cn

Yang Gao
Department of Computer Science
Nanjing University
Nanjing, 210000
gaoy@nju.edu.cn

Abstract

Regret minimization methods are a powerful tool for learning approximate Nash equilibrium (NE) in two-player zero-sum imperfect information extensive-form games (IIEGs). We consider the problem in the interactive bandit-feedback setting where we don’t know the dynamics of the IIEG. In general, only the interactive trajectory and the reached terminal node value $v(z^t)$ are revealed. To learn NE, the regret minimizer is required to estimate the full-feedback loss gradient $\ell^t$ by $v(z^t)$ and minimize the regret. In this paper, we propose a generalized framework for this learning setting. It presents a theoretical framework for the design and the modular analysis of the bandit regret minimization methods. We demonstrate that the most recent bandit regret minimization methods can be analyzed as a particular case of our framework. Following this framework, we describe a novel method SIX-OMD to learn approximate NE. It is model-free and extremely improves the best existing convergence rate from the order of $O(\sqrt{XB/T} + \sqrt{YC/T})$ to $O(\sqrt{M_X/XT} + \sqrt{M_Y/YT})$. Moreover, SIX-OMD is computationally efficient as it needs to perform the current strategy and average strategy updates only along the sampled trajectory.

1 Introduction

Imperfect information extensive-form games (IIEGs) are a standard class of games that can be used to model multiple agents, imperfect information, and random events. In such games, players can only make decisions based on their private information and public information. In other words, players do not know the true underlying state of the game. IIEGs have been widely used for modeling real-world applications, such as medical treatment [Sandholm, 2015], security games [Lisy et al., 2016], cybersecurity games [Chen et al., 2017], and recreational games [Brown and Sandholm, 2019b].

Recent research generally aims at learning approximate Nash equilibrium to construct competitive strategy in IIEGs. Nash equilibrium provides a specific concept for rational behavior in these games where no player can improve by deviating unilaterally from the equilibrium. Especially in two-person zero-sum games, any player who chooses to use a Nash equilibrium is guaranteed to not lose in expectation no matter what the opponent does. In principle, in two-person zero-sum IIEGs, the most popular method for learning approximate Nash equilibrium is regret minimization methods [Hoda et al., 2010, Zinkevich et al., 2007, Bowling et al., 2015, Brown and Sandholm, 2019a, Farina et al., 2020a, 2021a]. They have been used to construct many AI milestones in poker [Moravčík et al.,

$^1X, Y$ are the number of information sets and $A, B$ are the number of actions for the two players.

$^2M_X$ and $M_Y$ are defined in Section 5.2, they are extremely less than $X$ and $Y$. 

Preprint. Under review.
words, we must have a complete understanding of the game to calculate the gradient. This setting is also known as full-feedback. In most real-world scenarios, the full-feedback setting is prohibitively expensive and limits the application of regret minimization methods. To eliminate the requirement of the full-feedback, a more common setting for strategy learning is interactive bandit-feedback [Farina and Sandholm, 2021]. This setting has been widely used to build most Multi-Agent Reinforcement Learning (MARL) algorithms. It is a more challenging setting than full-feedback where we only observe the interactive trajectory and the payoff of the reached terminal node. We have no prior knowledge of the game, including the tree structure, the transition probabilities, and the observation/state information [Kozuno et al., 2021].

To solve this setting, various approaches have been proposed, including model-based exploration [Zhou et al., 2019] [Zhang and Sandholm, 2021], Online Mirror Descent (OMD) with unbiased/biased loss estimator [Farina et al., 2021b] [Kozuno et al., 2021] [Bai et al., 2022], and Monte-Carlo Counterfactual Regret Minimization (MCCFR) [Lanctot et al., 2009] [Farina et al., 2020b] [Farina and Sandholm, 2021] [Bai et al., 2022]. In a two-player zero-sum IIEG with $X, Y$ information sets (inforsets) and $B, C$ actions for the two players respectively, the best theoretical convergence rate is $O((\sqrt{XB} + \sqrt{YC})/\sqrt{T})$ achieved by a variant of OMD with biased loss estimator [Bai et al., 2022]. We provide an overview comparison of the above algorithms in Table 1.

In this paper, we investigate the problem that learning approximate Nash equilibrium in two-player zero-sum IIEGs with the interactive bandit-feedback setting. To summarize, we provide the following contributions:

- We develop a generalized framework for analyzing and constructing bandit regret minimization methods in IIEGs. In particular, we provide a way to combine any gradient estimator (unbiased or biased), any exploration strategy, any interactive strategy, with any full-feedback regret minimizer to assemble a bandit regret minimization method.

- We demonstrate that the most recent bandit regret minimization methods, i.e., MCCFR [Lanctot et al., 2009] [Farina et al., 2020b] [Farina and Sandholm, 2021], IXOMD [Kozuno et al., 2021] and balanced OMD/CFR [Bai et al., 2022], can be analyzed as a special case of our framework. We first present the theoretical bounds for biased gradient estimation bandit regret minimization methods in IIEGs.

- We show the validity of our framework. Following the generalized framework, with a simple but effective change, we design a new algorithm SIX-OMD that extremely improves the best known regret results of the min player from $O(\sqrt{TXB})$ to $O(\sqrt{TM})$, and learns an

### Table 1: Comparison of the recent bandit regret minimization methods

| Algorithm                | Model-Free | Convergence Rate                                    |
|--------------------------|------------|-----------------------------------------------------|
| [Lanctot et al., 2009]   | ✓          | $O((X\sqrt{B} + Y\sqrt{C})/\sqrt{T})$               |
| [Zhou et al., 2019]      | ✗          | $O(\max(X\sqrt{B} + Y\sqrt{C}, \sqrt{S})/\sqrt{T})$ |
| Farina et al., 2020b     | ✗          | $O((X\sqrt{B} + Y\sqrt{C})/\sqrt{T})$               |
| Farina and Sandholm, 2021| ✓          | $O(\text{poly}(X, B, Y, C)/T^{1/4})$                |
| Farina et al., 2021b     | ✓          | $O(((XB)^{3/2} + (YC)^{3/2})/\sqrt{T})$             |
| Kozuno et al., 2021      | ✓          | $O((\sqrt{XB} + \sqrt{YC})/\sqrt{T})$              |
| Bai et al., 2022         | ✗          | $O((\sqrt{MX} + \sqrt{MY})/\sqrt{T})$              |
| Our method               | ✓          | $O(\min(X, Y)/\sqrt{T})$                           |

[Farina and Sandholm, 2021]. These methods work by facing a loss gradient from the environment and making a new decision at each time. Specifically, the game tree is accessed only through the sampling strategy for calculating the loss gradient. Many of these methods can learning approximate Nash equilibrium with $O(T^{-1/2})$ or better theoretical convergence rates.

However, the computation of the loss gradient requires traversing the entire game tree. In other words, we must have a complete understanding of the game to calculate the gradient. This setting is also known as full-feedback. In most real-world scenarios, the full-feedback setting is prohibitively expensive and limits the application of regret minimization methods. To eliminate the requirement of the full-feedback, a more common setting for strategy learning is interactive bandit-feedback [Farina and Sandholm, 2021]. This setting has been widely used to build most Multi-Agent Reinforcement Learning (MARL) algorithms. It is a more challenging setting than full-feedback where we only observe the interactive trajectory and the payoff of the reached terminal node. We have no prior knowledge of the game, including the tree structure, the transition probabilities, and the observation/state space [Kozuno et al., 2021].

To solve this setting, various approaches have been proposed, including model-based exploration [Zhou et al., 2019] [Zhang and Sandholm, 2021], Online Mirror Descent (OMD) with unbiased/biased loss estimator [Farina et al., 2021b] [Kozuno et al., 2021] [Bai et al., 2022], and Monte-Carlo Counterfactual Regret Minimization (MCCFR) [Lanctot et al., 2009] [Farina et al., 2020b] [Farina and Sandholm, 2021] [Bai et al., 2022]. In a two-player zero-sum IIEG with $X, Y$ information sets (inforsets) and $B, C$ actions for the two players respectively, the best theoretical convergence rate is $O((\sqrt{XB} + \sqrt{YC})/\sqrt{T})$ achieved by a variant of OMD with biased loss estimator [Bai et al., 2022]. We provide an overview comparison of the above algorithms in Table 1.

In this paper, we investigate the problem that learning approximate Nash equilibrium in two-player zero-sum IIEGs with the interactive bandit-feedback setting. To summarize, we provide the following contributions:

- We develop a generalized framework for analyzing and constructing bandit regret minimization methods in IIEGs. In particular, we provide a way to combine any gradient estimator (unbiased or biased), any exploration strategy, any interactive strategy, with any full-feedback regret minimizer to assemble a bandit regret minimization method.

