Analyzing the Sample Complexity of Model-Free Opponent Shaping

Kitty Fung [1] **Qizhen Zhang** [2] **Chris Lu** [2] **Timon Willi** [2] **Jakob Foerster** [2]

Abstract

In mixed-incentive multi-agent environments, methods developed for zero-sum games often yield collectively sub-optimal results. Addressing this, *opponent shaping* (OS) strategies aim to actively guide the learning processes of other agents, empirically leading to enhanced individual and group performances. Early OS methods use higher-order derivatives to shape the learning of co-players, making them unsuitable to anticipate multiple learning steps ahead. Follow-up work Model-free Opponent Shaping (M-FOS) addresses the shortcomings of earlier OS methods by reframing the OS problem into a meta-game. In the meta-game, the meta-step corresponds to an episode of the "inner" game. The OS meta-state corresponds to the inner policies, while the meta-policy outputs an inner policy at each meta-step. Leveraging model-free optimization techniques, M-FOS learns meta-policies that demonstrate long-horizon opponent shaping, e.g., by discovering a novel extortion strategy in the Iterated Prisoner's Dilemma (IPD). In contrast to early OS methods, there is little theoretical understanding of the M-FOS framework. In this work, we derive the sample complexity bounds for the M-FOS agents theoretically and empirically. To quantify the sample complexity, we adapt the R_{max} algorithm, most prominently used to derive sample bounds for MDPs, as the meta-learner in the M-FOS framework. We derive a sample complexity that has an exponential relationship with the cardinality of inner state and action space and the number of agents. Our theoretical results are empirically supported in the Matching Pennies environment.

1 Introduction

Most multi-agent reinforcement learning (MARL) research has focused on fully-cooperative learning in Dec-POMDPs (Ellis et al., 2022) where choices are made separately by a set of decision-makers, or zero-sum games like Starcraft and Go (Silver et al., 2017; Vinyals et al., 2019). However, these situations constitute only a small portion of potential real-world multi-agent environments. General-sum games, which are not entirely cooperative or competitive, are more representative of many real-world scenarios such as agent-based modelling, social dilemmas, and interacting self-interested agents like autonomous vehicles.

In these settings, methods developed for the zero-sum setting often lead to catastrophic outcomes. For instance, in the Iterated Prisoner's Dilemma (IPD) (Axelrod and Hamilton, 1981, IPD), agents that treat their opponents as static usually end up with the globally worst outcome - unconditional mutual defection. Opponent Shaping (OS) methods like Learning with Opponent Learning Awareness (Foerster et al., 2018, LOLA) were introduced to mitigate such disastrous results, by considering the opponent's learning step to shape their policy. This was effective in a self-play setting, leading to the discovery of the reciprocating tit-for-tat (TFT) strategy in IPD (Foerster et al., 2018).

However, OS algorithms often assume that the opponent is a naive learning (NL) agent. An NL agent neglects the non-stationary environment and ignores the updates of opponents in their own update. It is often not the case that the opponent is an NL agent, especially in self-play, i.e. when two LOLA agents play against each other (Foerster et al., 2018; Letcher et al., 2018; Kim et al., 2021). Moreover, these methods rely on second-order derivatives, which are typically high-variance and result in unstable learning. They are also myopic, focusing only on the opponent's immediate future learning steps rather than their long-term development (Lu et al., 2022a).

Previous work, Model-free Opponent Shaping (Lu et al., 2022a, M-FOS), solves the above challenges. M-FOS introduces a *meta-game* structure, each *meta-step* representing an episode of the embedded "inner" game. The *meta-state* consists of "inner" policies, and the *meta-policy* generates an inner policy at each *meta-step*. M-FOS uses model-free optimisation techniques to train the meta-policy, eliminating the need for higher-order derivatives to accomplish long-horizon opponent shaping. The M-FOS framework has shown promising long-term shaping results in social-dilemma games (Lu et al., 2022a; Khan et al.)

For simpler, low-dimensional games, M-FOS learns policy updates directly by taking policies as input and outputting the next policy as an action. Inputting and outputting entire policies does not extend well to more complex, higher-dimensional games, e.g. when policies are represented as neural networks. The original M-FOS paper proposes a variant which uses trajectories as inputs instead of exact policy representations. In this work we derive the sample complexity for both cases.

Whereas some previous OS algorithms enjoy strong theoretical foundations thanks to the Differentiable Games framework (Balduzzi et al., 2018), the M-FOS framework has not been investigated theoretically. In this work, we derive the sample complexity of the M-FOS algorithm. Understanding the sample complexity of an algorithm is helpful in many ways, such as evaluating its efficiency, providing a performance metric for comparisons of similar methods, assisting in resource management, predicting the learning time or even guiding the discovery of new algorithms.

At a high level, we adapt the R_{MAX} algorithm (Brafman and Tennenholtz, 2001) to the M-FOS framework. R_{MAX} is a model-based reinforcement learning (MBRL) algorithm originally devised to analyse the sample complexity in zero-sum games and typically used in single-player settings. This lead to the following contributions:

1. We present the PAC upper-bound sample complexity for both cases in M-FOS in Section 4.
2. We verify the sample complexity *empirically* in Section 6 by implementing M-FOS with a tabular RL algorithm R_{MAX} as the meta-agent.[1]
3. We verify the sample complexity of the R_{MAX} algorithm empirically Section 6.

2 Related Work

Theoretical Analysis of Differentiable Games: Much past work assumes that the game being optimised is differentiable (Balduzzi et al., 2018). This assumption enables far easier theoretical analysis because one can directly use end-to-end gradient-based methods rather than reinforcement learning in those settings. Several works in this area investigate the convergence properties of various algorithms Letcher (2020); Schäfer and Anandkumar (2019); Balduzzi et al. (2018).

Opponent Shaping: More closely related to our work are methods that specifically analyse OS. SOS (Letcher et al., 2018) and COLA (Willi et al., 2022) both analyse opponent-shaping methods that operate in the differentiable games framework. These works provide theoretical *convergence* analysis for opponent-shaping algorithms; however, neither work analyzes sample complexity. POLA (Zhao et al., 2022) theoretically analyses an OS method that is invariant to policy parameterization. M-FOS does not operate in the differentiable games framework. While this enables M-FOS to scale to more challenging environments, such as Coin Game (Lu et al., 2022a), it comes at the cost of convenient theoretical analysis. Khan et al. empirically scales M-FOS to more challenging environments with larger state spaces, while Lu et al. (2022b) empirically investigates applying M-FOS to a state-based adversary. To the best of our knowledge, our work is the first to theoretically analyse OS outside of the differentiable games framework. Furthermore, our work is the first to analyse the sample complexity of an OS method.

Theoretical Analysis of Sample Complexity in RL: There are several works that use the R_{MAX} (Brafman and Tennenholtz, 2001) framework to derive the sample complexity of RL algorithms across a variety of settings. Closely related to our work is Zhang et al. (2022), which uses the R_{MAX} to derive sample complexity bounds for learning in fully-cooperative multi-agent RL.

Our work is also related to methods that analyse sample complexity on continuous-space RL. Analyzing the sample complexity of algorithms in continuous-space RL is particularly challenging because there are an infinite number of potential states. To address this, numerous techniques have been suggested that each make specific assumptions.

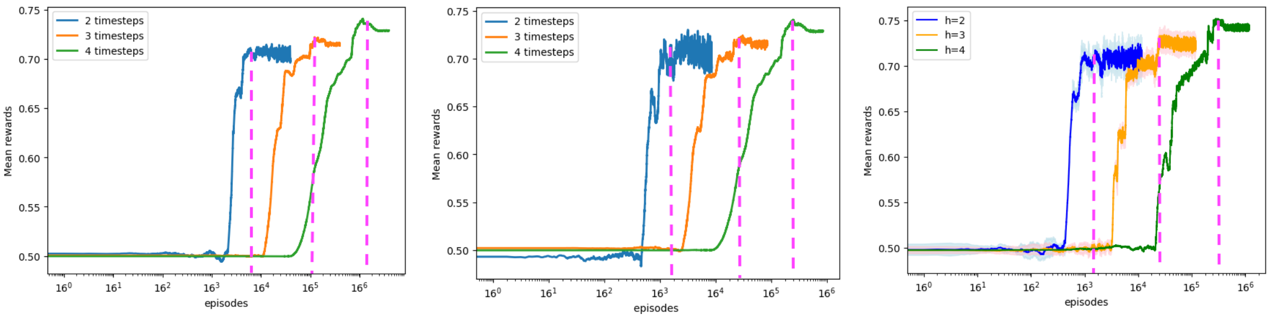
Liu and Brunskill (2018) assumes a **stationary asymptotic occupancy distribution** under a random walk in the MDP. Malik et al. (2021) uses an effective planning window to handle MDPs with non-linear transitions. However, neither of these assumptions applies to M-FOS.

Instead, this work focuses on **discretising** the continuous space and expresses the complexity bounds in terms of the discretisation grid size. This is related to the concept of *state aggregation* (Singh et al., 1994; Boutilier et al., 1999), which groups states into clusters and treats the clusters as the states of a new MDP. These previous works only formulated the aggregation setting in MDPs and did not provide theoretical or empirical sample complexity proofs.

Furthermore, prior studies on *PAC-MDP* did not empirically verify the connection between the sample complexity and size of the state space. In this work, we **empirically verify the relationship between the sample complexity and the cardinality of the inner-state space** in the Matching Pennies game.

[1] The project code is available on https://github.com/rmaxmfos/rmaxmfos

[3] For computational reasons, we display 5 seeds only for the $\varepsilon = 0.8$, $h = [2, 3]$ setting.



$\varepsilon = 0.2$ $\varepsilon = 0.5$ $\varepsilon = 0.8$

Figure 1. Mean reward of the M-FOS agent against meta-episodes on a log (base 16) scale for different ε [3]

Algorithm 1 The Adapted M-FOS Algorithm with R_{max} as the meta-agent

Meta-game Inputs: $\hat{\mathcal{S}}, \hat{\mathcal{A}}, \hat{\gamma}, m, \varepsilon, J$
Inner-game Inputs: $\mathcal{S}, \mathcal{A}, \gamma, k$
Initialisation: $\hat{Q}(\hat{s}, \hat{a}) \leftarrow 0,\ \hat{r}(\hat{s}, \hat{a}) \leftarrow 0,\ n(\hat{s}, \hat{a}) \leftarrow 0,\ n(\hat{s}, \hat{a}, \hat{s}') \leftarrow 0$

```
for meta-episode $n = 0, 1, .., J$ do
    Reset environment
    for meta-time step $t = 1, 2, ..., K$ do
        Sample $a^{-i} = \varepsilon\text{-}greedy(\phi_{\mathbf{t}}^{-\mathbf{i}})$
        Run inner game of length $l$
        $\hat{a}_t = \phi_t^i$
        $\hat{r}_t = r^i + \hat{\gamma}\hat{r}_{t-1}$
        Let $\hat{s}'_t$ be the next meta-state after executing meta-action $\hat{a}_t$ from meta-state $\hat{s}_t$
        $\hat{s}'_t = [\boldsymbol{\phi}_{\boldsymbol{n}}^{-\boldsymbol{i}}, \phi_t^i]$
        if $n(\hat{s}, \hat{a}) < m$ then
            $\hat{r}(\hat{s}_t, \hat{a}_t) \leftarrow \hat{r}(\hat{s}_t, \hat{a}_t) + R_t^i$
            $n(\hat{s}_t, \hat{a}_t) \leftarrow n(\hat{s}_t, \hat{a}_t) + 1$
            $n(\hat{s}_t, \hat{a}_t, \hat{s'}_t) \leftarrow n(\hat{s}_t, \hat{a}_t, \hat{s}'_t) + 1$
            if $n(\hat{s}_t, \hat{a}_t) = m$ then
                for $i = 1, 2, 3, \cdots, \lceil \frac{ln(\frac{1}{\varepsilon(1-\gamma)})}{1-\gamma} \rceil$ do
                    for all $(\hat{s}, \hat{a})$ do
                        if $n(\hat{s}, \hat{a}) \geq m$ then
                            $\hat{Q}(\hat{s}, \hat{a}) \leftarrow \hat{R}(\hat{s}, \hat{a}) + \hat{\gamma} \sum_{s'} \hat{T}(\hat{s}'|\hat{s}, \hat{a}) \max_{\hat{a}'} \hat{Q}(\hat{s}', \hat{a}')$
        $\hat{s} \leftarrow \hat{s}'$
```

3 Background

3.1 Stochastic Game

A stochastic game (SG) is given by a tuple $G = \langle \mathcal{I}, \mathcal{S}, \boldsymbol{\mathcal{A}}, T, \boldsymbol{R}, \gamma \rangle$. $\mathcal{I} = \{1, \cdots, n\}$ is the set of agents, $\mathcal{S}$ is the state space, $\boldsymbol{\mathcal{A}}$ is the cross-product of the action space for each agent such that the joint action space $\boldsymbol{\mathcal{A}} = \mathcal{A}^1 \times \cdots \times \mathcal{A}^n$, $T : \mathcal{S} \times \boldsymbol{\mathcal{A}} \mapsto \mathcal{S}$ is the transition function, $\boldsymbol{R}$ is the cross-product of reward functions for all agents such that the joint reward space $\boldsymbol{R} = R^1 \times \cdots \times R^n$, and $\gamma \in [0, 1)$ is the discount factor.

In an SG, agents simultaneously choose an action according to their stochastic policy at each timestep t, $a_t^i \sim \pi_{\phi^i}^i(\cdot|s_t^i)$. The joint action at timestep t is $\boldsymbol{a_t} = \{a_t^i, \boldsymbol{a_t^{-i}}\}$, where the superscript $\boldsymbol{-i}$ indicates all agents except agent i and ϕ^i is the policy parameter of agent i. The agents then receive reward $r_t^i = R^i(s_t, \boldsymbol{a_t})$ and observe the next state $s_{t+1} \sim T(\cdot|s_t, \boldsymbol{a_t})$, resulting in a trajectory $\tau^i = (s_0, \boldsymbol{a_0}, r_0^i, ..., s_T, \boldsymbol{a_T}, r_T^i)$, where T is the episode length.

3.2 Markov Decision Process

A Markov decision process (MDP) is a special case of stochastic game and can be described as $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, T, R, \gamma \rangle$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $T\left(s_{t+1} \mid s_t, a_t\right)$ is the transition function, $R\left(s_t, a_t\right)$ is the reward function, and γ is the discount factor. At each timestep t, the agent takes an action $a_t \in \mathcal{A}$ from a state $s_t \in \mathcal{S}$ and moves to a next state $s_{t+1} \sim T\left(\cdot \mid s_t, a_t\right)$. Then, the agent receives a reward $r_t = R\left(s_t, a_t\right)$.

3.3 Model-Free Opponent Shaping

Model-free Opponent Shaping (M-FOS) frames the OS problem as a meta-reinforcement-learning problem, in which the opponent shaper plays a meta-game. The meta-game is a partially-observable MDP, in which the meta-agent controls the inner agent in the inner game.

