Empirical Game-Theoretic Analysis for Mean Field Games

Yongzhao Wang
University of Michigan
Ann Arbor, USA
wangyzh@umich.edu

Michael P. Wellman
University of Michigan
Ann Arbor, USA
wellman@umich.edu

ABSTRACT

We present a simulation-based approach for solution of mean field games (MFGs), using the framework of *empirical game-theoretical analysis* (EGTA). Our primary method employs a version of the double oracle, iteratively adding strategies based on best response to the equilibrium of the *empirical MFG* among strategies considered so far. We present *Fictitious Play* (FP) and *Replicator Dynamics* as two subroutines for computing the empirical game equilibrium. Each subroutine is implemented with a query-based method rather than maintaining an explicit payoff matrix as in typical EGTA methods due to a representation issue we highlight for MFGs. We test the performance of our primary method in various games and show that it outperforms directly applying FP to MFGs with either subroutine. By introducing game model learning and regularization, we significantly improve the sample efficiency of the primary method without sacrificing the overall learning performance. Theoretically, we prove that a Nash equilibrium (NE) exists in the empirical MFG and show the convergence of iterative EGTA to NE of the full MFG.

KEYWORDS

Mean Field Games; Empirical Game-Theoretic Analysis; Game Model Learning

ACM Reference Format:
Yongzhao Wang and Michael P. Wellman. 2023. Empirical Game-Theoretic Analysis for Mean Field Games. In *Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), London, United Kingdom, May 29 – June 2, 2023*, IFAAMAS, 9 pages.

1 INTRODUCTION

Mean field games (MFGs) [7, 9] model strategic interactions among a conceptually infinite number of players. Aggregate player behavior is summarized by a distribution over states of the population, which constitutes the *mean field*. The MFG framework reduces analysis to the characterization of the optimal behavior of a single representative player in its interactions with the full population, as represented by the mean field. By considering the limit case of populations with mean-field structure, this framework can support game-theoretic analysis that would be intractable for a standard corresponding model of a game among a large but finite number of players. One common interpretation of MFG models is as an approximation of an actually finite many-player game.

The complexity of analytic solution of MFGs grows with the state space, necessitating numerical approaches for all but the simplest problems. Several such algorithms for MFGs have recently been proposed [12, 14, 15]. As befits the game nature of MFGs, some of these employ standard online game-solving algorithms, such as fictitious play (FP) [2, 16].

The methodology of *empirical game-theoretic analysis* (EGTA) [20, 24] represents a general approach for building and reasoning about game models based on simulation. EGTA works by estimating game models over restricted strategy sets, typically identified through a heuristic process as being strategically salient for analysis. The estimated game models are called *empirical games*. An iterative form of EGTA is presented by Lanctot et al. [8] as the *Policy Space Response Oracle* (PSRO) algorithm, where at each iteration, new strategies are generated through reinforcement learning (RL). There is a considerable prior literature that demonstrates the effectiveness of PSRO and variants in a range of applications [3, 22, 26].

We develop an iterative framework for numerically solving MFGs with EGTA. Under the framework, we first introduce a primary method to demonstrate the basic implementation of the framework, and then we apply game model learning (GML) and regularization to the primary method, which significantly improve the sample efficiency of the primary method without sacrificing the overall learning performance. Our primary method extends the double oracle method [10] to MFGs, iteratively adding strategies based on best response to the equilibrium of the *empirical MFG* among strategies considered so far. We propose FP and replicator dynamics (RD) as two subroutines for computing a Nash equilibrium (NE) of the empirical MFG and demonstrate that both subroutines are effective for the empirical game analysis. Each subroutine is implemented with a query-based method, in which we query the utilities of different strategies through simulations as needed rather than maintaining an explicit payoff matrix as in typical EGTA methods. This tweak is caused by the non-linearity of the MFG utility function in the population distribution, which we highlight for MFGs. We test our primary iterative EGTA framework in MFGs with various configurations and demonstrate the improved learning performance of EGTA over directly applying FP [15] to MFGs.

Despite its effectiveness for solving MFGs, the primary method requires a large number of utility simulations due to the non-linearity of the utility function in the population distribution, which we refer to as low sample efficiency in EGTA. To improve the sample efficiency, we introduce a GML approach and a regularized subroutine to our iterative EGTA framework. The GML approach is a form of regression that learns the utility function progressively over EGTA iterations. With a learned utility function, utility information can be predicted, and thus reducing the number of queries to the simulator. The regularized subroutine improves sample efficiency through reducing the number of iterations of subroutines in each EGTA iteration. In a separate work [23], we have shown that properly regularizing the best response target (i.e., not best-responding to an exact equilibrium) will lead to an improved learning performance for EGTA and the regularization can be achieved by early

Proc. of the 22nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2023), A. Ricci, W. Yeoh, N. Agmon, B. An (eds.), May 29 – June 2, 2023, London, United Kingdom. © 2023 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

stopping a subroutine within each EGTA iteration. For our purposes, early stopping a subroutine means less utility queries for each iteration of EGTA and improved sample efficiency if the overall learning performance would not decline. By introducing GML and regularization, we demonstrate a significant improvement on the sample efficiency (i.e., EGTA with GML and regularization only requires 1/6 of simulations needed by the primary method) over a variety of MFG configurations.

The theoretical results of this work are twofold. First, we prove the existence of NE in an empirical MFG with a restricted strategy space under a mild assumption, assuming the MFG is fully symmetric. Second, we prove that the iterative EGTA converges to NE of the full game if the best response target is NE across iterations and an exact best-response oracle is available.