- We demonstrate that the most recent bandit regret minimization methods, i.e., MCCFR [Lanctot et al., 2009] [Farina et al., 2020b] [Farina and Sandholm, 2021], IXOMD [Kozuno et al., 2021] and balanced OMD/CFR [Bai et al., 2022], can be analyzed as a special case of our framework. We first present the theoretical bounds for biased gradient estimation bandit regret minimization methods in IIEGs.

- We show the validity of our framework. Following the generalized framework, with a simple but effective change, we design a new algorithm SIX-OMD that extremely improves the best known regret results of the min player from $O(\sqrt{TXB})$ to $O(\sqrt{TM})$, and learns an

---

1 $S$ is the size of game state space and only in expectation.
2 Only in expectation.
3 In Markov games.
approximate Nash equilibrium within $O((\sqrt{MT_x} + \sqrt{MT_y})/\sqrt{T})$ rate when all players run in a self-play fashion (defined in Section 4.1).

2 Preliminaries

2.1 Imperfect Information Extensive-Form Games

In this subsection, we introduce the notation that will be used in the following. In this paper, we only focus on two-player zero-sum imperfect information extensive-form games with perfect recall.

Imperfect information extensive-form games (IIEGs) IIEGs are a model of sequential interaction involving multiple agents and represented by a tree rooted at a node $r$ (also called root node). Each node $h$ in the tree belongs to a player from the set $\{0, 1, c\}$, where $c$ is the chance player. We use $P(h)$ to denote the player that acts in node $h$. For each node belonging to the chance player, she picks an action from a distribution known to all others. And $A(h)$ denotes the action available at node $h$. Each node $z$ such that $A(z) = \emptyset$ is called leaf node and represents the terminal state of the game. We employ $Z$ to denote the set of leaf nodes. For each leaf node $z$, there is a pair $(v_0(z), v_1(z)) \in \mathbb{R}^2$ which denotes the payoffs for Player 0 and Player 1, respectively. In our setting, $v_1 = -v_2$. To represent the private information, the nodes for each player $i \in N$ are partitioned into a collection $I_i$, called information sets (inforsets). And perfect recall means that no player will forget any information that has been revealed.

Sequence-Form Strategy [Von Stengel, 1996] The sequence is a decision node-action pair $(I, a)$, where $I$ is an inforset and $a$ is an action belonging to $A(I)$. Each sequence identifies a path from the root node to the inforset $I$ and selects the action $a$. We denote the last sequence encountered on the path to $I$ by $\rho_I$. The set of sequences is denoted as $\sum = \{(I, a) : I \in I_i, a \in A(I) \} \cup \emptyset$, where $\emptyset$ is the empty sequence. Sequence-form strategy for player $i$ is a non-negative vector $x \in \mathbb{R}^{\sum}$ indexed over $\sum$. For each $q = (I, a) \in \sum$, the entry $x[q]$ contains the probability that if the player follows the strategy reaching information set $I$ and selects action $a$ in $I$, and $x[\emptyset] = 1$. To build a valid sequence-form strategy, we need to satisfy the following constraints: (i) $\sum_{a \in A(I)} x[I, a] = x[\rho_I]$ for all inforsets; (ii) $x[\emptyset] = 1$. $\mathcal{X}$ and $\mathcal{Y}$ denote the set of all sequence-form strategies for player 1 and player 2. Sequence-form strategy also known as treeplex which are nonempty convex compact sets in Euclidean spaces $E_x, E_y$. See more details in Appendix B.

2.2 Regret Minimization

In this subsection, we introduce regret minimization methods and the interactive bandit-feedback setting. The concept of regret is from online learning [Zinkevich, 2003].

Regret In this framework, a decision-maker selects a point $x^t$ from the convex compact set $\mathcal{X} \subset \mathbb{R}^d$ and faces a loss gradient $\ell^t \in \mathbb{R}^d$ on each time $t$. The goal is to minimize the regret $R^T(x^*) = \sum_{t=1}^T (\ell^t)^T (x^t - x^*)$, the difference between the cumulative loss of the sequence output $x^1, \ldots, x^T$ and the loss of a fixed point $x^*$. A good regret minimization method is such the regret grows sublinearly compared to any $x^* \in \mathcal{X}$ in $T$.

In online learning, different feedback types impose diverse constraints on regret minimization methods. Considering the specificity of IIEGs, we focus on two settings—the full-feedback setting and the interactive bandit-feedback setting.

Full-Feedback Setting In this setting, the environment reveals the loss gradient $\ell^t$ to the agent. With the theorem in the online learning framework, it is possible to build a regret minimizer that can achieve $O(\sqrt{T})$ regret bound in the worst case, even the loss gradient is chosen adversarially by the environment. Exploiting the regret minimization methods to learn approximate Nash equilibrium with this setting in IIEGs, there are two widely-used ways, counterfactual regret minimization (CFR) [Zinkevich et al., 2007, Bowling et al., 2015, Brown and Sandholm, 2019a] and first-order methods (FOMs) [Hoda et al., 2010, Kroer et al., 2015, Farina et al., 2021a]. The former converts the task of minimizing total regret into minimizing regret on each inforset. And a local simplex regret
where \( x \) and \( z \) with the path from the root node to a leaf node \( z' \) where the payoff is revealed. She only observes the information in the tree, precisely, the interactive trajectory \((I_1, a_1', \ldots, z')\) and the payoff \( v(z')\). Combining with the online learning framework, we have \((\ell^t)^T x^t = \mathbb{E}[v(z')]\), where \( x^t \) is the interactive strategy for \( i \) at time \( t \). According to the property of IIEGs, it is obvious that the loss gradient depends on the environmental dynamics and the opponent’s strategy. Assuming the strategies of chance player and player \( 1 - i \) are \( c \) and \( y \) respectively. Formally, we have (we assume that the environmental dynamics remain invariant over time)

\[
(\ell^t)^T x^t = \sum_{z \in Z} v(z)c[z|x|\rho_x]y|\rho_z|
\]  

(1)

### 2.3 Equilibrium Finding with Regret Minimization

In this subsection, we describe using the regret minimization methods to learn approximate Nash equilibrium in IIEGs. In such games, Nash equilibrium can be formulated as the solution of a Bilinear Saddle Point Problem (BSPP):

\[
\min_{x \in X} \max_{y \in Y} \langle x, Ay \rangle = \max_{y \in Y} \min_{x \in X} \langle x, Ay \rangle
\]  

(2)

Where \( x \) and \( y \) are the sequence-form strategies w.r.t player 0 and player 1, \( A \) is the payoff matrix. Given a pair \([x; y] \in X \times Y\), researchers normally quantify the accuracy of the candidate solution by saddle-point gap

\[
\varepsilon([x; y]) = \max_{y \in Y} \langle x, Ay \rangle - \min_{x \in X} \langle x, Ay \rangle
\]  

(3)

To learning approximate Nash equilibrium, we aim to find a candidate pair \([x; y]\) whose saddle-point gap converges to 0.

#### Learning Procedure

Regret Minimization can be used to achieve the goal. Formally, the first step is to instantiate two regret minimizers \( R_0 \) and \( R_1 \) for strategy sets \( X \) and \( Y \). Then on each iteration \( t \), regret minimizers \( R_0 \) and \( R_1 \) face the loss \( \ell^t_0 \) and \( \ell^t_1 \), output the strategies \( x^{t+1} \) and \( y^{t+1} \), which are used to produce the loss gradient in the next time \( t + 1 \):

\[
\ell^{t+1}_0 = Ay^{t+1}, \quad \ell^{t+1}_1 = -A^T x^{t+1}
\]  

(4)

Let \( \bar{x}, \bar{y} \) denotes the average strategies output by \( R_0 \) and \( R_1 \). A folk theorem shows

\[
\varepsilon([\bar{x}; \bar{y}]) \leq (R^2_0 + R^2_1)/T
\]  

(5)

However, in the interactive bandit-feedback setting, we don’t know the loss gradient \( \ell^t \) and only have the value \( v(z') \) (a real number). So in this setting, before using the full-feedback regret minimizer, it is necessary to estimate the the loss gradient \( \ell^t \) by \( v(z') \). There are two ways—the unbiased estimator [Lanctot et al., 2009] and [Farina and Sandholm, 2021] and the biased estimator [Kozuno et al., 2021] and [Bai et al., 2022]. The former enables the expectation of the output estimated gradient \( \mathbb{E}[-\ell^t] = \mathbb{E}[\ell^t|e^{t_1}, \ldots, e^{t-1}] = \ell^t \), but the latter drops this constraint. After estimating, the estimated gradient \( \hat{\ell}^t \) is fed to a full-feedback regret minimizer, such as CFR and OMD.