The inner game is the actual environment that our agents are playing, which is an SG. In a meta-game at timestep t, the M-FOS agent is at the meta-state $\hat{s}_t = [\phi_{t-1}^i, \boldsymbol{\phi_{t-1}^{-i}}]$, which contains all agents' policy parameters for the underlying SG. Alternatively, $\hat{s}_t = \boldsymbol{\tau}$ in cases where the trajectories represent the policies sufficiently.

The meta-agent takes a meta-action $\hat{a}_t = \phi_t^i \sim \pi_\theta(\cdot|\hat{o}_t)$, which is the M-FOS' inner agent's policy parameter. The action is chosen from the meta-policy π parameterized by parameter θ. The M-FOS agent receives reward $\hat{r}_t = \sum_{k=0}^{K} r_k^i(\phi_t^i, \boldsymbol{\phi_t^{-i}})$, where K is the number of inner episodes. A new meta-state is sampled from a stochastic transition function $\hat{s}_{t+1} \sim T(\cdot|\hat{s}_t, \hat{a}_t)$.

4 Sample Complexity Analysis with R_{MAX} as Meta-Agent

R_{MAX} (Brafman and Tennenholtz, 2001) is an MBRL algorithm for learning in tabular MDPs. In this work, we adapt the original M-FOS algorithm such that it uses R_{MAX} as the meta-agent (see Algorithm 1). We refer to this algorithm as the *Adapted M-FOS* from here on. We learn the transition model $\hat{T}_m$ and the reward model $\hat{R}_m$ using the empirical maximum likelihood estimator. Using the learned models, we construct an approximate m-known MDP $\hat{M}_m$ (see Definition A.14). Within $\hat{M}_m$, we evaluate the inner-game policy outputted by the meta-policy using episodic rollouts. The evaluation is then used to update the meta-policy according to the M-FOS algorithm. Our adapted M-FOS algorithm optimistically assigns rewards for all under-visited discretised (meta-state, meta-action) pairs to encourage exploration.

Definition 4.1 (m-Known MDP)**.** M_m is the expected version of $\hat{M}_m$ where:

$$T_m\left(\hat{s}' \mid \hat{s}, \hat{a}\right) := \begin{cases} T\left(\hat{s}' \mid \hat{s}, \hat{a}\right) & \text{if } (\hat{s}, \hat{a}) \in \text{m-known} \\ 1\left[\hat{s}' = \hat{s}\right] & \text{otherwise} \end{cases}$$

$$\hat{T}_m\left(\hat{s}' \mid \hat{s}, \hat{a}\right) := \begin{cases} \frac{n\left(\hat{s}, \hat{a}, \hat{s}'\right)}{n(\hat{s}, \hat{a})}, & \text{if } (\hat{s}, \hat{a}) \in \text{m-known} \\ 1\left[\hat{s}' = \hat{s}\right], & \text{otherwise} \end{cases}$$

$$R_m\left(\hat{s}, \hat{a}\right) := \begin{cases} R\left(\hat{s}, \hat{a}\right), & \text{if } (\hat{s}, \hat{a}) \in \text{m-known} \\ R_{\text{max}} & \text{otherwise} \end{cases}$$

$$\hat{R}_m(\hat{s}, \hat{a}) = \begin{cases} \frac{\sum_i^{n(\hat{s}, \hat{a})} r(\hat{s}, \hat{a})}{n(\hat{s}, \hat{a})}, & \text{if } (\hat{s}, \hat{a}) \in \text{m-known} \\ R_{\max}, & \text{otherwise} \end{cases}$$

We provide theory results for two cases of M-FOS' meta-agent as proposed by the original paper (Lu et al., 2022a). *Case I* uses all agents' policy parameters from the previous timestep as the meta-state. Instead, *Case II* uses the inner-game trajectory as the meta-state. In both cases, the meta-agent takes the inner agent's policy parameters as the meta-action. The detailed proofs are provided in the appendix, and they are heavily based on results from Brafman and Tennenholtz (2001), Strehl et al. (2009), Jiang (2020) and Kakade (2003).

4.1 *Case I*: Policy Parameters as the Meta-State

In *Case I*, the meta-state $\hat{s}_t$ is all inner agents' policies from the previous timestep. Formally, $\hat{s}_t := \boldsymbol{\phi}_{t-1} = [\boldsymbol{\phi}_{t-1}^{-i}, \phi_{t-1}^i]$. The meta-action $\hat{a}_t$ is MFOS' inner agent's current policy parameters ϕ_t^i. The policies are computed from the Q-tables in the inner game with Boltzmann sampling. The meta-reward is the average over K inner game rollouts' discounted episodic rewards.

4.1.1 Discretisation with ε-nets

To use R_{MAX} as the meta-agent, we first convert the meta-problem in M-FOS into a tabular setting. We discretise the meta-game's continuous meta-state space $\hat{S}$ and meta-action space $\hat{A}$ into discrete and finite spaces $\hat{S}_d$ and $\hat{A}_d$ respectively using ε-nets. We describe the discretisation process for $\hat{A}$ below. The process for $\hat{S}$ can be derived similarly (see Appendix A.3).

For the meta-action space $\hat{A}$ and a chosen discretisation error $\alpha > 0$, we obtain a finite set of points $\hat{A}_d \subset \hat{A}$ such that for all $\hat{a} \in \hat{A}$, there exist $\hat{a}_d \in \hat{A}_d$ where

$$\|\hat{a} - \hat{a}_d\| \leq \alpha. \tag{1}$$

To obtain a finite set of $\hat{A}_d$, the values in A must be bounded. This is satisfied because we follow the original R_{MAX} algorithm to assume a bounded inner game reward of $0 < r_t < R_{\text{MAX}}$. In our case, we set R_{MAX} arbitrarily as 1. Therefore, with an inner game discount factor of γ, each of the $|S| \times |A|$ entries in $\hat{a}$ is bounded between 0 and $\frac{1}{1-\gamma}$, i.e. $\hat{A} = \{\hat{a} \in \mathbb{R}^{|S| \times |A|} : ||\hat{a}|| \leq \frac{1}{1-\gamma}\}$.

To find the ε-net of $\hat{A}$, we divide the $|S| \times |A|$-dimensional ball into grids of equal size λ, meaning that the discretisation error $\alpha = \frac{\lambda\sqrt{|S||A|}}{2}$. This leads to the size of discretized meta-action space upper bounded by

$$|\hat{A}_d| \leq \left(\frac{\frac{2\sqrt{|S||A|}}{1-\gamma}}{\alpha}\right)^{|S||A|}. \tag{2}$$

The R_{MAX} meta-agent learns in the discretised meta-state and meta-action space. Our final sample complexity bound in Theorem 4.3 takes into account the discretisation error α as well.

4.1.2 Sample Complexity Bound

To study the sample complexity of the Adapted M-FOS, we define the value of a policy as follows:

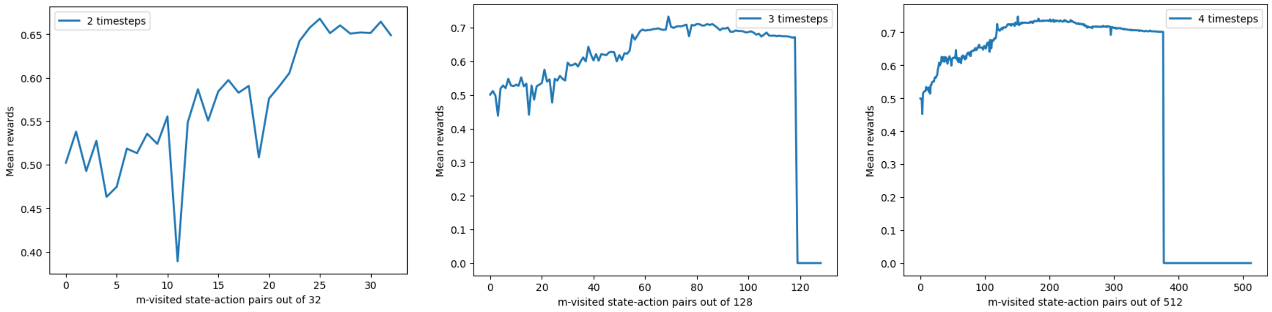


Figure 2. Mean reward against the number of m-visited state-action pairs, for $\varepsilon = 0.5$ and different h values

Definition 4.2. Given a meta-game policy π, we estimate its value $J(\pi)$ in an MDP M with initial state distribution d_0, defined as the expected return obtained by following the policy in M,

$$J_M(\pi) = \mathbb{E}_{s \sim d_0}\left[V^{\pi}(s)\right]. \tag{3}$$

where the value function is the expected reward for following policy π in the state s such that $V_\pi(s_t) = E_\pi[R_{t+1} + \gamma v_\pi(s_{t+1})|s_t]$

Theorem 4.3. *For $\varepsilon_d \in \{0,1\}$, $\delta \in \{0,1\}$, $\lambda \in \mathbb{R}^+$, let $M = \langle \hat{\mathcal{S}}, \hat{\mathcal{A}}, T, R, \rangle$ be the meta-game MDP, M_d be the discretised version of M with λ as the discretization radius, and $M_{d,m}$ be the m-Known, discretized MDP. Let G= $\langle \mathcal{I}, \mathcal{S}, \boldsymbol{\mathcal{A}}, T_{inner}, \boldsymbol{R}_{inner}, \gamma\rangle$ be a finite horizon SG as the inner-game, then there exist some constants $C > 0$ and inputs $m(\frac{1}{\varepsilon}, \frac{1}{\delta}, \frac{1}{\lambda}) = \mathcal{O}(\frac{|\hat{\mathcal{S}}| + \ln\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta}}{\varepsilon_d^2(1-\hat{\gamma})^4})$ such that if R_{MAX} is executed on G, the following holds:*
Let $\pi^*_{M_d}$ be a R_{MAX} policy of the M-FOS agent in Case I, with probability at least $1-\delta$, $J_{M_d}(\pi^*_{M_d}) - J_{M_d}(\pi^*_{\hat{M}_{d,m}}) \leq 2\varepsilon_d$ is true for all but

$$\mathcal{O}(\frac{(\frac{1}{\lambda(1-\gamma)})^{(2n+1)|\mathcal{S}||\mathcal{A}|}}{\varepsilon_d^3}\frac{1}{(1-\gamma)^5}\ln\frac{1}{\delta}) \tag{4}$$

episodes.

Theorem 4.3 shows that Adapted M-FOS algorithm acts near-optimally on all but an exponential number of steps.

4.2 *Case II*: Trajectory History as the Meta-State

In *Case II*, the meta-state $\hat{s}_t$ at timestep t is all inner agents' trajectories. Formally, $\hat{s}_t := \boldsymbol{\tau}_t$. The meta-action $\hat{a}$ is our inner agent's discretised policy parameterised by ϕ^i_t. The discretised policy is the discretised Q-table of that agent obtained in the inner game. The meta-reward is the discounted episodic rewards averaged over K inner game rollouts. Detailed proofs are available in Appendix A.6.

Theorem 4.4. *For $\varepsilon_d \in \{0,1\}$, $\delta \in \{0,1\}$, $\lambda \in \mathbb{R}^+$, let $M = \langle \hat{\mathcal{S}}, \hat{\mathcal{A}}, T, R, \rangle$ be the meta-game MDP, M_d be the discretised version of M with λ as the discretization radius, and $M_{d,m}$ be the m-Known, discretized MDP. Let $G = \langle \mathcal{I}, \mathcal{S}, \boldsymbol{\mathcal{A}}, T_{inner}, \boldsymbol{R}_{inner}, \gamma\rangle$ be a h-step SG as the inner-game, then there exists some constants $C > 0$, $C_1 > 0$ and inputs $m(\frac{1}{\varepsilon}, \frac{1}{\delta}, \frac{1}{\lambda}) \geq \mathcal{O}(\frac{|\hat{\mathcal{S}}| + \ln\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta}}{\varepsilon_d^2(1-\hat{\gamma})^4})$ such that if R_{MAX} is executed on G, the following holds:*
Let $\pi^*_{M_K}$ be a R_{MAX} policy of the M-FOS agent in Case II, with probability at least $1-\delta$, $J_{M_d}(\pi^*_{M_d}) - J_{M_d}(\pi^*_{\hat{M}_{d,m}}) \leq 2\varepsilon_d$ is true for all but

$$\mathcal{O}(\frac{(|\mathcal{S}||\mathcal{A}|)^{2nh}(\frac{1}{\lambda(1-\gamma)})^{2nh+|\mathcal{S}||\mathcal{A}|}}{\varepsilon_d^3}\frac{1}{(1-\gamma)^5}\ln\frac{1}{\delta}) \tag{5}$$

episodes.

In Section 5, we show empirically that the number of samples needed indeed scales by a factor of $|S||A|^{2nh}$, as seen in Theorem 4.4.

5 Experimental Setup

Matching Pennies (MP): is a two-player, single-shot, zero-sum game with a payoff matrix shown in Table 1, where agents either pick Heads (H) or Tails (T), $a^i \in \{H, T\}$ and $a^i \sim \pi_{\phi^i}(\cdot \mid \{\})$, where ϕ^i correspond to the probability of picking H of agent i. Note that in this work the game is not iterated, meaning that one episode has a length of 1 and the episodic return corresponds to the payoff after one interaction $J = r = P(a^1, a^2)$. For M-FOS, this means that a meta-step corresponds to one iteration of the Matching Pennies game. The meta-return corresponds to the discounted, cumulative meta-reward after playing the Matching Pennies K times. While the original M-FOS was evaluated on a more complex, iterated version of the Matching Pennies game, this simple setting with a binary action space is sufficient for our empirical validation and more practical, because the R_{MAX} algorithm memory usage grows exponentially with the size of the state and action space.

Table 1. Payoff Matrix for MP

Player 1\Player 2	Head	Tail
Head	(+1, -1)	(-1, +1)
Tail	(-1, +1)	(+1, -1)

For our experiments, we focus on empirically validating only *Case II*, because the R_{MAX} algorithm memory usage grows exponentially with the size of the state and action space, so *Case I*, i.e., outputting whole Q-tables, becomes computationally intractable even for the simplest settings. For *Case II*, we adapt the M-FOS algorithm to use smaller meta-state and meta-action spaces to ensure tractability. For example, if the meta-episode corresponds to repeatedly playing the single-shot MP, then the (partial) meta-trajectory (and correct input for M-FOS) corresponds to $\hat{s}_t = \hat{\tau}_t = (\boldsymbol{a_0}, ..., \boldsymbol{a_{t-1}})$, where t is the current meta-step. However, our meta-state corresponds only to a fixed-length window of the past actions taken in $\hat{s}_t = (a^1_{t-h}, a^2_{t-h}, ..., a^1_{t-1}, a^2_{t-1})$, where h is the window size, instead of all past actions taken in the meta-episode. The window size allows us to control the size of the meta-game state, i.e., $\hat{\mathcal{S}} \in \mathbb{R}^{2^h}$. The opponent is a standard Q-learning agent that updates the Q-values at every step and selects actions with an ϵ-greedy strategy. The pseudo-code is provided in Appendix B.1.