### 3 Our Framework

In this section, we describe our generalized regret minimization framework in IIEGs with the interactive bandit-feedback setting. It is composed of a gradient Constructor, a gradient sampler, a full-feedback regret minimizer, a strategy explorer, and a strategy sampler. All independent components of our framework are connected by wires on which the loss gradient and the decision can flow. The framework encodes the entire process of the interactive bandit-feedback setting and helps simplify the design and analysis of the algorithms.
3.1 Overview

In this subsection, we provide an overview of our framework, as shown in Figure 1. Each component of our framework is interchangeable. In other words, the various combination of them involves different algorithms.

**Gradient Constructor** The symbol Constructor denotes the generator operation of the full-feedback loss gradient \( \ell_t \) at time \( t \). Using the opponent’s strategy \( y^t \) and the payoff matrix \( A \) as the inputs, it outputs the gradient by \( \ell_t = Ay^t \).

**Gradient Estimator** After interacting with the environment with the interactive strategy \( \omega^t \), we have the interactive trajectory and the value \( (\ell^t)^T \pi^t \) where \( \pi^t \) is a pure strategy sampled from \( \omega^t \). There are two choices to reconstruct the gradient \( \hat{\ell}^t \)—the unbiased estimator \( \hat{\ell}_E \) and the biased estimator \( \tilde{\ell}_E \). In practice, we only activate one estimator.

**Internal Regret Minimizer** After estimating the gradient, the estimated \( \tilde{\ell}_E \) or \( \hat{\ell}_E \) is fed to a full-feedback regret minimizer \( \tilde{R} \) which guarantees a sublinear regret bound with high probability. In practice, \( \tilde{R} \) is generally CFR or OMD.

**Strategy Explorer** Considering the key issue in reinforcement learning (RL) (a similar setting as the interactive bandit-feedback), exploration and exploitation, we add an explorer behind the output decision \( \tilde{x}^{t+1} \) of \( \tilde{R} \). In principle, the explorer outputs the mixture strategy \( \tilde{x}^{t+1} = (1 - \lambda^{t+1})\hat{x}^{t+1} + \lambda^{t+1}\xi^{t+1} \), where \( \xi^{t+1} \) is the exploration strategy and \( \lambda^{t+1} \in [0, 1) \) is a constant.

**Strategy Sampler** The symbol S1 is the strategy sampler which samples a pure strategy \( x^{t+1} \) from \( \hat{x}^{t+1} \) with the constraint \( E_i[x_i^{t+1}] = \tilde{x}^{t+1} \) to reduce the cost of computing and storing the average strategy since we only update on the inforsets that can be reached by using \( x_i^{t+1} \) while all inforsets that have been sampled by using \( x_i^{t+1} \).

3.2 Main Theorem

To prove the validity of our framework, the most important is the following theorem, which extends the above folk theorem. (See the proof in Appendix A.1)

**Theorem 1.** In a two-player zero-sum HEG with perfect recall, let each player \( i \) have a regret minimizer \( R_i \). At each time \( t \), the regret minimizer \( R_i \) makes a decision \( x_i^t \) and faces a loss gradient \( \ell_i^t \). And for all \( \hat{x}_i \in X_i \), the regret of \( R_i \) at time \( T \) is

\[
P \left( R_i^T(\hat{x}_i) = \sum_{t=1}^{T} <\ell_i^t, x_i^t - \hat{x}_i > \leq M_i \right) \geq 1 - \delta_i \quad (6)
\]

where \( \delta_i \in (0, 1) \). If the loss gradient \( \ell_i^t = Ax_i^{t-1} \) for each player \( i \) at all times \( t \) and each \( M_i \) grows sublinearly, the average strategy \( <\bar{x}_0, \bar{x}_1> \) converges to Nash equilibrium with probability at least \( 1 - (\delta_0 + \delta_1) \).
From Theorem 1, we observe that the final average strategy will converge to Nash equilibrium with a high probability regardless of the internal structure of the regret minimizer \( R_t \) as long as the input loss gradient satisfies \( \ell_t^* = A x_t^{\gamma} \). As shown in Figure 1, we can obtain the convergence guarantee of the average strategy as long as the output decision is the input of the adversary regret minimizer at the next time whether it is \( \tilde{x}^t \), \( \tilde{x}^t \), or \( x^t \).

4 Components of Our Framework

4.1 Gradient Estimate Module

The most important component in the interactive bandit-feedback setting is the gradient estimator. In our framework, there are two gradient estimators—\( \tilde{E} \), and \( \hat{E} \). To simplify the analysis of the gradient estimator, we provide a more understandable sketch in Figure 2.

![Figure 2: The sketch of the gradient estimate module. The symbol \( \ell^t \) is the gradient, \( \sim \) represents the gradient estimator, \( \hat{\ell}^t \) denotes the estimated gradient, \( \hat{R} \) is the regret minimizer that outputs \( x^t \) at time \( t \).](image)

The upper gradient estimators \( \hat{E} \) is unbiased estimator which guarantees \( \ell^t = \mathbb{E}_t[\hat{\ell}^t] \). For this case, the following can be shown (see proof in Appendix A.2).

**Proposition 1.** In a two-player zero-sum IIEG with perfect recall, let \( \ell^t = \mathbb{E}_t[\hat{\ell}^t] \), the regret minimizer \( \hat{R} \) guarantee the regret \( \hat{R}^T(u) = \sum_{t=1}^{T} < \hat{\ell}^t, x^t - u > \leq \hat{M} \) with probability at least \( 1 - \hat{\delta} \) where \( \hat{\delta} \in (0,1) \), \( \hat{M} \) and \( \hat{M}' \) be positive constants such that \( |(\hat{\ell}^t)^T(z - z')| \leq \hat{M} \) and \( |(\hat{\ell}^t)^T(z - z')| \leq \hat{M}' \) for all \( t \in [T] \) and all feasible points \( z, z' \in X \), then for all \( u \in X \) and all \( \delta \in (0,1) \), we have

\[
P \left( \hat{R}^T(u) \leq \hat{M} + (M + M') \sqrt{2T \log(1/\delta)} \right) \geq 1 - (\hat{\delta} + \delta) \tag{7}
\]

As shown in Proposition 1 using an unbiased estimator incurs an additive regret degradation term that scales proportionally with \( \sqrt{T} \). In other words, the unbiased gradient estimation will lead the external regret bound to be never better than \( O(\sqrt{T}) \) despite the internal regret minimizer guaranteeing a better bound. [Hoda et al. 2010], [Kroer et al. 2020], [Farina et al. 2021a], [Munos et al. 2020].

We omit the details of building the unbiased estimator \( \hat{E} \) since it has been deeply investigated in the literature of bandit regret minimization [Abernethy and Rakhlin 2009], [Bartlett et al. 2008]. We only illustrate the simplest approach. Formally, let \( \tilde{z}^t \) denote the reached leaf node by playing the interactive strategy \( \omega^t \), the gradient is estimated by \( (e_{\rho_{\tilde{z}}} \) is the vector that has zeros everywhere but in the component corresponding to \( \rho_{\tilde{z}} \), where it has a one) \( \hat{\ell}^t \) is a pure strategy, we get \( \omega^t[\rho_{\tilde{z}}] = 1 \) for all \( z \in Z \). In other words, each element in this estimated gradient is 0 except the index corresponding to \( \rho_{\tilde{z}} \) where it is 1.

The biased gradient estimator \( \tilde{E} \) typically employs implicit exploration [Kocák et al. 2014], [Neu 2015], [Lattimore and Szepesvári 2020]. As did previously, \( \tilde{z}^t \) denotes the reached leaf node by playing the interactive strategy \( \omega^t \), the estimator estimates the loss gradient by (Note \( \Delta / 2 - v(z^t) \geq 0 \))

\[
\hat{\ell}^t = \frac{(\frac{1}{2} - v(z^t))}{\omega^t[\rho_{\tilde{z}}] + \gamma_{\rho_{\tilde{z}}}} e_{\rho_{\tilde{z}}} \tag{9}
\]
where $\gamma_{p,t}$ is a constant [Kozuno et al., 2021] or a real number related to the auxiliary strategy $\omega^*$ of the estimator [Bai et al., 2022]. We can observe that the most obvious difference between these two gradient estimation methods is that the biased gradient estimation method adds a bias term $\gamma_{p,t}$ in the denominator. For this case, the following can be shown (see more details in Appendix A.3).