We next outline the variables we analysed in our experiments to verify the sample complexity terms. As we aim to find the scaling law of the M-FOS algorithm, we vary the window-size h to understand how the sample complexity change accordingly. We expect that the empirical sample complexity reacts more strongly to changes in the meta-game state space size than other parameters like the discount factor γ.

The inner-game state and action space size is $|\mathcal{S}||\mathcal{A}| = |\mathcal{G}| = 2$ and the number of agents is $n = 2$ in the MP. The discount factor is set to $\gamma = 0.8$. Practically, $\gamma = 0.99$ is more commonly used to encourage long-term planning and improve stability. However, this would incur a very large sample complexity because of the term $\frac{1}{(1-\gamma)^5}$ in Equation 5.

We vary the length of the window h in the MP game to change the size of the meta-state space with $h \in [2, 3, 4]$. We also ablate the discretisation error ε. Ultimately, we are interested in analyzing the minimum number of episodes required to converge to an ε-optimal policy under these different settings.

We now discuss the expected sample complexity bounds in the MP. Given the setup we defined above, we find that:

$$|\hat{\mathcal{S}}| = |\boldsymbol{\tau}_h| = |\boldsymbol{sa}|^h = (|\mathcal{S}||\mathcal{A}|)^{nh} = (|\mathcal{G}|)^{nh}$$
$$|\hat{\mathcal{A}}| = |a| = 1$$

In the MP game, we have $|\mathcal{G}| = |\mathcal{S}| \times |\mathcal{A}| = 2$. Following the expression in Section 4 (Equation (20)),

$$\begin{aligned}\mathcal{O}(\frac{|\hat{\mathcal{S}}|^2|\hat{\mathcal{A}}|}{\varepsilon_d^3}\frac{1}{(1-\gamma)^5}\ln\frac{1}{\delta}) &\sim \mathcal{O}(\frac{|\mathcal{G}|^{2nh}}{\varepsilon^3(1-\gamma)^5}\ln\frac{1}{\delta})\\ &\sim \mathcal{O}(\frac{16^h}{\varepsilon_d^3(1-\gamma)^5}\ln\frac{1}{\delta})\end{aligned} \tag{6}$$

Thus, we hypothesize that there exists an exponential relationship between the window size h and the sample complexity, whereby the sample complexity increases exponentially $\sim \mathcal{O}(16^h)$.

6 Results & Discussion

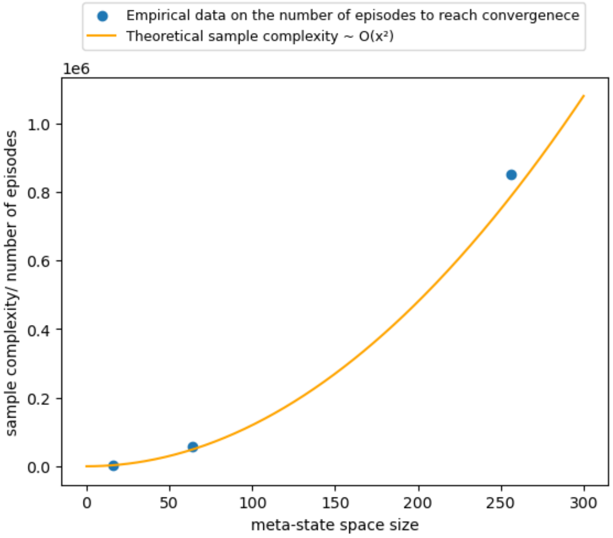
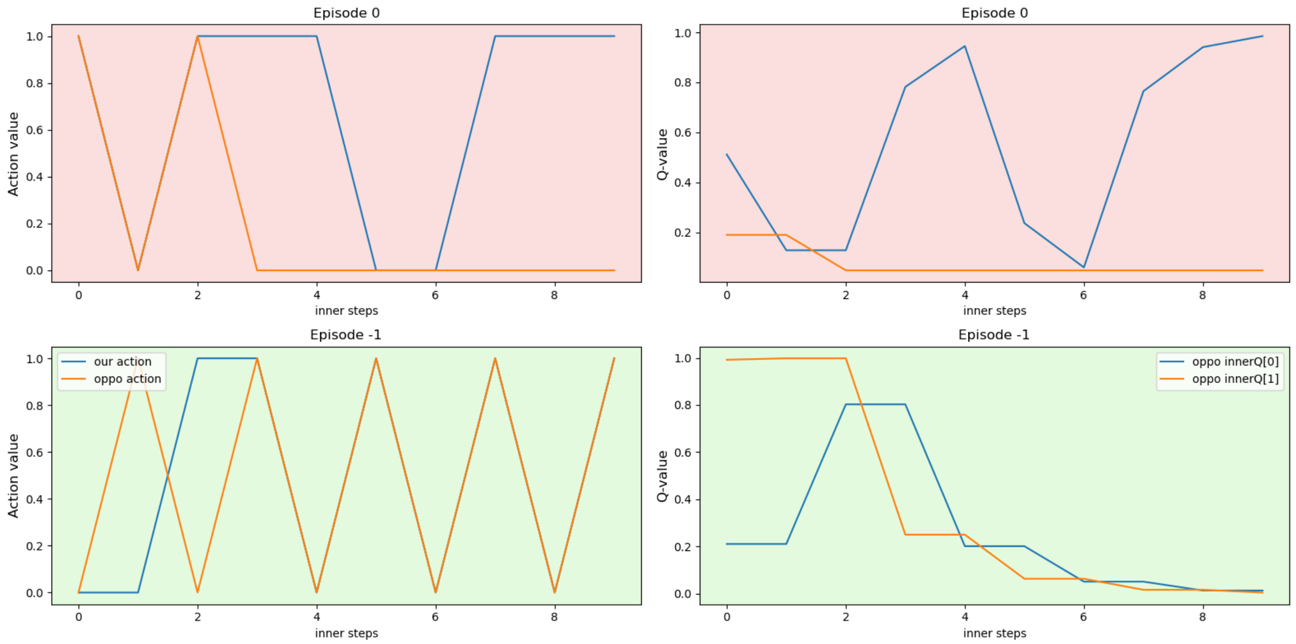


Figure 3. Empirical and theoretical sample complexity against meta-state space size for $\varepsilon = 0.8$

Figure 1 shows the reward curve across the meta episodes on a log scale for different ε. Every graph contains three mean reward curves for meta-trajectory length $h = [2, 3, 4]$. For each setting, we set $\hat{K} = 10 \times m \times |\hat{\mathcal{S}}|$ so that there are enough meta-episodes for the agent's performance to converge. The constant of 10 is arbitrary and can be set to any number.

Varying inner-state space size: In Figure 1, for $\varepsilon = 0.8$ and $h = [2, 3, 4]$, it takes approximately $[16^3, 16^4, 16^5]$ episodes to converge to $[0.702, 0.709, 0.724]$ respectively. In Equation (6), we proved that the sample complexity



(a) Inner game action

(b) Opponent's inner-Q values

Figure 4. Action value and opponent's inner Q table of the first meta-step of the first episode and last meta-step of the last meta-episode

grows by a factor of 16 when the trajectory length increases by 1. The empirical result confirms the exponential relationship between the sample complexity and inner-state space size. As shown in Table 2, as the meta-state space size increases, the converged reward and mean reward are higher. This is because the agent observes more past actions, allowing it to express more complex strategies given the history. However, note that the meta-agent can always find the optimal strategy given the history.

Table 2. Normalised mean reward of the M-FOS agent vs. opponent in different ε and trajectory length h

	trajectory length h		
ε	2 steps	3 steps	4 steps
0.2	[0.648, 0.352]	[0.696, 0.304]	[0.721, 0.279]
0.5	[0.645, 0.355]	[0.679, 0.321]	[0.709, 0.291]
0.8	[0.639, 0.361]	[0.660, 0.340]	[0.681, 0.319]

Figure 2 shows the average reward against the number of m-visited state-action pairs for $h = [2, 3, 4]$ and $\varepsilon = [0.2, 0.5, 0.8]$. The mean reward is set to 0 for state-action pairs that have not been visited at least m times. As the trajectory length increases, the percentage of m-visited state-action pairs decreases. As trajectory length increases, the meta-state space size increases. And since $m \propto |\mathcal{S}|$, m increases. Thus it is more difficult to visit every state-action pair in a larger state space for higher m times. Note that the algorithm still converges since R_{MAX} does not require all state-action pairs to be m-visited.

Varying ε: As ε increases, m decreases since $m \propto \frac{1}{\varepsilon^2}$. It takes fewer episodes to visit a state-action pair for m times, which leads to a less accurate maximum-likelihood estimate of the reward and transition function because of the smaller data size. This corresponds to a higher error tolerance level of ε. The mean reward increases from 0.5 when none of the state-action pairs is m-visited to a converged value of around 0.7. Note that when the number of m-visited state-action pairs is small, the variance of the mean reward is higher than when the number is large since the agent is still actively exploring.

Figure 3 shows the theoretical sample complexity curve and empirical data from running the M-FOS algorithm in the MP game. The empirical results support our theoretical proof of sample complexity. It is important to note that the computation is limited by the relationship between sample complexity and the size of the meta-state space. As the meta-state space size increases, the memory required to obtain each additional data point grows exponentially. Because of this, we only evaluated at three meta-state sizes.

6.1 Interpretation of the meta-policy

Figure 4 shows the action, cumulative reward and opponent's Q-table in the first and last meta-episode to compare the performance before and after learning. The M-FOS agent's goal is to match the opponent's action while the opponent's goal is to avoid matching. From Figure 4(a), initially the agents do not have sufficient information about the environment and act randomly in the first meta-episode. In the last meta-episode, 8 out of 10 action match and the M-FOS agent gets a cumulative reward of 8. The M-FOS agent exploits the opponent by recognising the pattern that if the opponent takes action a and suffers from a loss, the opponent tends to take the other action $\neg a$ that may have a higher value. This is shown in Figure 4(b), where the blue line indicates the opponent's Q-value for taking action 0 and the orange line indicates that for taking action 1. The opponent switches action if it does not get a reward, which leads to fluctuating action values in consecutive time steps.

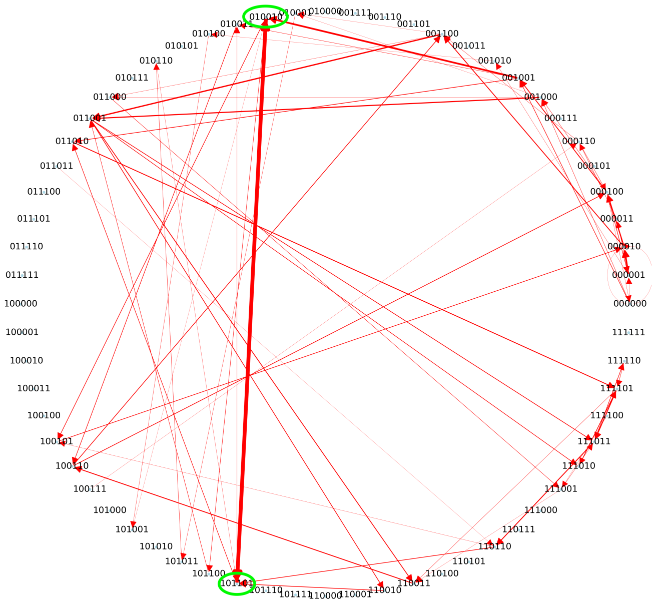


Figure 5. Transition diagram for $h = 3$, $\varepsilon = 0.8$

Figure 5 shows a transition diagram for the MP game with $h = 3$, $\varepsilon = 0.8$. The weight of the edge represents the visitation frequency such that wider red lines indicate more frequent transitions between those states. The nodes represent the past trajectory of both agents for 3 timesteps, i.e. $[a^{-i}_{t-3}, a^{-i}_{t-2}, a^{-i}_{t-1}, a^{i}_{t-3}, a^{i}_{t-2}, a^{i}_{t-1}]$. The most visited nodes are $[0, 1, 0, 0, 1, 0]$ and $[1, 0, 1, 1, 0, 1]$ (indicated by green circles). These two nodes have frequent transitions from and to each other. As an instance, in the previous step, the meta-state is $[0, 1, 0, 0, 1, 0]$, where the opponent acted 0 and our agent acted 0. The opponent lost in the previous step since the actions matched, thus it acted 1 the next round since it has a higher Q-value. Our M-FOS agent counteracted with an action of 1. This leads to the next meta-state $[1, 0, 1, 1, 0, 1]$. Note that the trajectories are alternations of 1s and 0s, which further confirms the M-FOS agent's ability to shape the opponent's policy.

7 Conclusion & Future Work

This work is the first to explore both theoretical and empirical sample complexity of a meta-RL algorithm and we prove that the sample complexity of M-FOS grows **exponentially** as the meta-state and discretisation grid size increase. We prove there exists an **exponential relationship** between the sample complexity and the number of agents. Moreover, we presented the *theoretical* sample complexity of the two cases in M-FOS in Section 4 and find the scaling rule in terms of the state and action space size of the inner game. We then implement M-FOS with a tabular RL algorithm R_{max} as the meta-agent in Section 5 and verify the sample complexity *empirically*. Finally, we are the first to verify the sample complexity of the R_{MAX} algorithm empirically.

There are many avenues for future work such as using a different meta-agent, trying to generalize the assumptions, investigating other model-free optimisation techniques or empirically evaluating more complex games.

References

Robert Axelrod and William D. Hamilton. The evolution of cooperation. *Science*, 211(4489):1390–1396, 1981.

David Balduzzi, Sebastien Racaniere, James Martens, Jakob Foerster, Karl Tuyls, and Thore Graepel. The mechanics of n-player differentiable games. In *International Conference on Machine Learning*, pages 354–363. PMLR, 2018.

Craig Boutilier, Thomas Dean, and Steve Hanks. Decision-theoretic planning: Structural assumptions and computational leverage. *J. Artif. Int. Res.*, 11(1):194, jul 1999. ISSN 1076-9757.

Ronen Brafman and Moshe Tennenholtz. R-max - a general polynomial time algorithm for near-optimal reinforcement learning. volume 3, pages 953–958, 01 2001. doi: 10.1162/153244303765208377.

Benjamin Ellis, Skander Moalla, Mikayel Samvelyan, Mingfei Sun, Anuj Mahajan, Jakob N. Foerster, and Shimon Whiteson. Smacv2: An improved benchmark for cooperative multi-agent reinforcement learning, 2022. URL `https://arxiv.org/abs/2212.07489`.

Murat A. Erdogdu. Covering with epsilon-nets, 2022. URL `https://erdogdu.github.io/csc2532/lectures/lecture05.pdf`.

Jakob N. Foerster, Richard Y. Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. Learning with opponent-learning awareness, 2018.

Nan Jiang. Notes on rmax exploration, 2020. URL `https://nanjiang.cs.illinois.edu/files/cs598/note7.pdf`.

Sham Kakade. *On the sample complexity of Reinforcement Learning*. PhD thesis, University of London, 2003.

Akbir Khan, Newton Kwan, Timon Willi, Chris Lu, Andrea Tacchetti, and Jakob Nicolaus Foerster. Context and history aware other-shaping.

Dong-Ki Kim, Miao Liu, Matthew D Riemer, Chuangchuang Sun, Marwa Abdulhai, Golnaz Habibi, Sebastian Lopez-Cot, Gerald Tesauro, and JONATHAN P HOW. A policy gradient algorithm for learning to learn in multiagent reinforcement learning, 2021. URL `https://openreview.net/forum?id=zdrls6LIX4W`.

Alistair Letcher. On the impossibility of global convergence in multi-loss optimization. *arXiv preprint arXiv:2005.12649*, 2020.

Alistair Letcher, Jakob Foerster, David Balduzzi, Tim Rocktäschel, and Shimon Whiteson. Stable opponent shaping in differentiable games. *arXiv preprint arXiv:1811.08469*, 2018.

Yao Liu and Emma Brunskill. When simple exploration is sample efficient: Identifying sufficient conditions for random exploration to yield pac rl algorithms, 05 2018.

Chris Lu, Timon Willi, Christian A. Schroeder de Witt, and Jakob N. Foerster. Model-free opponent shaping. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 14398–14411. PMLR, 2022a.

Chris Lu, Timon Willi, Alistair Letcher, and Jakob Foerster. Adversarial cheap talk. *arXiv preprint arXiv:2211.11030*, 2022b.

Dhruv Malik, Aldo Pacchiano, Vishwak Srinivasan, and Yuanzhi Li. Sample efficient reinforcement learning in continuous state spaces: A perspective beyond linearity. In *International Conference on Machine Learning*, 2021.

Florian Schäfer and Anima Anandkumar. Competitive gradient descent. *Advances in Neural Information Processing Systems*, 32, 2019.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, Yutian Chen, Timothy P. Lillicrap, Fan Hui, Laurent Sifre, George van den Driessche, Thore Graepel, and Demis Hassabis. Mastering the game of go without human knowledge. *Nat.*, 550(7676):354–359, 2017.

Satinder Singh and Richard Yee. An upper bound on the loss from approximate optimal-value functions. *Machine Learning*, 16, 10 1996. doi: 10.1023/A:1022693225949.

Satinder P. Singh, Tommi Jaakkola, and Michael I. Jordan. Reinforcement learning with soft state aggregation. In *Proceedings of the 7th International Conference on Neural Information Processing Systems*, NIPS'94, page 361368, Cambridge, MA, USA, 1994. MIT Press.

Alexander L. Strehl, Lihong Li, and Michael L. Littman. Reinforcement learning in finite mdps: Pac analysis. *J. Mach. Learn. Res.*, 10:24132444, dec 2009. ISSN 1532-4435.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. A Bradford Book, Cambridge, MA, USA, 2018. ISBN 0262039249.

Oriol Vinyals, Igor Babuschkin, Wojciech M. Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H. Choi, Richard Powell, Timo Ewalds, Petko Georgiev, Junhyuk Oh, Dan Horgan, Manuel Kroiss, Ivo Danihelka, Aja Huang, Laurent Sifre, Trevor Cai, John P. Agapiou, Max Jaderberg, Alexander Sasha Vezhnevets, Rémi Leblond, Tobias Pohlen, Valentin Dalibard, David Budden, Yury Sulsky, James Molloy, Tom Le Paine, Çaglar Gülçehre, Ziyu Wang, Tobias Pfaff, Yuhuai Wu, Roman Ring, Dani Yogatama, Dario Wünsch, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy P. Lillicrap, Koray Kavukcuoglu, Demis Hassabis, Chris Apps, and David Silver. Grandmaster level in starcraft II using multi-agent reinforcement learning. *Nat.*, 575(7782):350–354, 2019.

Tsachy Weissman, Erik Ordentlich, Gadiel Seroussi, Sergio Verdú, and Marcelo J. Weinberger. Inequalities for the l1 deviation of the empirical distribution. 2003.

Timon Willi, Alistair Hp Letcher, Johannes Treutlein, and Jakob Foerster. Cola: consistent learning with opponent-learning awareness. In *International Conference on Machine Learning*, pages 23804–23831. PMLR, 2022.

Qizhen Zhang, Chris Lu, Animesh Garg, and Jakob Foerster. Centralized model and exploration policy for multi-agent rl. In *International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, 2022. URL `https://arxiv.org/abs/2107.06434v2`.

Stephen Zhao, Chris Lu, Roger B Grosse, and Jakob Foerster. Proximal learning with opponent-learning awareness. *Advances in Neural Information Processing Systems*, 35: 26324–26336, 2022.

A Theoretical Analysis

We will first delineate the notations and assumptions that will be used in the sample complexity proof.

Table 3. Nomenclature

Symbol	Definition
$\hat{s}_t = \hat{s}$	Meta-state at time t, time subscript t is omitted for convenience
$\hat{a}_t = \hat{a}$	Meta-action at time t, time subscript t is omitted for convenience
$(\hat{s}_d, \hat{a}_d)$	Discretized state-action pair in meta-game
$\hat{r}_d = \hat{r}(\hat{s}_d, \hat{a}_d)$	Meta-reward function parameterised by discretized meta-state-action pair
$\phi_t^i = \phi^i$	The set of inner-game policy parameters of our agent at time t, time subscript t is omitted for convenience
$\phi_{t,d}^i = \phi_d^i$	The set of discretized inner-game policy parameters of our agent at time t, time subscript t is omitted for convenience
$\boldsymbol{\phi}^{-i}$	The set of inner-game policy parameters of all agents except our agent at time t, time subscript t is omitted for convenience
$\boldsymbol{\phi_d^{-i}}$	The set of discretized inner-game policy parameters of all agents except our agent at time t, time subscript t is omitted for convenience
$\hat{R}(\hat{s}, \hat{a}), \hat{T}(\hat{s}, \hat{a})$	Empirical estimate of reward and transition distribution
$R(\hat{s}, \hat{a}), T(\hat{s}, \hat{a})$	True reward and transition distribution
$\varepsilon_{m,d}^R, \varepsilon_{m,d}^T$	Error between empirical discretized reward/ transition distribution and true reward/ transition distribution
$\varepsilon_m^R, \varepsilon_m^T$	Error between empirical reward/ transition distribution and true reward/ transition distribution
${\varepsilon_m^R}', {\varepsilon_m^T}'$	Error between empirical reward/ transition distribution and empirical discretized reward/ transition distribution
δ_R, δ_T	Probability that the difference between empirical discretized reward/ transition distribution and the true reward/ transition distribution is larger than $\varepsilon_{m,d}^R / \varepsilon_{m,d}^T$

A.1 Assumptions

We first outline all assumptions made in deriving the sample complexity of the M-FOS algorithm.

Assumption A.1. The observation function in the meta-game is deterministic.

Assumption A.2. Meta-reward distribution $R(s, a)$ is Lipschitz continuous.

Assumption A.3. Meta-transition distribution $T(\hat{s}, \hat{a})$ is Lipschitz continuous.

Assumption A.4. The first timestep τ in which the R_{max} explores a new state is finite.

Assumption A.5. Transition and reward distribution of $\hat{M}$ and $\hat{M}_m$ are identical for state-actions in K.

Assumption A.6. Failure probability is evenly split between the reward and transition estimation events.

Assumption A.7. The error between empirical discretized reward distribution and true reward distribution ${\varepsilon_{m,d}^R}' \leq C\frac{\varepsilon_d(1-\hat{\gamma})}{V_{max}}$.

Assumption A.8. The error between empirical discretized transition distribution and true transition distribution ${\varepsilon_m^T}' \leq C\frac{\varepsilon_d(1-\hat{\gamma})}{V_{max}}$.

Assumption A.9. Discount factor in the meta-game $\hat{\gamma}$ and that of the inner game γ are the same, i.e. $\hat{\gamma} = \gamma$.

Assumption A.10. The inner game is assumed to be a partially observable discrete and deterministic episodic game.

A.2 R_{max}

Definition A.11 (m-known MDP)**.** Let $M = \langle \mathcal{S}, \mathcal{A}, T, \mathcal{R}, \gamma \rangle$ be a MDP with action values $Q(s, a)$ for each state-action pair (s, a). Define K as the known set of state-action pairs and $M_m = \langle \mathcal{S}, \mathcal{A}, T_m, \mathcal{R}_m, \gamma \rangle$ as the known state-action MDP.

The algorithm has two categories of state-action pair: the *known* and the *unknown* state-action pair. A state-action pair is *m-known* if it has been visited for *m* times, and otherwise unknown if it is visited for less than *m* times. It is considered a successful exploration if the agent visits a state-action pair $(s, a) \notin K$. For all state-action pairs in K, the induced MDP M_m is identical to M. For state-action pairs not in K, these unknown pairs are self-absorbing and maximally rewarding (i.e. $r = r_{max}$), so the optimal policy for the empirical MDP $\hat{M}_m$ is either producing near-optimal reward or exploring states outside of K. These properties are listed in Table 4.

Table 4. Relationship between $M, M_m, \hat{M}_m$

	true MDP M	true induced MDP M_m	empirical MDP $\hat{M}_m$
Known	= M	= M	$\approx M$
Unknown	= M	self-loop with maximum reward	

The maximally rewarding property leads to an admissible heuristic for the value function $U(s, a)$ which is upper bounded by $V_{max} = \frac{R_{max}}{1-\gamma} = \frac{1}{1-\gamma}$.

$$0 \leq U(s, a) \leq \frac{1}{1-\gamma} \qquad \forall (s, a) \in \mathcal{S} \times \mathcal{A}$$

Definition A.12 (Sample complexity definition from Kakade 2003)**.** Let $c = [s_1, a_1, r_1, s_2, a_2, r_2, \cdots]$ be a random path generated by executing an algorithm $\mathcal{A}$ in an MDP M. For any fixed $\varepsilon > 0$, the sample complexity of exploration of $\mathcal{A}$ is the number of timesteps t such that the policy at time t, $\mathcal{A}_t$, satisfies $V^{\mathcal{A}_t}(s_t) \leq V^*(s_t) - \varepsilon$.

Definition A.13 (PAC-MDP definition from Strehl et al. 2009)**.** An algorithm $\mathcal{A}$ is said to be a PAC algorithm if, for any $\varepsilon > 0$ and $0 \leq \delta \leq 1$, the space and sample complexity of $\mathcal{A}$ per timestep are less than some polynomial dependence on $(\mathcal{S}, \mathcal{A}, \frac{1}{\varepsilon}, \frac{1}{\delta}, \frac{1}{1-\gamma})$ with probability at least 1-δ.

A.2.1 Empirical functions

Definition A.14 (m-Known MDP)**.** M_m is the expected version of $\hat{M}_m$ where:

$$T_m\left(\hat{s}' \mid \hat{s}, \hat{a}\right) := \begin{cases} T\left(\hat{s}' \mid \hat{s}, \hat{a}\right) & \text{if } (\hat{s}, \hat{a}) \in \text{m-known} \\ 1\left[\hat{s}' = \hat{s}\right] & \text{otherwise} \end{cases}$$

$$\hat{T}_m\left(\hat{s}' \mid \hat{s}, \hat{a}\right) := \begin{cases} \frac{n(\hat{s}, \hat{a}, \hat{s}')}{n(\hat{s}, \hat{a})}, & \text{if } (\hat{s}, \hat{a}) \in \text{m-known} \\ 1\left[\hat{s}' = \hat{s}\right], & \text{otherwise} \end{cases}$$

$$R_m\left(\hat{s}, \hat{a}\right) := \begin{cases} R\left(\hat{s}, \hat{a}\right), & \text{if } (\hat{s}, \hat{a}) \in \text{m-known} \\ R_{\text{max}} & \text{otherwise} \end{cases}$$

$$\hat{R}_m(\hat{s}, \hat{a}) = \begin{cases} \frac{\sum_i^{n(\hat{s}, \hat{a})} r(\hat{s}, \hat{a})}{n(\hat{s}, \hat{a})}, & \text{if } (\hat{s}, \hat{a}) \in \text{m-known} \\ R_{\text{max}}, & \text{otherwise} \end{cases}$$

$\hat{R}_m(\hat{s}, \hat{a})$ and $\hat{T}_m\left(\hat{s}' \mid \hat{s}, \hat{a}\right)$ are the maximum-likelihood estimates for the reward and transition distribution of state-action pair (s, a) with $n(s, a) = m$ observations of (s, a). m is a value that acts like a threshold for doing the Q-value update step, and it will be derived in the later section. The R_{max} algorithm takes a greedy action $\max_a Q(s, a)$ and the Q-function is updated only when the visitation count is larger than or equal to m, i.e.

$$Q(s, a) = \begin{cases} R_m(s, a) + \gamma \sum_{s'} T_m(s'|s, a) \max_a Q(s', a'), & \text{if } n(s, a) \geq m \\ U(s, a), & \text{otherwise} \end{cases} \tag{7}$$

A.2.2 Value iteration

To solve equation 7, we use value iteration, a standard and simple approach. R_{max} guarantees a near-optimal greedy policy rather than the exact solution for equation 7. The algorithm performs value iteration several times (will be quantified below) to obtain an ε-optimal policy.