**Proposition 2.** In a two-player zero-sum IIEG with perfect recall, let the biased gradient estimator use implicit exploration, the regret minimizer $\hat{R}$ guarantee the regret $\hat{R}^T(u) = \sum_{t=1}^{T} x_{t}^{(u)} - u > \tilde{M}$ with probability at least $1 - \tilde{\delta}$ where $\tilde{\delta} \in (0, 1)$, $C$ be a positive constant such that $\left(\frac{x^T[q]}{\omega^*} + \gamma_q\right) \leq C$ for all $t \in [T]$ and $q \in \sum$. $M_q$ be the value that $M_x = \max_{x \in \mathcal{X}} \|x\|_1$, then for all $u \in \mathcal{X}$ and all $\bar{\delta}^+, \bar{\delta}^- \in (0, 1)$, with probability at least $1 - (\bar{\delta} + \bar{\delta}^+ + \bar{\delta}^-)$, we have

$$R^T(u) \leq \tilde{M} + \Delta \sum_{t=1}^{T} \sum_{q \in \sum} x_{t}^{(q)} \gamma_q x_{t}^{(q)} + \Delta \sum_{q \in \sum} \frac{\log(1/\bar{\delta}^-)}{\min_{u \in [T]} \gamma_q}$$

The analysis of the two different estimators immediately implies that we can employ any interactive strategy to interact with the environment. In other words, instead of using the previous round’s decision for gradient evaluation (also called self-play), we can use another strategy to explore more states of the game. In the literature of RL, we observe a similar idea called off-policy [Dann et al., 2014, Geist et al., 2014].

**4.2 Exploration Module**

![Figure 3: The sketch of the exploration module. The symbol $\ell^t$ denotes the gradient, $\hat{R}$ represents the regret minimizer that outputs $x^{t+1}$ at time $t$, the explorer outputs the mixture strategy $x^{t+1} = (1 - \xi^{t+1})x^{t+1} + \lambda^{t+1}\xi^{t+1}$ where $\xi^{t+1}$ is the exploration strategy at time $t$ and $\lambda^{t+1} \in [0, 1)$ is a constant.

Considering the trade-off between exploration and exploitation, we add the exploration module since using limited information for regret minimization may enter the local optimal points. To modular analyze the explorer, we provide a more understandable sketch in Figure 3. In this case, we obtain the following (see more details in Appendix A.7).

**Proposition 3.** In a two-player zero-sum IIEG with perfect recall, let the regret minimizer $\hat{R}$ guarantee the regret $\hat{R}^T(u) = \sum_{t=1}^{T} x_{t}^{(u)} - u > \tilde{M}$ with probability at least $1 - \tilde{\delta}$ where $\tilde{\delta} \in (0, 1)$, and $x^t = (1 - \lambda^t)x^t + \lambda^t \xi^t$. Then for all $u \in \mathcal{X}$ and all $\delta \in (0, 1)$, we have

$$P \left( R^T(u) \leq \tilde{M} + \Delta \sum_{t=1}^{T} \lambda^t \right) \geq 1 - (\tilde{\delta} + \delta)$$

Observing the above inequality, the explorer incurs an additive regret degradation term that scales proportionally with the sum of $\lambda^t$. Using Lemma $\sum_{t=1}^{T} 1/\sqrt{T} \leq 2\sqrt{T}$, the sum can grow at a rate of $O(\sqrt{T})$ by setting $\lambda^t = 1/\sqrt{T}$ [Joulani et al., 2017, McMahan, 2017].

**4.3 Strategy Sample Module**

The last module of our framework is the strategy sampler, as shown in Figure 4. This module samples a strategy $x^{t+1}$ from the output strategy $\hat{x}^{t+1}$ of the regret minimizer $\hat{R}$ with $\hat{x}^{t+1} = \mathbb{E}[x^{t+1}]$ to reduce the time of updating the average strategy. The pure strategy is a sparse vector in which
each element is 0 or 1 and means that the player only has a deterministic action to select on each information set. We are only required to update the average strategy on the sequences that \( \hat{x}_t[q] = 1 \) while all with \( \tilde{x}_t+1 \). In this case, we can prove the following (see more details in Appendix A.8).

**Proposition 4.** In a two-player zero-sum IIEG with perfect recall, let the regret minimizer \( \tilde{R} \) guarantee the regret \( \tilde{R}_T(u) = \sum_{t=1}^{T} < \ell_t, \tilde{x}_t-u > \leq \tilde{M} \) with probability at least \( 1 - \delta \) where \( \delta \in (0, 1) \), let \( \tilde{x}_t = E_t[x_t] \). Then for all \( u \in X \) and all \( \delta \in (0, 1) \), we have

\[
P \left( R_T(u) \leq \tilde{M} + \Delta \sqrt{2T \log(1/\delta)} \right) \geq 1 - (\tilde{\delta} + \delta)
\]

(12)

From the above inequality, we can reduce the cost of updating average strategy by adding a \( \Delta \sqrt{2T \log(1/\delta)} \) degradation regret term. In large-scale games, it is useful. In such games, the number of sequences that need to be updated by using \( \tilde{x}_t+1 \) is exponentially larger than \( x_t+1 \) while the value of the term \( \Delta \sqrt{2T \log(1/\delta)} \) is small since it depends on \( \Delta \). In other words, it is independent of the size of the game.

5 Application: Reconstruction of Existing Works

In this section, we show our framework can be used to reconstruct existing works, such as MCCFR (outcome sampling) [Lanctot et al., 2009], the model-free algorithm proposed by [Farina and Sandholm, 2021], IXOMD [Kozuno et al., 2021] and Balanced OMD/CFR [Bai et al., 2022].

5.1 Unbiased Gradient Estimation Methods

In this subsection, we describe that using our framework to construct the unbiased gradient estimation methods, including MCCFR (outcome sampling) and the model-free algorithm proposed by [Farina and Sandholm, 2021]. Precisely, the unbiased gradient estimator \( \tilde{E} \) is activated while the biased gradient estimator \( \hat{E} \), the strategy explorer and the strategy sampler are all turned off, with probability at least \( 1 - \delta \), the regret of MCCFR is

\[
R_T(u) \leq \tilde{R}_T(u) + (\Delta + \Delta \cdot max_{q \in Q} \sum_{t \in [T]} 1/\omega_t[q]) \sqrt{2T \log(1/\delta)}
\]

(14)

Compared with MCCFR, the algorithm proposed by [Farina and Sandholm, 2021] demonstrates better practical performance, despite its lower theoretical convergence speed. We now construct the algorithm with our framework by applying the same gradient estimator as MCCFR. However, the
strategy explorer and the strategy sampler are activated in this algorithm. As discussed above, they incur \( \Delta \sum_{t=1}^{T} \lambda t \) and \( \Delta \sum_{t=1}^{T} \sqrt{2T \log (3/\delta)} \) additive regret degradation terms respectively. Each agent uses the last time decision \( x^t \) as the interactive strategy \( \omega^t \). Using the Proposition 2 in [Farina and Sandholm [2021]], with probability at least \( 1 - \delta \), the regret of the algorithm is (see more details about \( \omega \) in [Farina and Sandholm [2021]])

\[
R^T(u) \leq \tilde{R}^T(u) + \Delta \sum_{t=1}^{T} \lambda t + \Delta (1 + \frac{1}{\lambda^2}) \sqrt{2T \log (3/\delta)}
\]

From their analysis, the theoretical bound \( O(T^{3/4}) \) of this algorithm is limited by \( \lambda T \). However, if we don’t use \( x^t \) as the interactive strategy \( \omega^t \), then the term \( \frac{1}{\lambda^2} \) will become a time-independent variable. For example, the most simple method is sampling a pure strategy from the last time decision \( x^t \) as the interactive strategy \( \omega^t \). Then the term \( \Delta (1 + \frac{1}{\lambda^2}) \) will becomes \( 2\Delta \). It is obvious that the term can increase \( O(\sqrt{T}) \) with subtle design. In other words, the bound can be improved to \( O(\sqrt{T}) \).

To some extent, this can explain why the theoretical bound of this algorithm is worse than MCCFR but the practical convergence is better.

5.2 Biased Gradient Estimation Methods

In this subsection, we demonstrate that our framework can be used to construct unbiased gradient estimation methods, such as IXOMD and Balanced OMD/CFR, where the biased gradient estimator \( \tilde{E} \) is activated with the unbiased gradient estimator disabled.

In the original literature of these methods, the environment is considered an imperfect Markov game. Now, as far as we know, we firstly provide their theoretical regret bounds in IIEGs. These methods all employ a biased gradient estimator with implicit exploration except the different choice of the variable \( \gamma^t \).