Proposition A.15. *(Singh and Yee, 1996) Let $Q(\cdot,\cdot)$ and $Q^*(\cdot,\cdot)$ be two Q functions over the same state and action spaces of an MDP M, and Q^* is the optimal value function. Let π be a greedy policy to Q, π^* be a greedy policy to Q^* (also the optimal policy for M). For any $\varepsilon \geq 0$ and $\gamma \leq 1$, if $\max_{s,a} |Q(s,a) - Q^*(s,a)| \leq \frac{\varepsilon(1-\gamma)}{2}$, then $\max_s\{V^{\pi^*}(s) - V^{\pi}(s)\} \leq \varepsilon$.*

Proposition A.16. *(Strehl et al., 2009) Let $\varepsilon \in \mathbb{R}^+_*$ which satisfies $\varepsilon < \frac{1}{1-\gamma}$. Suppose value iteration is run for $\lceil \frac{\ln(\frac{1}{\varepsilon(1-\gamma)})}{1-\gamma} \rceil$ iterations and $Q(\cdot,\cdot)$ is initialised such that $0 \leq Q(\cdot,\cdot) \leq \frac{1}{1-\gamma}$, we have $\max_{s,a}\{|Q(s,a) - Q^*(s,a)|\} \leq \varepsilon$.*

Proof. Let $Q_i(s,a)$ be the action-value estimate after the i^{th} value iteration and $\delta_i \coloneqq \max_{s,a} |Q^*(s,a) - Q_i(s,a)|$, such that

$$\begin{aligned}
\Delta_i &= \max_{s,a} |(R(s,a) + \gamma \sum_{s'} T(s,a,s')V^*(s')) - (R(s,a) + \gamma \sum_{s'} T(s,a,s')V_{i-1}(s')) \\
&= \max_{s,a} |\gamma \sum_{s'} T(s,a,s')(V^*(s') - V_{i-1}(s'))| \\
&\leq \gamma \Delta_{i-1}
\end{aligned}$$

Because of the initialisation, $\Delta_0 \leq \frac{1}{1-\gamma}$. Thus $\Delta_i \leq \frac{\gamma^i}{1-\gamma}$. We limit Δ_i to be at most ε and solve for i:

$$\begin{aligned}
\varepsilon &\leq \frac{\gamma^i}{1-\gamma} \\
\varepsilon(1-\gamma) &\leq \gamma^i \\
i &\geq \frac{\ln(\varepsilon(1-\gamma))}{\ln(\gamma)}
\end{aligned}$$

With the identity $e^x \geq 1 + x$, we have $1 - \gamma \leq -\ln \gamma$:

$$\begin{aligned}
\frac{\ln(\varepsilon(1-\gamma))}{\ln(\gamma)} &\leq \frac{\ln(\frac{1}{\varepsilon(1-\gamma)})}{1-\gamma} \\
i &\geq \frac{\ln(\frac{1}{\varepsilon(1-\gamma)})}{1-\gamma}
\end{aligned}$$

□

Thus it is sufficient to run value iteration for $\mathcal{O}(\frac{ln(\frac{1}{\varepsilon(1-\gamma)})}{1-\gamma})$ times to produce an ε-optimal policy.

A.3 Discretization Setup with ε-Net

We apply the concept of ε-Net to discretize our meta-state and meta-action space.

Definition A.17. (ε-Net, Erdogdu 2022) For $\varepsilon > 0$, $\mathcal{N}_\varepsilon$ is an ε-net over the set $\Theta \subseteq \mathbb{R}^d$ if for all $\theta \in \Theta$, there exists $\theta' \in \mathcal{N}_\varepsilon$ such that $\|\theta - \theta'\| \leq \varepsilon$. The size of the ε-net with smallest size $|\mathcal{N}_\varepsilon|$ is called the covering number.

Figure 6 shows an example of a 2-dimensional ε-Net. We divide the circle into grids of size λ so the total number of points required in this 2-dimensional space is $(\frac{2R}{\lambda} + 1)^2$. The power of 2 corresponds to the number of dimensions. Within each grid, the highest length between the vertices and any interior points is the apothem, which is $\frac{\lambda\sqrt{d}}{2} = \frac{\sqrt{2}\lambda}{2}$. Therefore, the largest grid size that we can have should satisfy $\varepsilon = \frac{\lambda\sqrt{d}}{2} = \frac{\sqrt{2}\lambda}{2}$ to ensure all points in the ball are covered.

An agent's inner Q-table has $|\mathcal{S}| \times |\mathcal{A}|$ parameters, so the size of an agent's policy parameters is $|\phi^i| = |\mathcal{S}| \times |\mathcal{A}|$. This equates to the number of dimensions in the ε-Net. We discretize each entry of an inner Q-table/ each policy parameter according to radius λ so that

$$\left\|\phi^i - \phi^i_d\right\|_2 \leq \frac{\lambda\sqrt{|\mathcal{S}||\mathcal{A}|}}{2}$$

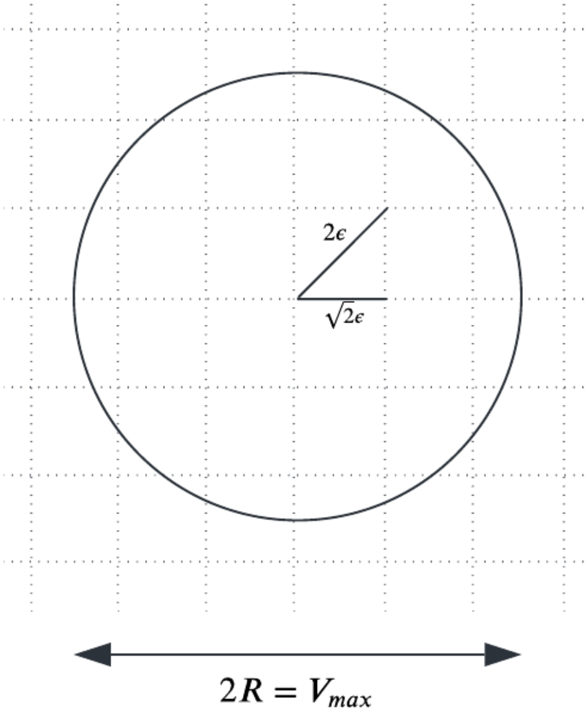


Figure 6. ε-Net for $\Theta = \{\theta \in \mathbb{R}^2 : \|\theta\| \leq R\}$

To avoid confusion in notation, we define $\alpha = \frac{\lambda\sqrt{|\mathcal{S}||\mathcal{A}|}}{2}$, where α is essentially ε in ε-Net. We then apply this α-Net to meta-state $\hat{s}_d$ and discretized meta-action $\hat{a}_d$,

$$\begin{aligned} \hat{s} &= \boldsymbol{\phi} \\ \|\hat{s} - \hat{s}_d\|_2 &\leq \frac{\lambda\sqrt{n|\mathcal{S}||\mathcal{A}|}}{2} \\ &\leq \sqrt{n}\alpha \end{aligned} \qquad \begin{aligned} \hat{a} &= \phi^i \\ \|\hat{a} - \hat{a}_d\|_2 &\leq \frac{\lambda\sqrt{|\mathcal{S}||\mathcal{A}|}}{2} \\ &\leq \alpha \end{aligned}$$

The size of this α-net will be derived in Appendix A.5 for the sample complexity expression.

A.4 Basic Lemmas and Definitions

Lemma A.18. *(Hoeffding's inequality) Let $Z_1, \cdots, Z_n$ be independent bounded random variables with $Z_i \in [a, b]$ for all I, where $-\infty < a \leq b < \infty$. Then*

$$P(\frac{1}{n}\sum_{i=1}^{n}(Z_i - \mathbb{E}[Z_i]) \geq t) \leq e^{-\frac{2nt^2}{(b-a)^2}} \tag{8}$$

Lemma A.19. *(Weissman's inequality for L1 deviation of the empirical distribution from Weissman et al. 2003) Let $\mathcal{D}$ be a probability distribution on the set $\mathcal{A} = \{1, \cdots, a\}$ and $\boldsymbol{\mathcal{X}^m} = X_1, X_2, \cdots, X_m \in \boldsymbol{\mathcal{A}^m}$ be independent identically distributed random variables distributed according to $\mathcal{Q}$. $\hat{\mathcal{Q}}_{\boldsymbol{\mathcal{X}^m}}$ is the empirical probability distribution on $\mathcal{A}$ such that $\hat{\mathcal{Q}}_{\boldsymbol{\mathcal{X}^m}}(j) = \frac{1}{m}\sum_{i=1}^{m}\mathbf{1}(x_i = j)$. ($\mathbf{1}(\cdot)$ is the indicator function of the specified event.)*
For $p \in [0, 0.5)$, we define function

$$\psi(p) = \frac{1}{1-2p}\log\frac{1-p}{p}. \tag{9}$$

For a probability distribution $\mathcal{Q}$ on $\mathcal{A}$, we define

$$\pi_{\mathcal{Q}} = \max_{A \subseteq \mathcal{A}}[min(\mathcal{Q}(A), 1 - \mathcal{Q}(A))]. \tag{10}$$

Then, $\forall \varepsilon > 0$,

$$P(\quad \left\| \mathcal{Q} - \hat{\mathcal{Q}}_{\boldsymbol{\mathcal{X}}^m} \right\|_1 \geq \varepsilon) \leq (2^a - 2)e^{\frac{-m\psi(\pi_{\mathcal{Q}})\varepsilon^2}{4}} \tag{11}$$

Lemma A.20. *(Chernoff-Hoeffding Bound) When flipping a weighted coin, there is a probability $p > 0$ of landing with heads-up. Then, for any positive integer k and real number $\delta \inf[0, 1]$, there exists a number $m = \mathcal{O}(\frac{k}{p} \ln \frac{1}{\delta})$, such that after m tosses, we will observe k or more heads with probability at least $1 - \delta$.*

Lemma A.21. *Let $M = \langle \hat{\mathcal{S}}, \hat{\mathcal{A}}^i, T, R^i, \hat{\gamma} \rangle$ be an MDP in the meta-game in which its optimal value function is upper bounded by V_{max}, and $\hat{M}_m$ be a known state-action MDP defined with value function $Q(\hat{s}, \hat{a})$. Then $\forall \hat{s} \in \hat{S}$,*

$$V^*_{\hat{M}_m}(\hat{s}) \leq V_{max} + \frac{\sqrt{n+1}\alpha(1 - \hat{\gamma}^{\tau-1})}{1 - \hat{\gamma}} + \max_{\hat{s}, \hat{a}} Q(\hat{s}', \hat{a}')$$

Proof. For any policy π and any state $\hat{s} \in \hat{S}$, let $(\hat{s}_1, \hat{a}_1, \hat{r}_1, \hat{s}_2, \hat{a}_2, \hat{r}_2, \cdots)$ be a path generated by starting in state $\hat{s} = \hat{s}_1$ and following π in the known state-action MDP $\hat{M}_m$, where $\hat{s}_t$ and $\hat{r}_t$ are the meta-state and reward at timestep t, and $\hat{a}_t = \pi(\hat{s}_t)$. The value function $V^*_{\hat{M}_m}(\hat{s})$ is the expected discounted total reward accumulated on the random path and can be written as (Sutton and Barto, 2018).

$$V^*_{\hat{M}_m}(\hat{s}) = E_{\hat{M}_m}[\hat{r}_1 + \hat{\gamma}\hat{r}_2 + \hat{\gamma}^2\hat{r}_3 + \cdots | \hat{s}_1 = \hat{s}, \pi]$$

Let τ be the first timestep in which $(\hat{s}_{d,\tau}, \hat{a}_{d,\tau}) \notin K$ so that with the construction of $\hat{M}_m$ (see Table 4) and assuming τ is finite,

$$\hat{s}_{d,\tau} = \hat{s}_{d,\tau+1} = \hat{s}_{d,\tau+2} = \cdots, \qquad \hat{a}_{d,\tau} = \hat{a}_{d,\tau+1} = \hat{a}_{d,\tau+2} = \cdots = \pi(\hat{s}_{d,\tau})$$

From Equation (7):

$$\hat{r}_{d,\tau} = \hat{r}_{d,\tau+1} = \hat{r}_{d,\tau+2} = \cdots = (1 - \hat{\gamma})Q(\hat{s}_{d,\tau}, \hat{a}_{d,\tau}).$$

Thus, for any fixed $\tau \geq 1$, the discounted total reward is

$$\begin{aligned} &\hat{r}_1 + \hat{\gamma}\hat{r}_2 + \hat{\gamma}^2\hat{r}_3 + \cdots \\ &\leq \hat{r}_{d,1} + \hat{\gamma}\hat{r}_{d,2} + \cdots + \hat{\gamma}^{\tau-2}\hat{r}_{d,\tau-1} + \hat{\gamma}^{\tau-1}Q(\hat{s}_{d,\tau}, \hat{a}_{d,\tau}) \\ &\leq \hat{r}_{d,1} + \hat{\gamma}\hat{r}_{d,2} + \cdots + \max_{\hat{s}', \hat{a}'} Q(\hat{s}', \hat{a}') \end{aligned} \tag{12}$$

where the $\max_{\hat{s}', \hat{a}'} Q(\hat{s}', \hat{a}')$ is formed because of the way we define the transition and reward functions in $\hat{M}_m$. Since this upper bound holds for all fixed $\tau \geq 1$ and assuming the transition and reward functions of M and $\hat{M}_m$ are identical for state-actions in K, we have

$$\begin{aligned} &\mathbb{E}_{\hat{M}_m}[\hat{r}_1 + \hat{\gamma}\hat{r}_2 + \hat{\gamma}^2\hat{r}_3 + \cdots | \hat{s}_1 = \hat{s}, \pi] \\ &\leq \mathbb{E}_{\hat{M}_m}[\hat{r}_{d,1} + \hat{\gamma}\hat{r}_{d,2} + \cdots + \hat{\gamma}^{\tau-2}\hat{r}_{d,\tau-1} + \max_{\hat{s}', \hat{a}'} Q(\hat{s}', \hat{a}') | \hat{s}_1 = \hat{s}_d, \pi] \\ &\leq \mathbb{E}_M[\hat{r}_{d,1} + \hat{\gamma}\hat{r}_{d,2} + \cdots + \hat{\gamma}^{\tau-2}\hat{r}_{d,\tau-1} | \hat{s}_1 = \hat{s}, \pi] + \max_{\hat{s}', \hat{a}'} Q(\hat{s}', \hat{a}') \\ &\leq V^\pi_M(\hat{s}_d) + \max_{\hat{s}', \hat{a}'} Q(\hat{s}', \hat{a}') \\ &\leq V_{max} + \max_{\hat{s}', \hat{a}'} Q(\hat{s}', \hat{a}'). \end{aligned}$$

□

Lemma A.22. *Suppose $\hat{r}_1, \hat{r}_2, \cdots, \hat{r}_m$ are m rewards drawn from rewards distribution $\mathcal{R}(\hat{s}_d, \hat{a}_d)$. Let $\hat{R}(\hat{s}_d, \hat{a}_d)$ be the empirical estimate of $R(\hat{s}_d, \hat{a}_d)$, $\delta_R \in \mathbb{R}^+$ and $\delta_R < 1$, with probability at least $1 - \delta_R$,*

$$|\hat{R}(\hat{s}_d, \hat{a}_d) - R(\hat{s}_d, \hat{a}_d)| \leq \frac{1}{2}\sqrt{\frac{2}{m} ln \frac{2}{\delta_R}} \tag{13}$$

is true for all $\hat{s}_d \in \hat{\mathcal{S}}_d, \hat{a}_d \in \hat{\mathcal{A}}_d$.

Proof. Let ε_m^R be the error between the empirical discretized reward distribution $\hat{R}(\hat{s}_d, \hat{a}_d)$ and the true discretized reward function $R(\hat{s}_d, \hat{a}_d)$ such that with Equation (8),

$$\begin{aligned} P(|\hat{R}(\hat{s}_d, \hat{a}_d) - R(\hat{s}_d, \hat{a}_d)| \geq \varepsilon_m^R) &\leq 2e^{-2(\varepsilon_m^R)^2 m} \\ \delta_R &\leq 2e^{-2(\varepsilon_m^R)^2 m} \\ \frac{2}{\delta_R} &\geq e^{2(\varepsilon_m^R)^2 m} \\ \frac{1}{2m} ln \frac{2}{\delta_R} &\geq (\varepsilon_m^R)^2 \\ \varepsilon_m^R &\leq \frac{1}{2}\sqrt{\frac{2}{m}\ln\frac{2}{\delta_R}} \end{aligned}$$

Lemma A.23. *Suppose $\hat{T}(\hat{s}_d, \hat{a}_d)$ is the empirical transition distribution for discretized state-action pair $(\hat{s}_d, \hat{a}_d)$ using m samples of next states drawn independently from the true transition distribution $T(\hat{s}_d, \hat{a}_d)$. We define*

$$T(s'|s_d, a_d) = \int_{s\prime \in s_d \pm \alpha} f(s'|s, a) ds'$$

$\forall s_d, a_d$, where $f(\cdot|s, a)$ is the probability density function. Let $\delta_T \in \mathbb{R}^+$ and $\delta_T < 1$, with probability at least $1 - \delta_T$,

$$\left\|\hat{T}(\hat{s}_d, \hat{a}_d) - T(\hat{s}_d, \hat{a}_d)\right\|_1 \leq \sqrt{\frac{2[\ln(2^{|\hat{\mathcal{S}}|} - 2) - \ln\delta_T]}{m}}$$

is true for all $\hat{s}_d \in \hat{\mathcal{S}}_d, \hat{a}_d \in \hat{\mathcal{A}}_d$.

Proof. With Weissman's inequality from Equation (11), we need to find the bound of the function $\psi(p)$ (Equation (9)) first. Given the input $p \in [0, 0.5)$, $\psi(p) \in (2, \infty]$, the upper bound of $e^{-\psi(p)}$ is 2. Let ε_m^T be the error between the empirical discretized transition distribution $\hat{T}(\hat{s}_d, \hat{a}_d)$ and the true discretized transition function $T(\hat{s}_d, \hat{a}_d)$ such that

$$\begin{aligned} P(\left\|\hat{T}(\hat{s}_d, \hat{a}_d) - T(\hat{s}_d, \hat{a}_d)\right\|_1 \geq \varepsilon_m^T) &\leq (2^{|\hat{\mathcal{S}}|} - 2)e^{\frac{-m\psi(\pi_T)\varepsilon_m^{T\,2}}{4}} \\ \delta_T &\leq (2^{|\hat{\mathcal{S}}|} - 2)e^{\frac{-m2\varepsilon_m^{T\,2}}{4}} \\ \ln\frac{2^{|\hat{\mathcal{S}}|} - 2}{\delta_T} &\geq \frac{m\varepsilon_m^{T\,2}}{2} \\ \varepsilon_m^{T\,2} &\leq \frac{2}{m}(\ln(2^{|\hat{\mathcal{S}}|} - 2) - \ln\delta_T) \\ \varepsilon_m^T &\leq \sqrt{\frac{2[\ln(2^{|\hat{\mathcal{S}}|} - 2) - \ln\delta_T]}{m}} \end{aligned}$$

Corollary A.24. *Suppose m transitions and m rewards are drawn independently from the transition distribution $T(\hat{s}_d, \hat{a}_d)$ and reward distribution $R(\hat{s}_d, \hat{a}_d)$ and let $\delta \in \mathbb{R}^+$ and $\delta \in (0, 1]$. With probability at least $1 - \delta$,*

$$\left\|\hat{T}(\hat{s}_d, \hat{a}_d) - T(\hat{s}_d, \hat{a}_d)\right\|_1 \leq \varepsilon_m^T, \quad \varepsilon_m^T := \sqrt{\frac{2[\ln(2^{|\hat{\mathcal{S}}|} - 2) + \ln\frac{2|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta}]}{m}} \tag{14}$$

$$|\hat{R}(\hat{s}_d, \hat{a}_d) - R(\hat{s}_d, \hat{a}_d)| \leq \varepsilon_m^R, \quad \varepsilon_m^R := \frac{1}{2}\sqrt{\frac{2}{m}\ln\frac{4|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta}} \tag{15}$$

Proof. By applying union bound to Theorem A.22 and Theorem A.23 and setting $\delta_R = \delta_T = \frac{\delta}{2|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}$. The factor $\frac{1}{2}$ is created by assuming that the failure probability is evenly split between the reward and transition estimation events. We then split $\frac{\delta}{2}$ among the state-action pairs, resulting in a factor $\frac{1}{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}$. □

Lemma A.25. *Suppose MDP M_1 and M_2 only differ in dynamics, i.e. $M_1 = \langle \hat{\mathcal{S}}, \hat{\mathcal{A}}^i, T_1, R_1^i, \hat{\gamma} \rangle$ and $M_2 = \langle \hat{\mathcal{S}}, \hat{\mathcal{A}}^i, T_2, R_2^i, \hat{\gamma} \rangle$. Let $\max_{s,a} |R_1(s,a) - R_2(s,a)| \leq \varepsilon_R$ and $\max_{s,a} |T_1(s,a) - T_2(s,a)| \leq \varepsilon_T$,then $\forall \pi : \hat{\mathcal{S}} \to \hat{\mathcal{A}}$ and $\forall s \in \hat{\mathcal{S}}$,*

$$\left\| V^*_{M_1} - V^*_{M_2} \right\|_\infty \leq \frac{\varepsilon_R}{1-\hat{\gamma}} + \frac{\hat{\gamma}\varepsilon_T V_{max}}{2(1-\hat{\gamma})}$$

Proof. Let $\mathscr{T}_1, \mathscr{T}_2$ be the Bellman update operator of M_1, M_2 respectively such that

$$\left\| V^*_{M_1} - V^*_{M_2} \right\|_\infty = \left\| V^*_{M_1} - \mathscr{T}_2 V^*_{M_2} \right\|_\infty = \left\| V^*_{M_1} - \mathscr{T}_2 V^*_{M_1} + \mathscr{T}_2 V^*_{M_1} - \mathscr{T}_2 V^*_{M_2} \right\|_\infty$$

$$\begin{aligned}
\left\| V^*_{M_1} - \mathscr{T}_2 V^*_{M_1} \right\|_\infty &= \left\| \mathscr{T}_1 V^*_{M_1} - \mathscr{T}_2 V^*_{M_1} \right\|_\infty \\
&= \max_{s,a} |R_1(s,a) + \hat{\gamma}\mathbb{E}_{s' \sim T_1(s,a)}[V^*_{M_1}(s')] - R_2(s,a) - \hat{\gamma}\mathbb{E}_{s' \sim T_2(s,a)}[V^*_{M_1}(s')]| \\
&= \varepsilon_R + \hat{\gamma} \max_{s,a} < \ T_1(s,a) - T_2(s,a), V^*_{M_1} - \frac{V_{max}}{2} \cdot \mathbf{1}_{|\hat{\mathcal{S}}| \times 1} \ >
\end{aligned}$$

By Holder's inequality:

$$\begin{aligned}
&\leq \varepsilon_R + max_{s,a} \left\| T_1(s,a) - T_2(s,a) \right\|_1 \left\| V^*_{M_1} - \frac{V_{max}}{2} \cdot \mathbf{1} \right\|_\infty \\
&\leq \varepsilon_R + \hat{\gamma}\varepsilon_T \cdot \frac{V_{max}}{2}.
\end{aligned}$$

By the property of an optimal value function V^*,

$$\left\| \mathscr{T}_2 V^*_{M_1} - \mathscr{T}_2 V^*_{M_2} \right\|_\infty = \hat{\gamma} \left\| V^*_{M_1} - V^*_{M_2} \right\|_\infty$$

Thus,

$$\begin{aligned}
\left\| V^*_{M_1} - V^*_{M_2} \right\|_\infty &= \left\| V^*_{M_1} - \mathscr{T}_2 V^*_{M_1} + \mathscr{T}_2 V^*_{M_1} - \mathscr{T}_2 V^*_{M_2} \right\|_\infty \\
&\leq \varepsilon_R + \hat{\gamma}\varepsilon_T \cdot \frac{V_{max}}{2} + \hat{\gamma} \left\| V^*_{M_1} - V^*_{M_2} \right\|_\infty \\
&\leq \frac{\varepsilon_R}{1-\hat{\gamma}} + \frac{\hat{\gamma}\varepsilon_T V_{max}}{2(1-\hat{\gamma})}
\end{aligned}$$

□

Lemma A.26. *(Simulation Lemma) Let $M_1 = \langle \hat{\mathcal{S}}, \hat{\mathcal{A}}^i, T_1, R_1^i, \hat{\gamma} \rangle$ and $M_2 = \langle \hat{\mathcal{S}}, \hat{\mathcal{A}}^i, T_2, R_2^i, \hat{\gamma} \rangle$ be two MDP with non-negative rewards bounded by $R_{max} = 1$ and optimal value functions bounded by V_{max}. Let $max_{s,a}|R_1(s,a) - R_2(s,a)| \leq \varepsilon_R$ and $max_{s,a}|T_1(s,a) - T_2(s,a)| \leq \varepsilon_T$, then $\forall \pi : \hat{\mathcal{S}} \to \hat{\mathcal{A}}$ and $\forall s \in \hat{\mathcal{S}}$,*

$$\left\| V^\pi_{M_1} - V^\pi_{M_2} \right\|_\infty \leq \frac{\varepsilon_R}{1-\hat{\gamma}} + \frac{\hat{\gamma}\varepsilon_T R_{max}}{2(1-\hat{\gamma})^2}$$

□

Proof.

$$\begin{aligned}
|V^\pi_{M_1} - V^\pi_{M_2}| &= |R_1(s,\pi) + \hat{\gamma}\langle T_1(s,\pi), V^\pi_{M_1} \rangle - R_2(s,\pi) - \hat{\gamma}\langle T_2(s,\pi), V^\pi_{M_2} \rangle| \\
&\leq \varepsilon_R + \hat{\gamma}|\langle T_1(s,\pi), V^\pi_{M_1} \rangle - \langle T_2(s,\pi), V^\pi_{M_1} \rangle + \langle T_2(s,\pi), V^\pi_{M_1} \rangle - \langle T_2(s,\pi), V^\pi_{M_2} \rangle| \\
&\leq \varepsilon_R + \hat{\gamma}|\langle T_1(s,\pi) - T_2(s,\pi), V^\pi_{M_1} \rangle| + \hat{\gamma} \left\| V^\pi_{M_1} - V^\pi_{M_2} \right\|_\infty \\
&= \varepsilon_R + \hat{\gamma}|\langle T_1(s,\pi) - T_2(s,\pi), V^\pi_{M_1} - \frac{R_{max}}{2(1-\hat{\gamma})} \rangle| + \hat{\gamma} \left\| V^\pi_{M_1} - V^\pi_{M_2} \right\|_\infty
\end{aligned}$$

By Holder's inequality,

$$\begin{aligned}
&\leq \varepsilon_R + \hat{\gamma}|T_1(s,\pi) - T_2(s,\pi)|_1 \cdot \left\| V^\pi_{M_1} - \frac{R_{max}}{2(1-\hat{\gamma})} \right\|_\infty + \hat{\gamma} \left\| V^\pi_{M_1} - V^\pi_{M_2} \right\|_\infty \\
&\leq \varepsilon_R + \frac{\hat{\gamma}\varepsilon_T R_{max}}{2(1-\hat{\gamma})} + \hat{\gamma} \left\| V^\pi_{M_1} - V^\pi_{M_2} \right\|_\infty
\end{aligned}$$

□

An explanation for line 4: Since $T_1(s,\pi)$ and $T_2(s,\pi)$ are probability distributions that sum up to 1, we subtract $\frac{R_{max}}{2(1-\hat{\gamma})} \cdot \mathbf{1}$ to centre the range of $V^{\pi}_{M_1}$ around 0. This step achieves a tighter bound.

Corollary A.27. *From Theorem A.25 and Theorem A.26, we can show that* $\forall \pi$*,*

$$|J_{M_{2,K}}(\pi) - J_{M_{1,K}}(\pi)| \leq \frac{\varepsilon_R}{1-\hat{\gamma}} + \frac{\hat{\gamma}\varepsilon_T R_{max}}{2(1-\hat{\gamma})^2}$$

and

$$|J_{M_{1,K}}(\pi^*_{M_{1,K}}) - J_{M_{2,K}}(\pi^*_{M_{2,K}})| \leq \frac{\varepsilon_R}{1-\hat{\gamma}} + \frac{\hat{\gamma}\varepsilon_T R_{max}}{2(1-\hat{\gamma})^2}$$

Proof. Let d_0 be the initial state distribution such that $J(\pi) = \mathbb{E}_{s\sim d_0}[V^\pi] = \langle d_0, V^\pi \rangle$

$$\begin{aligned}
|J_{M_{2,K}}(\pi) - J_{M_{1,K}}(\pi)| &= |\langle d_0, V_{M_{2,K}} \rangle - \langle d_0, V_{M_{1,K}} \rangle| \\
&= |\langle d_0, V_{M_{2,K}} - V_{M_{1,K}} \rangle| \\
&\leq \langle d_0, \left\| V_{M_{2,K}} - V_{M_{1,K}} \right\|_\infty \cdot \mathbf{1}_{|\hat{\mathcal{S}}|\times 1} \rangle \\
&= \left\| V_{M_{2,K}} - V_{M_{1,K}} \right\|_\infty \cdot \langle d_0, \mathbf{1} \rangle \\
&= \left\| V_{M_{2,K}} - V_{M_{1,K}} \right\|_\infty && (d_0 \text{ is a probability distribution}) \\
&\leq \frac{\varepsilon_R}{1-\hat{\gamma}} + \frac{\hat{\gamma}\varepsilon_T R_{max}}{2(1-\hat{\gamma})^2} && (\text{by Theorem A.25})
\end{aligned}$$

The proof for $|J_{M_{1,K}}(\pi^*_{M_{1,K}}) - J_{M_{2,K}}(\pi^*_{M_{2,K}})|$ can be obtained by replacing the last step with Theorem A.26. □

Lemma A.28. *(Applied Simulation Lemma) Let* $M_d = \langle \hat{\mathcal{S}}_d, \hat{\mathcal{A}}^i_d, T_M, R^i_M, \hat{\gamma} \rangle$ *be the true discretized MDP and* $\hat{M}_{d,m} = \langle \hat{\mathcal{S}}_d, \hat{\mathcal{A}}^i_d, T_{\hat{M}_{d,m}}, R^i_{\hat{M}_{d,m}}, \hat{\gamma} \rangle$ *be the empirical discretized m-known MDP, both with non-negative rewards bounded by* $R_{max} = 1$ *and optimal value functions bounded by* V_{max}*. Let* $max_{s_d,a_d}|R_{M_d}(\hat{s}_d, \hat{a}_d) - R_{\hat{M}_{d,m}}(\hat{s}_d, \hat{a}_d)| \leq \varepsilon_R$*, then* $\forall \pi : \hat{\mathcal{S}}_d \to \hat{\mathcal{A}}_d$ *and* $\forall s_d \in \hat{\mathcal{S}}_d$*,*

$$\left\| V^\pi_{M_d} - V^\pi_{\hat{M}_{d,m}} \right\|_\infty \leq \frac{\varepsilon_R}{1-\hat{\gamma}} + \frac{\hat{\gamma}\varepsilon^T_m R_{max}}{2(1-\hat{\gamma})^2}$$