In IXOMD, \( \gamma_q = \gamma \) is a positive constant for all \( t \in [T] \) and \( q \in \mathbb{S} \). The method uses \( x^t \) as the interactive strategy \( \omega^t \) at time \( t \). Using our framework, with probability at least \( 1 - \delta \), the regret of IXOMD for the min player is (see more details in Appendix A.9)

\[
R^T(u) \leq M + 2\sqrt{TXB} \|A' + D\|_{\infty} \log (3/\delta) M_X + \Delta \sqrt{2T \log (3/\delta)}
\]

(16)

where \( M_X = \max_{u \in X} ||u||_1 \). Actually, \( M_X \) is the maximum number of information sets with nonzero probability of being reached when player 1 has to follow a pure strategy while the other player may follow a mixed strategy [Kroer et al. [2020]]

In balanced OMD, \( \gamma_q = \gamma \omega^*[q] \) where \( \omega^* \) is the auxiliary strategy which is model-based. As did in IXOMD, the interactive strategy \( \omega^t = x^t \). By using our framework, with probability at least \( 1 - \delta \), the regret of balanced OMD for the min player is (see more details in Appendix A.10)

\[
R^T(u) \leq M + 2\sqrt{2T \log (3/\delta) XB} + \Delta \sqrt{2T \log (3/\delta)}
\]

(17)

We don’t demonstrate the details of the construction of Balanced CFR since it is too similar to Balanced OMD, except the full-feedback regret minimizer is replaced with CFR and using a different interactive strategy \( \omega^t \), which is a mixture of \( x^t \) and \( \omega^* \).

6 Application: SIX-OMD

In this section, we describe our new method SIX-OMD. To our knowledge, it first obtains \( O(\sqrt{T M_X}) \) regret bound, an exponential improvement compared to the current state-of-the-art regret bound \( O(\sqrt{T XB}) \) provided by Balanced OMD [Bai et al. [2022]]. In addition, our method is model-free and can exponentially reduce the time of updating the average strategy.

Following our framework, a little change can lead to a huge improvement. SIX-OMD is a variant of IXOMD and only adds a strategy sample module. We provide an overview for SIX-OMD in Figure 5. In this case, the following regret bound can be shown (see proof in Appendix A.11)
Figure 5: The overview of SIX-OMD. The symbol $\ell_t$ denotes the full-feedback gradient, $\tilde{E}$ represents the biased estimator, $\tilde{\ell}_t$ is the estimated gradient, $\tilde{R}$ is the regret minimizer and outputs $\tilde{x}^{t+1}$ at time $t$, the sampler samples a pure strategy $x^{t+1}$ from $\tilde{x}^{t+1}$.

**Proposition 5.** In a two-player zero-sum IIEG with perfect recall, let the OMD regret minimizer $\tilde{R}$ guarantee the regret $\tilde{R}^T(u) = \sum_{t=1}^{T} <\ell_t, \tilde{x}_t^t - u> \leq \tilde{M}$ with probability at least $1 - \delta/4$ where $\delta \in (0, 1)$. Then with probability at least $1 - \delta$, the regret of SIX-OMD is (for the min player)

$$R^T(u) \leq \tilde{M} + 2\sqrt{2T\log(4/\delta)M_X} + \Delta \sqrt{2T\log(4/\delta)}$$  \hspace{1cm} (18)

We don’t show the details of the internal regret minimizer since the regret bound of SIX-OMD doesn’t depend on the internal regret minimizer. There are various variants of OMD that can be chosen to get better regret bound than $o(\sqrt{T})$. For example, we can employ Optimistic OMD proposed by [Farina et al., 2019] which achieves $O(1)$ regret bound in IIEGs. This bound is extremely less than $O(\sqrt{T})$.

As mentioned in Section 5.2 $M_X = \max_{u \in X} ||u||_1$. From the definition of treeplex (see more details in [B]), it is obvious that $M_X$ is exponentially smaller than $XB$ or $X$. Furthermore, in almost all games, $\Delta = 2 \leq M_X$ always holds by rescaling the range of payoffs to $[-1, 1]$. These imply that the regret bound of SIX-OMD for the min player is $O(\sqrt{T M_X})$ ($O(\sqrt{T M_Y})$ for the max player).

In addition, as we discussed in Section 4.3 adding a strategy sample module will extremely reduce the time of updating average strategy. So compared to the current SOTA method [Kroer et al., 2020], our method is model-free, provides better regret bound, and has less computation time.

### 7 Conclusion

We develop a new generalized bandit regret minimizer framework for solving IIEGs with perfect recall assumption. It facilitates a modular analysis of existing interactive bandit-feedback regret minimization methods and allows us to integrate various algorithmic techniques easily. We show that our framework can be used to construct MCCFR, the algorithm proposed by [Farina and Sandholm, 2021], IXOMD and Balanced OMD/CFR, and analyze their theoretical regret bound with our framework. Furthermore, to our knowledge, we presents the first line of algorithms for learning Nash equilibrium in a rate of $O(\sqrt{M_X/T} + \sqrt{M_Y/T})$. In addition, this method is model-free and exponentially reduce the time of updating the average strategy. We believe it is an interesting future direction to combine the most recent technological advances in MARL like CTDE [Oliehoek et al., 2008, Kraemer and Banerjee, 2016, Rashid et al., 2018] with our framework to solve more general IIEG-like problems.
References

Jacob Abernethy and Alexander Rakhlin. Beating the adaptive bandit with high probability. In 2009 Information Theory and Applications Workshop, pages 280–289. IEEE, 2009.

Kazuoki Azuma. Weighted sums of certain dependent random variables. Tohoku Mathematical Journal, Second Series, 19(3):357–367, 1967.

Yu Bai, Chi Jin, Song Mei, and Tiancheng Yu. Near-optimal learning of extensive-form games with imperfect information. arXiv preprint arXiv:2202.01752, 2022.

Peter Bartlett, Varsha Dani, Thomas Hayes, Sham Kakade, Alexander Rakhlin, and Ambuj Tewari. High-probability regret bounds for bandit online linear optimization. In Proceedings of the 21st Annual Conference on Learning Theory-COLT 2008, pages 335–342. Omnipress, 2008.

Michael Bowling, Neil Burch, Michael Johanson, and Oskari Tammelin. Heads-up limit hold’em poker is solved. Science, 347(6218):145–149, 2015.

Noam Brown and Tuomas Sandholm. Superhuman ai for heads-up no-limit poker: Libratus beats top professionals. Science, 359(6374):418–424, 2018.

Noam Brown and Tuomas Sandholm. Solving imperfect-information games via discounted regret minimization. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, pages 1829–1836, 2019a.

Noam Brown and Tuomas Sandholm. Superhuman ai for multiplayer poker. Science, 365(6456):885–890, 2019b.

Xianfu Chen, Zhu Han, Honggang Zhang, Guoliang Xue, Yong Xiao, and Mehdi Bennis. Wireless resource scheduling in virtualized radio access networks using stochastic learning. IEEE Transactions on Mobile Computing, 17(4):961–974, 2017.

Christoph Dann, Gerhard Neumann, Jan Peters, et al. Policy evaluation with temporal differences: A survey and comparison. Journal of Machine Learning Research, 15:809–883, 2014.

Gabriele Farina and Tuomas Sandholm. Model-free online learning in unknown sequential decision making problems and games. arXiv preprint arXiv:2103.04539, 2021.

Gabriele Farina, Christian Kroer, and Tuomas Sandholm. Optimistic regret minimization for extensive-form games via dilated distance-generating functions. Advances in Neural Information Processing Systems, 32, 2019.

Gabriele Farina, Christian Kroer, and Tuomas Sandholm. Faster game solving via predictive blackwell approachability: Connecting regret matching and mirror descent. arXiv preprint arXiv:2007.14358, 2020a.

Gabriele Farina, Christian Kroer, and Tuomas Sandholm. Stochastic regret minimization in extensive-form games. In International Conference on Machine Learning, pages 3018–3028. PMLR, 2020b.

Gabriele Farina, Christian Kroer, and Tuomas Sandholm. Better regularization for sequential decision spaces: Fast convergence rates for nash, correlated, and team equilibria. arXiv preprint arXiv:2105.12954, 2021a.

Gabriele Farina, Robin Schmucker, and Tuomas Sandholm. Bandit linear optimization for sequential decision making and extensive-form games. arXiv preprint arXiv:2103.04546, 2021b.

Matthieu Geist, Bruno Scherrer, et al. Off-policy learning with eligibility traces: a survey. J. Mach. Learn. Res., 15(1):289–333, 2014.

Samid Hoda, Andrew Gilpin, Javier Pena, and Tuomas Sandholm. Smoothing techniques for computing nash equilibria of sequential games. Mathematics of Operations Research, 35(2):494–512, 2010.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. In The collected works of Wassily Hoeffding, pages 409–426. Springer, 1994.
Pooria Joulani, András György, and Csaba Szepesvári. A modular analysis of adaptive (non-) convex optimization: Optimism, composite objectives, and variational bounds. In *International Conference on Algorithmic Learning Theory*, pages 681–720. PMLR, 2017.