Proof.

$$\begin{aligned}
|V^\pi_{M_d} - V^\pi_{\hat{M}_{d,m}}| &= |R_{M_d}(s_d, \pi^*_{M_d}) + \hat{\gamma}\langle T_{M_d}(s_d, \pi^*_{M_d}), V^{\pi^*_{M_d}}_{M_d} \rangle - R_{\hat{M}_{d,m}}(s_d, \pi^*_{\hat{M}_{d,m}}) - \hat{\gamma}\langle T_{\hat{M}_{d,m}}(s_d, \pi^*_{\hat{M}_{d,m}}), V^{\pi^*_{\hat{M}_{d,m}}}_{\hat{M}_{d,m}} \rangle| \\
&\leq \varepsilon_R + \hat{\gamma}|\langle T_{M_d}(s_d, \pi^*_{M_d}), V^{\pi^*_{M_d}}_{M_d} \rangle - \langle T_{\hat{M}_{d,m}}(s_d, \pi^*_{\hat{M}_{d,m}}), V^{\pi^*_{M_d}}_{M_d} \rangle + \langle T_{\hat{M}_{d,m}}(s_d, \pi^*_{\hat{M}_{d,m}}), V^{\pi^*_{M_d}}_{M_d} \rangle \\
&- \langle T_{\hat{M}_{d,m}}(s_d, \pi^*_{\hat{M}_{d,m}}), V^{\pi^*_{\hat{M}_{d,m}}}_{\hat{M}_{d,m}} \rangle| \\
&\leq \varepsilon^R_m + \hat{\gamma}|\langle T_{M_d}(s_d, \pi^*_{M_d}) - T_{\hat{M}_{d,m}}(s_d, \pi^*_{\hat{M}_{d,m}}), V^{\pi^*_{M_d}}_{M_d} \rangle| + \hat{\gamma}\left\| V^{\pi^*_{M_d}}_{M_d} - V^{\pi^*_{\hat{M}_{d,m}}}_{\hat{M}_{d,m}} \right\|_\infty \\
&\leq \varepsilon^R_m + \hat{\gamma}|\langle \varepsilon^T_m, V^{\pi^*_{M_d}}_{M_d} - \frac{R_{max}}{2(1-\hat{\gamma})} \rangle| + \hat{\gamma}\left\| V^{\pi^*_{M_d}}_{M_d} - V^{\pi^*_{\hat{M}_{d,m}}}_{\hat{M}_{d,m}} \right\|_\infty
\end{aligned}$$

By Holder's inequality,

$$\begin{aligned}
&\leq \varepsilon^R_m + \hat{\gamma}\varepsilon^T_m \cdot \left\| V^{\pi^*_{M_d}}_{M_d} - \frac{R_{max}}{2(1-\hat{\gamma})} \right\|_\infty + \hat{\gamma}\left\| V^{\pi^*_{M_d}}_{M_d} - V^{\pi^*_{\hat{M}_{d,m}}}_{\hat{M}_{d,m}} \right\|_\infty \\
&\leq \varepsilon^R_m + \frac{\hat{\gamma}\varepsilon^T_m R_{max}}{2(1-\hat{\gamma})} + \hat{\gamma}\left\| V^{\pi^*_{M_d}}_{M_d} - V^{\pi^*_{\hat{M}_{d,m}}}_{\hat{M}_{d,m}} \right\|_\infty
\end{aligned}$$

□

Lemma A.29. *(Induced Inequality) Suppose the reward and transition functions of M_1 and M_2 agree exactly on $K \subseteq \hat{\mathcal{S}} \times \hat{\mathcal{A}}$. Let escape$_m(\tau)$ be 1 if trajectory τ visits $(s,a) \notin K$, 0 otherwise. Let $J_M(\pi) := \mathbb{E}[\sum_{h-1}^{\infty} \hat{\gamma}^{h-1} r_h|\pi]$ be a measure of policy π's performance. $\forall \pi : \hat{\mathcal{S}} \times \hat{\mathcal{A}}$,*

$$|J_{M_1}(\pi) - J_{M_2}(\pi)| \le V_{max} \cdot P_{M_1}[\mathit{escape}_m(\tau)|\pi]$$

Proof. Let $R_M(\tau)$ be the sum of discounted reward in τ according to the reward function of M such that $v^{\pi}_{M_1} = \sum_{\tau} P_{M_1}[\tau|\pi] R_{M_1}(\tau)$ and $v^{\pi}_{M_2} = \sum_{\tau} P_{M_2}[\tau|\pi] R_{M_2}(\tau)$. For τ that satisfies $\text{escape}_m(\tau) = 1$, we define $\text{pre}_m(\tau)$ as the prefix of τ where only the last state-action pair of τ is not in K (only the last state-action pair escapes). We define $\text{suf}_m(\tau)$ as the remainder of the episode. Let $R(\text{pre}_m(\tau))$ be the sum of discounted rewards within the prefix, $P_{M_1}[\text{pre}_m(\tau)|\pi]$ be the marginal probability of the prefix assigned by M under policy π. To find the upper bound of $J_{M_1}(\pi) - J_{M_2}(\pi)$, we first find the upper bound of $J_{M_1}(\pi)$,

$$\begin{aligned}
&J_{M_1}(\pi) \\
&= \sum_{\tau:\text{escape}_m(\tau)=1} P_{M_1}[\tau|\pi](R_{M_1}(\text{pre}_m(\tau)) + R_{M_1}(\text{suf}_m(\tau))) + \sum_{\tau:\text{escape}_m(\tau)=0} P_{M_1}[\tau|\pi] R_{M_1}(\tau) \\
&\le \sum_{\tau:\text{escape}_m(\tau)=1} P_{M_1}[\tau|\pi](R_{M_1}(\text{pre}_m(\tau)) + V_{max}) + \sum_{\tau:\text{escape}_m(\tau)=0} P_{M_1}[\tau|\pi] R_{M_1}(\tau) \\
&\le \sum_{\text{pre}_m(\tau)} P_{M_1}[\text{pre}_m(\tau)|\pi](R_{M_1}(\text{pre}_m(\tau)) + V_{max}) + \sum_{\tau:\text{escape}_m(\tau)=0} P_{M_1}[\tau|\pi] R_{M_1}(\tau)
\end{aligned}$$

We then find the lower bound of $J_{M_2}(\pi)$,

$$\begin{aligned}
&J_{M_2}(\pi) \\
&= \sum_{\tau:\text{escape}_m(\tau)=1} P_{M_2}[\tau|\pi](R_{M_2}(\text{pre}_m(\tau)) + R_{M_2}(\text{suf}_m(\tau))) + \sum_{\tau:\text{escape}_m(\tau)=0} P_{M_2}[\tau|\pi] R_{M_2}(\tau) \\
&\ge \sum_{\tau:\text{escape}_m(\tau)=1} P_{M_2}[\tau|\pi](R_{M_2}(\text{pre}_m(\tau)) + 0) + \sum_{\tau:\text{escape}_m(\tau)=0} P_{M_1}[\tau|\pi] R_{M_2}(\tau) \\
&\ge \sum_{\text{pre}_m(\tau)} P_{M_2}[\text{pre}_m(\tau)|\pi](R_{M_2}(\text{pre}_m(\tau))) + \sum_{\tau:\text{escape}_m(\tau)=0} P_{M_2}[\tau|\pi] R_{M_2}(\tau)
\end{aligned}$$

Note that when $\text{escape}_m(\tau) = 0$, $P_{M_1}[\tau|\pi] = P_{M_2}[\tau|\pi]$ and $R_{M_1}(\tau) = R_{M_2}(\tau)$. Similarly, $P_{M_1}[\text{pre}_m(\tau)|\pi] = P_{M_2}[\text{pre}_m(\tau)|\pi]$. Thus we have,

$$\begin{aligned}
J_{M_1}(\pi) - J_{M_2}(\pi) &\le \sum_{\text{pre}_m(\tau)} P_{M_1}[\text{pre}_m(\tau)|\pi] V_{max} \\
&\le P_{M_1}[\text{escape}_m(\tau)|\pi] V_{max}
\end{aligned}$$

□

A.5 Sample Complexity Analysis

Event A.30. For all stationary policy π, timesteps t and states s during the execution of R_{max} on some MDP M, the discretized empirical known state-action MDP ($\hat{M}_{d,m_t}$) is ε_d-close to the value in true discretized state-action MDP (M_{d,m_t}).

$$|V^{\pi}_{M_{d,m_t}}(\hat{s}) - V^{\pi}_{\hat{M}_{d,m_t}}(\hat{s})| \le \varepsilon_d$$

where $\varepsilon_d \in \mathbb{R}^+$ is the distance between the true known discretized state-action MDP (M_{d,m_t}) and the discretized empirical known state-action MDP ($\hat{M}_{d,m_t}$), which depends on the discretization radius λ.

Lemma A.31. *There exists a constant C such that if R_{max} with parameters m and ε_d is executed on any MDP $M = \langle \mathcal{S}, \mathcal{A}^i, \mathcal{T}, \mathcal{R}^i, \gamma \rangle$ and m satisfies*

$$m \ge C V_{max}^2 \frac{|\hat{\mathcal{S}}| + \ln \frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta}}{(\varepsilon_d(1-\hat{\gamma}))^2}$$

Event A.30 will occur with probability at least $1 - \delta$.

Proof. Event 1 occurs if R_{max} maintains a close approximation of its known state-action MDP. By Theorem A.21 and Theorem A.25, it is sufficient to obtain $C\frac{\varepsilon_d(1-\hat{\gamma})}{V_{max}}$-approximate transition and reward functions for any constant C and discretized meta-state-action pairs in $K_{d,t}$. Since transition and reward functions of R_{max} are maximum-likelihood estimates with first m samples for each $(\hat{s}_d, \hat{a}_d) \in K$. As long as m is large enough, it is highly probable that the empirical estimates of discretized meta-state-action pairs will be accurate.

Assuming $\varepsilon^R_{m,d} \leq C\frac{\varepsilon_d(1-\hat{\gamma})}{V_{max}}$ and solving for m:

$$\begin{aligned}
\frac{1}{2}\sqrt{\frac{2}{m}\ln\frac{4|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta}} &\leq C\frac{\varepsilon_d(1-\hat{\gamma})}{V_{max}} \\
\frac{2}{m}\ln\frac{4|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta} &\leq (C\frac{\varepsilon_d(1-\hat{\gamma})}{V_{max}})^2 \\
\frac{m}{2} &\geq \frac{1}{(C\frac{\varepsilon_d(1-\hat{\gamma})}{V_{max}})^2}\ln\frac{4|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta} \\
m &\geq \frac{CV^2_{max}}{(\varepsilon_d(1-\hat{\gamma}))^2}\ln\frac{4|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta}
\end{aligned}$$

Assuming $\varepsilon^T_{m,d} \leq C\frac{\varepsilon_d(1-\hat{\gamma})}{V_{max}}$ and solving for m:

$$\begin{aligned}
\sqrt{\frac{2[\ln(2^{|\hat{\mathcal{S}}|}-2)+\ln\frac{2|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta}]}{m}} &\leq C\frac{\varepsilon_d(1-\hat{\gamma})}{V_{max}} \\
\frac{2[\ln\frac{(2^{|\hat{\mathcal{S}}|}-2)(2|\hat{\mathcal{S}}||\hat{\mathcal{A}}|)}{\delta}]}{m} &\leq (C\frac{\varepsilon_d(1-\hat{\gamma})}{V_{max}})^2 \\
m &\geq \frac{CV^2_{max}}{(\varepsilon_d(1-\hat{\gamma}))^2}\ln\frac{(2^{|\hat{\mathcal{S}}|}-2)(2|\hat{\mathcal{S}}||\hat{\mathcal{A}}|)}{\delta}
\end{aligned}$$

When $m \geq CV^2_{max}\frac{\hat{\mathcal{S}}+ln(\hat{\mathcal{S}}\hat{\mathcal{A}}/\delta)}{(\varepsilon_d(1-\hat{\gamma}))^2}$, both conditions are satisfied for any constant $C > 0$. □

Theorem A.32. *For $\varepsilon_d \in \{0,1\}$, $\delta \in \{0,1\}$, $\lambda \in \mathbb{R}^+$, M be any MDP, there exists some constants $C > 0$ and inputs $m = m(\frac{1}{\varepsilon}, \frac{1}{\delta}, \frac{1}{\lambda})$ such that if R_{max} is executed on M, the following holds:*
Let $\pi^*_{M_m}$ be R_{max}'s policy, with probability at least $1-\delta$, $J_{M_d}(\pi^*_{M_d}) - J_{M_d}(\pi^*_{\hat{M}_{d,m}}) \leq 2\varepsilon_d$ is true for all but

$$\mathcal{O}(\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{C\varepsilon_d(\varepsilon_d(1-\hat{\gamma}))^2}V^3_{max}(|\hat{\mathcal{S}}| + \ln\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta})\ln\frac{1}{\delta})$$

episodes.

Proof.

$$\begin{aligned}
J_{M_d}(\pi^*_{M_d}) - J_{M_d}(\pi^*_{\hat{M}_{d,m}}) &\leq J_{M_{d,m}}(\pi^*_{M_d}) - J_{M_d}(\pi^*_{\hat{M}_{d,m}}) && \text{(optimism)} \\
&\leq J_{M_{d,m}}(\pi^*_{M_{d,m}}) - J_{M_d}(\pi^*_{\hat{M}_{d,m}}) && \text{(greedy policy)} \\
&\leq J_{M_{d,m}}(\pi^*_{M_{d,m}}) - J_{\hat{M}_{d,m}}(\pi^*_{\hat{M}_{d,m}}) + J_{\hat{M}_{d,m}}(\pi^*_{\hat{M}_{d,m}}) - J_{M_d}(\pi^*_{\hat{M}_{d,m}}) \\
&\leq \frac{\varepsilon^R_{m,d}}{1-\hat{\gamma}} + \frac{\hat{\gamma}\varepsilon^T_m R_{max}}{2(1-\hat{\gamma})^2} + J_{\hat{M}_{d,m}}(\pi^*_{\hat{M}_{d,m}}) - J_{M_d}(\pi^*_{\hat{M}_{d,m}}) && \text{(by Theorem A.28)} \\
&\leq 2\frac{\varepsilon^R_{m,d}}{1-\hat{\gamma}} + \frac{\gamma\varepsilon^T_m \hat{R}_{max}}{(1-\hat{\gamma})^2} + J_{M_{d,m}}(\pi^*_{\hat{M}_{d,m}}) - J_{M_d}(\pi^*_{\hat{M}_{d,m}}) && \text{(by Theorem A.27)} \\
&\leq 2\frac{\varepsilon^R_{m,d}}{1-\hat{\gamma}} + \frac{\hat{\gamma}\varepsilon^T_m R_{max}}{(1-\hat{\gamma})^2} + V_{max} \cdot P_M[escape(\tilde{\tau}|\pi)] && \text{(by Theorem A.29)}
\end{aligned}$$

For the $P_M[escape(\tilde{\tau}|\pi)]$ term,

1. If $P_M[escape(\tilde{\tau}|\pi)] < \frac{\varepsilon_d}{V_{max}}$: the policy is $2\varepsilon_d$-optimal.
2. If $P_M[escape(\tilde{\tau}|\pi)] \geq \frac{\varepsilon_d}{V_{max}}$: Successful exploration occurs at most $m|\hat{\mathcal{S}}||\hat{\mathcal{A}}|$ times and with Chernoff-Hoeffding bound (Theorem A.20), we observe $h = m|\hat{\mathcal{S}}||\hat{\mathcal{A}}|$ or more successful exploration (heads) with a probability at least $1-\delta$, after W episodes (tosses), where

$$\begin{aligned} W &= \mathcal{O}(\frac{m|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{P_M[escape(\tilde{\tau}|\pi)]}\ln\frac{1}{\delta}) \\ &= \mathcal{O}(\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\varepsilon_d^3(1-\hat{\gamma})^2}V_{max}^3(|\hat{\mathcal{S}}| + \ln\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta})\ln\frac{1}{\delta}) \quad \square \end{aligned}$$

Sample complexity analysis

The expression in Item 2 quantifies the sample complexity of a R_{max} algorithm, where the states and actions are discretized to a radius λ. To find the sample complexity in the M-FOS context, we need to express $|\hat{\mathcal{S}}|$ and $|\hat{\mathcal{A}}|$ in terms of the inner-game state space size $|\mathcal{S}|$ and inner-game action space size $|\mathcal{A}|$.