Tomáš Kocák, Gergely Neu, Michal Valko, and Rémi Munos. Efficient learning by implicit exploration in bandit problems with side observations. *Advances in Neural Information Processing Systems*, 27, 2014.

Tadashi Kozuno, Pierre Ménard, Rémi Munos, and Michal Valko. Model-free learning for two-player zero-sum partially observable markov games with perfect recall. *arXiv preprint arXiv:2106.06279*, 2021.

Landon Kraemer and Bikramjit Banerjee. Multi-agent reinforcement learning as a rehearsal for decentralized planning. *Neurocomputing*, 190:82–94, 2016.

Christian Kroer, Kevin Waugh, Fatma Kilinc-Karzan, and Tuomas Sandholm. Faster first-order methods for extensive-form game solving. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, pages 817–834, 2015.

Christian Kroer, Kevin Waugh, Fatma Kilinc-Karzan, and Tuomas Sandholm. Faster algorithms for extensive-form game solving via improved smoothing functions. *Mathematical Programming*, 179 (1):385–417, 2020.

Marc Lanctot, Kevin Waugh, Martin Zinkevich, and Michael Bowling. Monte carlo sampling for regret minimization in extensive games. *Advances in neural information processing systems*, 22:1078–1086, 2009.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Viliam Lisy, Trevor Davis, and Michael Bowling. Counterfactual regret minimization in sequential security games. In *Thirtyith AAAI conference on artificial intelligence*, 2016.

H Brendan McMahan. A survey of algorithms and analysis for adaptive online learning. *The Journal of Machine Learning Research*, 18(1):3117–3166, 2017.

Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisý, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337):508–513, 2017.

Remi Munos, Julien Perolat, Jean-Baptiste Lespiau, Mark Rowland, Bart De Vylder, Marc Lanctot, FinbarTimbers, Daniel Hennes, ShayegeoMomidshafiei, AudrunasGruslys, et al. Fast computation of nash equilibria in imperfect information games. In *International Conference on Machine Learning*, pages 7119–7129. PMLR, 2020.

Gergely Neu. Explore no more: Improved high-probability regret bounds for non-stochastic bandits. *Advances in Neural Information Processing Systems*, 28, 2015.

Frans A Oliehoek, Matthijs TJ Spaan, and Nikos Vlassis. Optimal and approximate q-value functions for decentralized pomdpds. *Journal of Artificial Intelligence Research*, 32:289–353, 2008.

Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 4295–4304. PMLR, 2018.

Tuomas Sandholm. Steering evolution strategically: Computational game theory and opponent exploitation for treatment planning, drug design, and synthetic biology. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

Bernhard Von Stengel. Efficient computation of behavior strategies. *Games and Economic Behavior*, 14(2):220–246, 1996.

Brian Zhang and Tuomas Sandholm. Finding and certifying (near-) optimal strategies in black-box extensive-form games. In *AAAI Workshop on Reinforcement Learning in Games*, 2021.
Yichi Zhou, Jialian Li, and Jun Zhu. Posterior sampling for multi-agent reinforcement learning: solving extensive games with imperfect information. In *International Conference on Learning Representations*, 2019.

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.

Martin Zinkevich, Michael Johanson, Michael Bowling, and Carmelo Piccione. Regret minimization in games with incomplete information. *Advances in neural information processing systems*, 20: 1729–1736, 2007.
A Proofs

A.1 Proof of Theorem 1

Now, we have

\[ P \left( R^T(\hat{x}) = \sum_{t=1}^{T} <Ay, x^t - \hat{x}> \leq M_0 \right) \geq 1 - \delta_0 \]  \hspace{1cm} (19)

\[ P \left( R^T(\hat{y}) = \sum_{t=1}^{T} <-A^T x, y^t - \hat{y}> \leq M_1 \right) \geq 1 - \delta_1 \]  \hspace{1cm} (20)

Taking the union bound, we have

\[ P \left( \left( \sum_{t=1}^{T} <Ay, x^t - \hat{x}> + \sum_{t=1}^{T} <-A^T x, y^t - \hat{y}> \right) \leq (M_0 + M_1) \right) \geq 1 - (\delta_0 + \delta_1) \]  \hspace{1cm} (21)

Then,

\[ P \left( (\hat{x}^T Ay - \hat{x}^T A\hat{y}) \leq (M_0 + M_1) \right) \geq 1 - (\delta_0 + \delta_1) \]  \hspace{1cm} (22)

This includes the proof.

A.2 Proof of Proposition 1

As observed in the body, \( d^t = (\ell_t^t - \tilde{\ell}_t^t)^T (x^t - u^t) \) is a martingale difference sequence. Furthermore, at all times \( t \)

\[ |d^t| = |(\ell_t^t)^T (x^t - u^t) - (\tilde{\ell}_t^t)^T (x^t - u^t)| \]
\[ \leq |(\ell_t^t)^T (x^t - u^t)| + |(\tilde{\ell}_t^t)^T (x^t - u^t)| \]
\[ \leq M + M' \]

which means \(- (M + M') \leq d^t \leq (M + M')\) for each \( t \). And we have,

\[ \sum_{t=1}^{T} d^t = \left( \sum_{t=1}^{T} (\ell_t^t)^T (x^t - u^t) \right) - \left( \sum_{t=1}^{T} (\tilde{\ell}_t^t)^T (x^t - u^t) \right) = R^T(u) - \tilde{R}^T(u) \]  \hspace{1cm} (24)

Then by Azuma-Hoeffding inequality [Azuma 1967, Hoeffding 1994], we have

\[ P \left( \sum_{t=1}^{T} d^t = R^T(u) - \tilde{R}^T(u) \leq (M + M') \sqrt{2T \log (1/\delta)} \right) \geq 1 - \delta \]  \hspace{1cm} (25)

Finally, combining the equality with the union bound yields the proof.

A.3 Proof of Proposition 2

Before proving Proposition 2, we have to make some modifications. First, we scale the range of payoffs from \([-\frac{\Delta}{2}, \frac{\Delta}{2}]\) to \([0, 1]\) by setting the payoffs in each leaf node for all players are \( \frac{v_1(z^1)}{\Delta} + \frac{1}{2} \) and \( \frac{v_2(z^2)}{\Delta} + \frac{1}{2} \) respectively. Following these, then we scale the range of the values in loss gradient from \([-\frac{\Delta}{2}, \frac{\Delta}{2}]\) to \([0, 1]\) for all players by setting the losses in each leaf node for all players are \( \frac{1}{2} - \frac{v_1(z^1)}{\Delta} \) and \( \frac{1}{2} - \frac{v_2(z^2)}{\Delta} \) respectively. It implies that if the min player plays \( x \), the max plays \( y \), the loss gradients for them are \( (A' + D)y \) and \( -(A' + D)^T x \) respectively (the new gradient is denoted by \( \ell \) to distinguish with \( \ell \), and \( A' = A/\Delta \)). For all \( x, x' \in X \) and \( y, y' \in Y \), we have

\[ x^T Dy = x'^T Dy = x^T Dy' = x'^T D y' = \frac{1}{2} \]  \hspace{1cm} (26)

Following these, then we get

\[ R^T(u) = \sum_{t=1}^{T} \langle x^t - u, \ell^t \rangle = \Delta \sum_{t=1}^{T} \langle x^t - u, \tilde{\ell}^t \rangle \]  \hspace{1cm} (27)
Decompose the regret as

$$R^T(u) = \Delta \sum_{t=1}^{T} < x^t - u, \tilde{e}^t >$$

$$\Delta \left( \leq \sum_{t=1}^{T} < x^t, \tilde{e}^t - \hat{e}^t > + \sum_{t=1}^{T} < u, \tilde{e}^t - \hat{e}^t > + \sum_{t=1}^{T} < x^t - u, \hat{e}^t > \right)$$

(28)

We now state two lemmas that bound each of the former two terms since the last one is the regret of the internal regret minimizer. The proofs are presented in Section A.5 and A.6 respectively.

**Lemma 1.** With probability at least $1 - \delta^+$, we have

$$BIAS^1 \leq \sum_{t=1}^{T} \sum_{q \in \Sigma} x^t[q] \gamma_t \tilde{e}^t[q] \omega^t(q) + \gamma_t + \Delta \sqrt{2T \log (1/\delta^+)}$$

(29)

**Lemma 2.** With probability at least $1 - \delta^-$, we have

$$BIAS^2 \leq \log(1/\delta^-) \sum_{q \in \Sigma} \frac{u[q]}{\min_{t \in [T]} \gamma_t}$$

(30)

Putting the bounds together, we have with probability at least $1 - (\delta + \delta^+ + \delta^-)$

$$R^T(u) \leq \tilde{M} + \Delta \sum_{t=1}^{T} \sum_{q \in \Sigma} x^t[q] \gamma_t \tilde{e}^t[q] \omega^t(q) + \gamma_t + \Delta \sqrt{2T \log (1/\delta^+)} + \Delta \sum_{q \in \Sigma} \log(1/\delta^-) u[q] \frac{\min_{t \in [T]} \gamma_t}{\gamma_t}$$

(31)

### A.4 A Concentration Result

To prove the Lemma 1 and Lemma 2, we provide a useful concentration result in the following. The result is a variant of [Bai et al., 2022] and [Neu, 2015].

**Lemma 3.** Let $\alpha^t(q) \in [0, 2\gamma_q]$ be $\mathcal{F}^{t-1}$ - measure random variable for any $q \in \Sigma$. Then with probability $1 - \delta$,

$$\sum_{t=1}^{T} \alpha^t(q) \left( \hat{e}^t[q] - \tilde{e}^t[q] \right) \leq \log(1/\delta)$$

(32)

**Proof.** Define the unbiased importance-sampling estimate of $\tilde{e}^t$ as $\hat{e}^t$. We have

$$\hat{e}^t[q] = \frac{1}{2} \frac{-v(z^t)}{\omega^t[\rho_{z^t}] + \gamma^t_{\rho_{z^t}}} \cdot 1\{q = \rho_{z^t}\}$$

$$\leq \frac{1}{2} \frac{-v(z^t)}{\omega^t[\rho_{z^t}] + \gamma^t_{\rho_{z^t}}} \cdot 1\{q = \rho_{z^t}\}$$

$$\leq \frac{1}{2} \frac{-v(z^t)}{2\gamma^t_{\rho_{z^t}}} \omega^t[\rho_{z^t}] + \gamma^t_{\rho_{z^t}} \frac{(1/2) - v(z^t)}{\Delta} \cdot 1\{q = \rho_{z^t}\}$$

$$\leq \frac{1}{2} \frac{1}{\gamma^t_{\rho_{z^t}}} \omega^t[\rho_{z^t}] + \gamma^t_{\rho_{z^t}} \frac{(1/2) - v(z^t)}{\Delta} \cdot 1\{q = \rho_{z^t}\}$$

$$\leq \frac{1}{2} \frac{2\gamma^t_{\rho_{z^t}} \hat{e}^t[q]}{1 + \gamma^t_{\rho_{z^t}} \hat{e}^t[q]}$$

with $\frac{z}{1+2z^2} \leq \log(1+z)$ for any $z \geq 0$, we have

$$\hat{e}^t[q] \leq \frac{1}{2\gamma^t_{\rho_{z^t}}} \log \left( 1 + 2\gamma^t_{\rho_{z^t}} \hat{e}^t[q] \right)$$

(34)
Let $\hat{\beta}^t = \sum_{q \in \Sigma} \alpha^t(q)\hat{\ell}_t[q]$ and $\beta^t = \sum_{q \in \Sigma} \alpha^t(q)\ell^t[q]$. Note that we want to show $\sum_{t=1}^T (\hat{\beta}^t - \beta^t) \leq \log(1/\delta)$. Then we have

$$E^t - \exp(\hat{\beta}^t) \leq E^t - \exp(\beta^t)$$

where the second line follows from $z \log (1 + z') \leq \log (1 + zz')$ for any $z \in [0, 1]$ and $z' \in (-1, +\infty)$, the third line follows from $\ell^t[q]\ell^t[q'] = 0$ for any $q \neq q'$, and the last line follows from $1 + z \leq \exp(z)$ for any $z \in \mathbb{R}$.

Define $Z_t = \exp(\hat{\beta}^t - \beta^t)$ and $M_t = \prod_{u=1}^t Z_u$. From the above inequality, we have that $E[M_t] = E[E^{t-1}[M_t]] = E[M_{t-1}E^{t-1}[Z_t]] \leq E[M_{t-1}] \leq \cdots \leq 1$. As a result, Markov’s inequality implies

$$P\left(\sum_{t=1}^T (\hat{\beta}^t - \beta^t) \geq \log(1/\delta)\right) = P\left(\log(M_T) \geq \log(1/\delta)\right)$$

$$= P(M_T \geq 1) \leq E[M_t] \delta \leq \delta$$

This concludes the proof.

A.5 Proof of Lemma 1

First, decompose $BIAS^1$ as

$$BIAS^1 = \sum_{t=1}^T <x^t, \ell^t - \hat{\ell}^t> = \sum_{t=1}^T <x^t, \ell^t - E[\ell^t]> + \sum_{t=1}^T <x^t, E[\ell^t] - \hat{\ell}^t>$$

To bound A, plug in the definition of loss estimator,

$$\sum_{t=1}^T <x^t, \ell^t - E[\ell^t]>$$

$$= \sum_{t=1}^T \sum_{q \in \Sigma} x_t[q] \left[\ell_t[q] - \hat{\ell}_t[q]\omega_t(q) + \gamma_t\right]$$

$$= \sum_{t=1}^T \sum_{q \in \Sigma} x_t[q]\ell_t[q] \left[\frac{\gamma_t}{\omega_t(q) + \gamma_t}\right]$$

$$= \sum_{t=1}^T \sum_{q \in \Sigma} x_t[q]\gamma_t\ell_t[q] \left[\frac{1}{\omega_t(q) + \gamma_t}\right]$$
To bound B, we have
\[
<x^t, \hat{\ell}^t> = \sum_{q \in \Sigma} x^t[q] \left( \frac{1}{2} - \frac{v(z^t)}{\Delta} \right) \cdot 1\{q = \rho_z^t\}
\]
\[
\leq \frac{1}{2} \sum_{q \in \Sigma} x^t[q] \omega^t[q] 1\{q = \rho_z^t\}
\]
(39)
So, we have
\[
<x^t, E[\hat{\ell}^t] - \hat{\ell}^t> \leq \max_{q \in \Sigma} \left( \frac{x^t[q]}{\omega^t[q] + \gamma_q^t} \right) \leq C
\]
(40)
And it is obvious \(E[x^t, E[\hat{\ell}^t] - \hat{\ell}^t] = 0\). Then by Azuma-Hoeffding inequality, with probability at least 1 - \(\delta\),
\[
\sum_{t=1}^{T} <x^t, E[\hat{\ell}^t] - \hat{\ell}^t> \leq C \sqrt{2T \log \frac{1}{\delta}}
\]
(41)
Combining the bounds for A and B gives the desired result.

A.6 Proof of Lemma 2
We have
\[
BIAS^2 = \sum_{t=1}^{T} <u, \hat{\ell}^t - \bar{\ell}^t>
\]
\[
= \sum_{t=1}^{T} \sum_{q \in \Sigma} u[q] \left[ \hat{\ell}^t[q] - \bar{\ell}^t[q] \right]
\]
\[
= \sum_{t=1}^{T} \sum_{q \in \Sigma} \frac{u[q]}{\min_{t \in \{T\}} \gamma_q^t} \min_{t \in \{T\}} \gamma_q^t \left[ \hat{\ell}^t[q] - \bar{\ell}^t[q] \right]
\]
\[
= \sum_{q \in \Sigma} \frac{u[q]}{\min_{t \in \{T\}} \gamma_q^t} \sum_{t=1}^{T} \min_{t \in \{T\}} \gamma_q^t \left[ \hat{\ell}^t[q] - \bar{\ell}^t[q] \right]
\]
\[
\leq \log(1/\delta) \sum_{q \in \Sigma} \frac{u[q]}{\min_{t \in \{T\}} \gamma_q^t}
\]
(42)
Where the inequality is by applying Lemma 3. This completes the proof.

A.7 Proof of Proposition 3
From the definition, we have
\[
\hat{R}^T(u) = \sum_{t=1}^{T} <\ell^t, \hat{x}^t - u>, R^T(u) = \sum_{t=1}^{T} <\ell^t, x^t - u>
\]
(43)
It is obvious that
\[
R^T(u) = \hat{R}^T(u) + \sum_{t=1}^{T} (\ell^t)^T(x^t - \hat{x}^t)
\]
(44)
Substituting \(x^t = (1 - \lambda^t)\hat{x}^t + \lambda^t \xi^t\), we have
\[
\sum_{t=1}^{T} (\ell^t)^T(x^t - \hat{x}^t) = \sum_{t=1}^{T} (\ell^t)^T((1 - \lambda^t)\hat{x}^t + \lambda^t \xi^t - \hat{x}^t)
\]
\[
= \sum_{t=1}^{T} \lambda^t (\ell^t)^T(\xi^t - \hat{x}^t)
\]
\[
\leq \Delta \sum_{t=1}^{T} \lambda^t
\]
(45)
Combining with union bound completes the proof.