Let $M = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma \rangle$ be a t-step MDP in the inner-game. An inner agent's Q table has $|\mathcal{S}| \times |\mathcal{A}|$ entries, and there are infinite possible values that an entry can theoretically take. We can get the bound of the inner-game Q-value: $Q(s,a) \in (0, \frac{R_{max}}{1-\gamma})$. After discretizing the Q-values to radius λ, an entry can take $\frac{R_{max}}{\lambda(1-\gamma)}$ possible values. Thus, the cardinality of an agent's discretized Q-table is $(\frac{R_{max}}{\lambda(1-\gamma)})^{|\mathcal{S}||\mathcal{A}|}$.

Assuming there are n agents in the game, $R_{max} = 1$, $\hat{\gamma} = \gamma$ and $\varepsilon_d = \varepsilon + \sqrt{n+1}\mathcal{L}\alpha$,

$$|\hat{\mathcal{S}}| = |\phi^{-\mathbf{i}}| = (\frac{1}{\lambda(1-\gamma)})^{n|\mathcal{S}||\mathcal{A}|} \qquad , \qquad |\hat{\mathcal{A}}| = |\phi^i| = (\frac{1}{\lambda(1-\gamma)})^{|\mathcal{S}||\mathcal{A}|}$$

which gives us the final expression of the sample complexity of M-FOS, implemented in case 1:

$$\begin{aligned} &\mathcal{O}(\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\varepsilon_d^3(1-\hat{\gamma})^2}V_{max}^3(|\hat{\mathcal{S}}| + \ln\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta})\ln\frac{1}{\delta}) \\ &\sim \mathcal{O}(\frac{(\frac{1}{\lambda(1-\gamma)})^{2n|\mathcal{S}||\mathcal{A}|}(\frac{1}{\lambda(1-\gamma)})^{|\mathcal{S}||\mathcal{A}|}}{\varepsilon_d^3(1-\hat{\gamma})^2}\frac{1}{(1-\gamma)^3}\ln\frac{1}{\delta}) \\ &\sim \mathcal{O}(\frac{(\frac{1}{\lambda(1-\gamma)})^{(2n+1)|\mathcal{S}||\mathcal{A}|}}{\varepsilon_d^3(1-\hat{\gamma})^5}\ln\frac{1}{\delta}) \end{aligned} \tag{16}$$

Note that this term drops the logarithmic factor $\ln\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta}$ and is expressed in terms of ε_d for simplicity.

A.6 Case 2 - Past trajectories

The second case only differs from the first by the meta-state representation, where $\hat{S}$ is the trajectory of all agents τ. The meta-action $\hat{A}$ is still our agent's discretized policy ϕ^i, which we discretize with radius λ. The sample complexity proof remains mostly the same, except that only the meta-action is discretized. Thus the proofs below are based on Section 4.1 and only changes due to the different representation will be displayed.

Discretization Setup:

We define the meta-reward after discretization as

$$R(\hat{s}, \hat{a}_d) = \frac{1}{\sum_{\hat{a}_d-\alpha}^{\hat{a}_d+\alpha} n(\hat{s},\hat{a})}\sum_{\hat{a}_d-\alpha}^{\hat{a}_d+\alpha}\hat{r}(\hat{s},\hat{a}) \quad where \quad \hat{a} \in [\hat{a}_d - \alpha, \hat{a}_d + \alpha]$$

Assuming α is small enough such that $\forall \hat{s}_d \in \hat{\mathcal{S}}_d$ and $\forall \hat{a}_d \in \hat{\mathcal{A}}_d$,

$$\begin{aligned} |R(\hat{s},\hat{a}) - R(\hat{s},\hat{a}_d)| &< \max_{\hat{a}} \big\|(\hat{s},\hat{a}) - (\hat{s},\hat{a}_d)\big\|_2 \\ &< \max_{\hat{a}} (\hat{a} - \hat{a}_d) \\ &< \alpha \end{aligned} \tag{17}$$

Sample complexity analysis:
Since the Lemmas and Theorems used to prove case 1 and case 2 are the same, by observation, we replace the factor of $\boldsymbol{\sqrt{n+1}}$ in case 1 by $\mathbf{1}$ in case 2. This thus gives us the expression for m and sample complexity W for constants $C > 0$:

$$m \geq C V_{max}^2 \frac{\hat{\mathcal{S}} + ln(\hat{\mathcal{S}}\hat{\mathcal{A}}/\delta)}{\varepsilon_d^3 (1-\hat{\gamma})^2} \tag{18}$$

$$\begin{aligned} W &= \mathcal{O}(\frac{m|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{P_M[escape(\tilde{\tau}|\pi)]} \ln\frac{1}{\delta}) \\ &= \mathcal{O}(\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\varepsilon_d^3(1-\hat{\gamma})^2} V_{max}^3 (|\hat{\mathcal{S}}| + \ln\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta}) \ln\frac{1}{\delta}) \end{aligned} \tag{19}$$

Sample complexity in the M-FOS context

Let $M = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma \rangle$ be a t-step MDP in the inner-game. Again, we express $|\hat{\mathcal{S}}|$ and $|\hat{\mathcal{A}}|$ in terms of the inner-game state space size $|\mathcal{S}|$ and inner-game action space size $|\mathcal{A}|$. There are still $(\frac{R_{max}}{\lambda(1-\gamma)})^{|\mathcal{S}||\mathcal{A}|}$ different combinations for one agent's discretized Q-table.

Assuming there are n agents in the game, $R_{max} = 1$ and $\hat{\gamma} = \gamma$. In the M-FOS paper, the meta-state is the trajectory in the inner game of all agents, i.e. $\tau = \{\boldsymbol{s}_1, \boldsymbol{a}_1, \boldsymbol{r}_1, \cdots, \boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{r}_t, \}$. However, this leads to enormous combinations and is infeasible to implement in R_{max} (which requires a transition matrix of size $|\hat{\mathcal{S}}| \times |\hat{\mathcal{A}}| \times |\hat{\mathcal{S}}|$). Thus, we feed in limited history as the meta-state, i.e. $\hat{s}_t = \tau_h(t) = \{\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{r}_t, \cdots, \boldsymbol{s}_{t+h}, \boldsymbol{a}_{t+h}, \boldsymbol{r}_{t+h}, \}$, the number of combination of a state-action-reward tuple is $|\boldsymbol{sar}| = (|\mathcal{S}||\mathcal{A}||\mathcal{R}|)^n$. Thus,

$$|\hat{\mathcal{S}}| = |\boldsymbol{\tau}_h| = |\boldsymbol{sar}|^h = (|\mathcal{S}||\mathcal{A}||\mathcal{R}|)^{nh} = (|\mathcal{S}||\mathcal{A}|)^{nh}(\frac{1}{\lambda(1-\gamma)})^{nh}$$

$$|\hat{\mathcal{A}}| = |\phi^i| = (\frac{1}{\lambda(1-\gamma)})^{|\mathcal{S}||\mathcal{A}|}$$

which gives us the final expression of the sample complexity of M-FOS, implemented in case 2:

$$\begin{aligned} &\mathcal{O}(\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\varepsilon_d^3(1-\hat{\gamma})^2} V_{max}^3 (|\hat{\mathcal{S}}| + \ln\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta}) \ln\frac{1}{\delta}) \\ &\sim \mathcal{O}(\frac{(|\mathcal{S}||\mathcal{A}|)^{2nh}(\frac{1}{\lambda(1-\gamma)})^{2nh}(\frac{1}{\lambda(1-\gamma)})^{|\mathcal{S}||\mathcal{A}|}}{\varepsilon_d^3(1-\hat{\gamma})^2} \frac{1}{(1-\gamma)^3} \ln\frac{1}{\delta}) \\ &\sim \mathcal{O}(\frac{(|\mathcal{S}||\mathcal{A}|)^{2nh}(\frac{1}{\lambda(1-\gamma)})^{2nh+|\mathcal{S}||\mathcal{A}|}}{\varepsilon_d^3(1-\hat{\gamma})^5} \ln\frac{1}{\delta}) \end{aligned}$$

Note that this term drops the logarithmic factor $\ln\frac{|\hat{\mathcal{S}}||\hat{\mathcal{A}}|}{\delta}$ and is expressed in terms of ε_d for simplicity.

B Experiments

B.1 Pseudo-code

Algorithm 2 Adapetd M-FOS Algorithm with an R_{max} meta-agent

```
Meta-game Inputs: $\hat{\mathcal{S}}, \hat{\mathcal{A}}, \hat{\gamma}, m, \varepsilon, U(\cdot,\cdot), \hat{T}, \hat{K}$        Inner-game Inputs: $\mathcal{S}, \mathcal{A}, \gamma, T$
Initialisation: $\hat{Q}(s,a) \leftarrow U(s,a),\ \hat{r}(\hat{s},\hat{a}) \leftarrow 0,\ v(\hat{s},\hat{a}) \leftarrow 0,\ v(\hat{s},\hat{a},\hat{s}') \leftarrow 0$
for meta-episode $n = 0$ to $\hat{J}$ do
    Reset environment
    for meta-time step $t = 1$ to $\hat{K}$ do
        Sample $a^{-i} = \varepsilon\text{-}greedy(\phi_{\mathbf{t}}^{-\mathbf{i}})$
        Run inner game of length T and collect trajectory $[\boldsymbol{\tau}_{\hat{\boldsymbol{t}}}^{-\boldsymbol{i}}, \tau_{\hat{t}}^{i}]$
        $\hat{a} = a^i$
        $\hat{r}_{\hat{t}} = r^i + \hat{\gamma}\hat{r}_{\hat{t}-1}$
        $\hat{s}' = [\boldsymbol{\tau}_{\hat{\boldsymbol{t}}}^{-\boldsymbol{i}}, \tau_{\hat{t}}^{i}]$
        Let $\hat{s'}$ be the next meta-state after executing meta-action $\hat{a}$ from meta-state $\hat{s}$
        if $v(\hat{s},\hat{a}) < m$ then
            $\hat{r}(\hat{s},\hat{a}) \leftarrow \hat{r}(\hat{s},\hat{a}) + R_{\hat{t}}^{i}$
            $v(\hat{s},\hat{a}) \leftarrow v(\hat{s},\hat{a}) + 1$                {Increment visitation frequency}
            $v(\hat{s},\hat{a},\hat{s'}) \leftarrow v(\hat{s},\hat{a},\hat{s'}) + 1$
            if $v(\hat{s},\hat{a}) = m$ then
                for $i = 1, 2, 3, \cdots, \lceil \frac{ln(\frac{1}{\varepsilon(1-\gamma)})}{1-\gamma} \rceil$ do
                    Let $(\bar{\hat{s}}, \bar{\hat{a}})$ be the STATE-action pair that has been visited FOR at least m times
                    for all $(\bar{\hat{s}}, \bar{\hat{a}})$ do
                        if $v(\bar{\hat{s}}, \bar{\hat{a}}) \geq m$ then
                            $\hat{R}(\bar{\hat{s}}, \bar{\hat{a}}) := \frac{1}{v(\bar{\hat{s}}, \bar{\hat{a}})}\hat{r}(\bar{\hat{s}}, \bar{\hat{a}})$
                            $\hat{T}(\hat{s}' | \bar{\hat{s}}, \bar{\hat{a}}) := \frac{v(\bar{\hat{s}}, \bar{\hat{a}}, \hat{s}')}{v(\bar{\hat{s}}, \bar{\hat{a}})}$
                            $\hat{Q}(\bar{\hat{s}}, \bar{\hat{a}}) \leftarrow \hat{R}(\bar{\hat{s}}, \bar{\hat{a}}) + \hat{\gamma} \sum_{s'} \hat{T}(\hat{s}' | \bar{\hat{s}}, \bar{\hat{a}}) \max_{\hat{a'}} \hat{Q}(\hat{s}', \hat{a}')$
        $\hat{s} \leftarrow \hat{s}'$
```

C Limitations

C.1 Choice of meta-agent

R_{max} is an intuitive and simple algorithm to implement as a meta-learner and to quantify the sample complexity of M-FOS. However, it comes with a number of limitations:

- *Tabular RL:* R_{max} requires finite and discrete state and action space. In practical RL scenarios, the state space is often infinite, creating a need for discretization. This limits the algorithm's effectiveness in achieving an optimal policy.
- *Linear function approximator:* There is a discrepancy between theoretical and empirical work in RL. Most theoretical literature assumed a linear MDP and approximate transitions and rewards with linear functions to simplify the problem statement. For example, R_{max} approximates the transitions and rewards function linearly using maximum likelihood estimation. However, empirical evidence shows that non-linear function approximators like neural networks or deep learning perform better due to their complexity. R_{max} thus cannot capture the dynamics in a high-dimensional environment as efficiently as a non-linear function approximator can.
- *Model-based RL:* R_{max} is an MBRL algorithm with some limitations compared to an MFRL algorithm. MBRL relies on accurate models of the environment, which is challenging to construct in high-dimensional or stochastic environments. Thus, it is faintly applicable to the multi-agent opponent-shaping goal that M-FOS is designed for.

These limitations hinder the empirical and theoretical investigation of the sample complexity of M-FOS. Ideally, we aim to develop a method for quantifying the sample complexity of actor-critic or policy-gradient methods to bridge the gap between theoretical and empirical knowledge.