A.8 Proof of Proposition 4

From the definition, we have
\[
\hat{R}^T(u) = \sum_{t=1}^{T} <\ell^t, \hat{x}^t - u >, R^T(u) = \sum_{t=1}^{T} <\ell^t, x^t - u >
\]  
(46)

Setting \(\sum_{t=1}^{T} d^t = R^T(u) - \hat{R}^T(u) = \sum_{t=1}^{T} <\ell^t, \ell^t - \hat{x}^t >\), therefore
\[
\mathbb{E}_t[d^t] = 0, |d^t| \leq \Delta
\]  
(47)

By Azuma-Hoeffding inequality, with probability at least \(1 - \delta\), we have
\[
\sum_{t=1}^{T} d^t \leq \Delta \sqrt{2T \log (1/\delta)}
\]  
(48)

Combining with union bound yields the proof.

A.9 Proof of Eq. 16

From the analysis of our framework, with probability at least \(1 - \delta\), the following can be shown (\(\bar{\ell}^t\), \(\bar{A}'\), and \(D\) is defined in Appendix A.3, and \(\ell^t[q] \geq 0\))
\[
R^T(u) \leq \tilde{M} + \Delta \sum_{t=1}^{T} \sum_{q \in \Sigma} \frac{x_t[q] \gamma \bar{\ell}_t[q]}{x_t'[q] + \gamma} + \Delta \sqrt{2T \log (3/\delta)} + \Delta \log (3/\delta) \sum_{q \in \Sigma} \frac{u[q]}{\gamma}
\]
(49)

where the second inequality follows from \(\sum_{q \in \Sigma} \ell_t[q] = \sum_{q \in \Sigma} (A' + D)[q]\), \(\gamma = \sqrt{\log (3/\delta) M_X} / \sqrt{TXB ||A' + D||_{\infty}}\), and the last inequality follows from \(\gamma = \sqrt{2 \log (3/\delta) M_X} / \sqrt{TXB ||A' + D||_{\infty}}\).

A.10 Proof of Eq. 17

From the analysis of our framework, with probability at least \(1 - \delta\), the following can be shown (\(\bar{\ell}^t\) is defined in Appendix A.3, and \(\ell^t[q] \geq 0 \) for all \(q \in \Sigma\))
\[
R^T(u) \leq \tilde{M} + \Delta \sum_{t=1}^{T} \sum_{q \in \Sigma} \frac{x_t[q] \gamma \omega^t[q] \bar{\ell}_t[q]}{x_t'[q] + \gamma \omega^t[q]} + \Delta \sqrt{2T \log (3/\delta)} + \Delta \log (3/\delta) \sum_{q \in \Sigma} \frac{u[q]}{\gamma \omega^t[q]}
\]
(50)

where the second inequality follows from \(\omega^t[q] / (x_t^[q] + \gamma) \leq 1\), the third inequality follows from \(\langle \omega, \ell^t \rangle \leq 2\) and Lemma C.4 in [Bai et al., 2022], and the last inequality follows from \(\gamma = \sqrt{2 \log (3/\delta) M_X} / \sqrt{TXB ||A' + D||_{\infty}}\).
A.11 Proof of Proposition \[5\]

From the analysis of our framework, using Proposition \[2\] and Proposition \[4\] with probability at least \(1 - \delta\), the following can be shown (\(\ell^t\) is defined in Appendix \[A.3\] and \(\ell^t[q] \geq 0\) for all \(q \in \Sigma\))

\[
R^T(u) \leq \tilde{M} + \Delta \sum_{t=1}^{T} \sum_{q \in \Sigma} x^t[q] \bar{\gamma}^t[q] + 2\Delta \sqrt{2T \log \left( \frac{4}{\delta} \right)} + \Delta \log(4/\delta) \sum_{q \in \Sigma} u[q] \tag{51}
\]

Then, with the assumption, \(x^t\) is a pure strategy. It means that \(x^t[q] = 0\) or \(x^t[q] = 1\) for all \(q \in \Sigma\). Then we have

\[
R^T(u) \leq \tilde{M} + \Delta \sum_{t=1}^{T} \sum_{q \in \Sigma} 1\{x^t[q] = 1\} \bar{\gamma}^t[q] + 2\Delta \sqrt{2T \log \left( \frac{4}{\delta} \right)} + \Delta \log(4/\delta) \frac{M_X}{\gamma} \leq \tilde{M} + \frac{\Delta \gamma}{1 + \gamma} \sum_{t=1}^{T} \sum_{q \in \Sigma} x^t[q] \bar{\gamma}^t[q] + 2\Delta \sqrt{2T \log \left( \frac{4}{\delta} \right)} + \Delta \log(4/\delta) \frac{M_X}{\gamma} \leq \tilde{M} + 2\gamma T \Delta + 2\Delta \sqrt{2T \log \left( \frac{4}{\delta} \right)} + \Delta \log(4/\delta) \frac{M_X}{\gamma} \leq \tilde{M} + 2\gamma T \Delta + 2\Delta \sqrt{2T \log(4/\delta)} M_X + 2\Delta \sqrt{2T \log(4/\delta)} \tag{52}
\]

where the second inequality follows from \(M_X = \max_{u \in X} \sum_{q \in \Sigma} u[q]\), the third line follows from the definition of pure strategy, the third inequality follows from \(\langle x^t, \bar{\gamma}^t \rangle \leq 2\), the fourth inequality follows from \(\frac{\gamma}{1 + \gamma} < 1\) for all \(\gamma > 0\), and the last inequality follows from setting \(\gamma = \frac{2\log(4/\delta)}{M_X}\).

B Treeplex

The concept of treeplex was first introduced by [Hoda et al., 2010]. We follow the form describe in [Farina et al., 2019]. Treeplex is a class of convex polytopes that encompass the sequence-form description of strategy spaces in perfect-recall sequential games:

Treeplexes are described recursively:

- Basic sets: The standard simplex \(\Delta_m = \{x \in [0,1]^m : \sum_{k=1}^{m} x_k = 1\}\) is a treeplex.
- Cartesian product: If \(Q_1, \cdots, Q_k\) are treeplexes, then \(Q_1 \times \cdots \times Q_k\) is a treeplex.
- Branching: If \(P\) is a simplex \(\Delta_p\) and \(Q = \{Q_1, \cdots, Q_k\}\), where \(Q_j \subseteq [0,1]^{q_j}\) are treeplexes and \(l = \{l_1, \cdots, l_k\} \subseteq \{1, \cdots, p\}\), then

\[
\overline{P \bigcup Q} = \{(x_1, y_1, \cdots, y_k) \in \mathbb{R}^{p+\sum_{j=1}^{k} q_j} : x \in P, y_1 = x_{l_1} \cdot Q_1, \cdots, y_k = x_{l_k} \cdot Q_k\} \tag{53}
\]

is a treeplex and \(x_{l_i}\) are the branching variables for \(Q_j\).

A treeplex is a tree of simplices, where children are connected to parents through the branching operation, which scales the child simplex by the value of the parent branching variable. In IIEGs, the simplices correspond to the information sets of a single player and the whole treeplex represents that player’s strategy space. The branching operation has a sequential interpretation: The vector \(x\) represents the decision variables at some stage, while the vectors \(y_{l_i}\) represent the decision variables at the \(k\) potential following stages, depending on external outcomes. Here \(k \leq p\), as some variables in \(x\) may not have subsequent decisions. As pointed out by [Von Stengel, 1996], the treeplex can be represented by the linear equations \(E q = e\) for a matrix \(E\) with entries in \{-1, 0, 1\}, a treeplex vector \(q\), and a vector \(e\) with entries in \{0, 1\}. An example treeplex is given in Figure \[6\].
Figure 6: An example treplex constructed from 8 simplices. Cartesian product operation is denoted by $\times$. 

root

$q[0] = 1$

$q[0] \Delta^3 \times q[0] \Delta^2 \times q[0] \Delta^3$

$q[1] \Delta^3$
$q[2] \Delta^2$
$q[10] \Delta^0$

$q[1] \Delta^4 \times q[2] \Delta^5$

$q[1] \Delta^4$
$q[4] \Delta^6$
$q[5] \Delta^6$

$q[2] \Delta^5$
$q[8] \Delta^7$
$q[6] \Delta^7$

$q[4] \Delta^6$
$q[10] \Delta^0$
$q[11] \Delta^0$
$q[12] \Delta^0$

$q[9] \Delta^7$
$q[7] \Delta^7$

$q[11] \Delta^0$
$q[12] \Delta^0$