Contributions of this study include:

(1) We propose an iterative EGTA framework for solving MFGs with FP and RD as two practical subroutines for the empirical game analysis.
(2) By introducing GML and regularization, we significantly improve the sample efficiency of the iterative EGTA framework.
(3) We demonstrate the effectiveness of approaches for solving MFGs in various configurations.
(4) Theoretically, we prove the existence of NE in an empirical MFG among a restricted set of strategies under a mild assumption and prove the convergence of the iterative EGTA to NE of the full MFG.

2 RELATED WORK

2.1 Learning Mean Field Games

There is a large prior literature on learning solutions of MFGs. Here we survey few that are closely related to our approach. Elie et al. [4] first studied the convergence of approximate discrete-time FP in MFGs. Perrin et al. [15] further proved the convergence rate of continuous-time FP in MFGs and extended the study to MFGs with common noise. Moreover, FP for MFGs with finite time horizons was demonstrated effective in their work. Perolat et al. [12] criticized that FP is not scalable to MFGs with large state spaces and proposed Online Mirror Descent (OMD) as a solution. They empirically showed that OMD converges significantly faster than FP in MFGs with a large number of states. In the aforementioned algorithms, a fixed initial distribution of the population is required. Perrin et al. [13] argued that a fixed initial distribution restricts the practical applications of MFGs since a real initial distribution could be different from the one used for training. They proposed a learning algorithm to learn a *Master Policy*, which takes the distribution of population as input and thus taking the initial distribution into consideration. They demonstrated the ability of generalization of the learned master policy.

In a recent work that is simultaneous with and independent of our **primary method**, Muller et al. [11] adapted PSRO to MFGs and analyzed convergence properties with various solution concepts. Both theirs and our primary method are based on an iterative EGTA/PSRO framework for MFGs, highlighting the issue that the utility function for MFGs is not generally linear in the distribution. Theoretically, both works prove the existence of NE in the empirical game with restricted strategy set and the convergence of iterative EGTA/PSRO to NE. What is unique to their work is that they investigate the modifications for PSRO to converge to (coarse) correlated equilibria in MFGs as well as the theoretical counterparts. The contribution of our primary method focuses on practical techniques for the empirical game analysis and goes beyond their work in including full details of how online learning algorithms (i.e., FP and RD as subroutines) realize the framework of EGTA for MFGs. We also include more experimental results on performance of these methods for deriving approximate equilibria for MFGs. Since both theirs and our primary method rely on either black-box optimization or online approaches for computing NE of intermediate empirical games, the issue of low sample efficiency exists. As a major contribution of our work, our GML and regularization successfully addresses this issue.

2.2 Game Model Learning

To the best of our knowledge, our work is the first study that combines GML with the iterative EGTA framework. Here we briefly survey prior work on GML that assumes the full game is given. Vorobeychik et al. [21] first introduced the concept of learning normal-form game models as utility function regression from simulation data sampled over continuous strategy spaces. They demonstrated the approach using single-parameter strategy representation. Ficici et al. [6] clustered a large number of players into two roles based on data consisting of strategy profiles and utilities. Then regression of the utility function was applied for each role. Wiedenbeck et al. [25] deployed Gaussian process regression to learn the utility function of large symmetric games. Sokota et al. [18] extended GML to role-symmetric games by regressing the deviation payoff function rather than the payoff function.

3 PRELIMINARIES

We study MFGs with a temporal structure, where representative players interact with populations over a shared state space X, action space A and time horizon T. A *multi-population mean field game* (MP-MFG) in normal form is given by $\mathcal{G} = ([N_p], (S_i), (u_i))$. $[N_p] = \{1, \ldots, N_p\}$ is a set of N_p populations indexed by i. Each population corresponds to a conceptually infinite and interchangeable set of players playing a particular role in the game. S_i denotes the set of strategies of population i and u_i denotes the utility function. State space X, action space A and time horizon T are encoded in the definitions of a strategy $s_i \in S_i$ and the utility function u_i as described below.

A strategy of the representative player of population i at time t, $s_{i,t}$, maps from state space X to the space of action distributions $\Delta(A)$, where Δ represents the probability simplex over a set. The overall strategy $s_i = (s_{i,t})_{t\in[0,T-1]} \in S_i$ is a sequence of strategies from time 0 through horizon T.

The behavior of each population is summarized by a mean field, defined as distributions over an underlying state space X of the game environment. Denote distributions of all populations over time horizon T as $\mu = \{\mu_1, \ldots, \mu_{N_p}\}$, where $\mu_i = (\mu_{i,t})_{t\in[0,T]} \in \Delta(X)^{T+1}$.

Utility functions $u_i : S_i \times \Delta(X)^{N_p \times (T+1)} \to \mathbb{R}$ define the payoff to a representative player of population $i \in [N_p]$ playing its strategy against the distributions of all populations μ (i.e., mean fields).

A mixed strategy σ_i is a probability distribution over strategies in S_i, with $\sigma_i(s_i)$ denoting the probability the representative player of population i plays strategy s_i. Let σ be the profile of strategies across populations (i.e., $\sigma = (\sigma_1, \ldots, \sigma_{N_p})$).

The representative player of population i's *best response* to population distributions μ is any strategy yielding maximum payoff for the player, holding the distributions μ constant:

$$br_i(\mu) = \operatorname*{argmax}_{\sigma_i \in \Delta(S_i)} u_i(\sigma_i, \mu).$$

Let $br(\mu) = \prod_{i \in N_p} br_i(\mu)$ be the overall best-response correspondence for populations' distributions μ. Here $\prod$ is the Cartesian product. A Nash equilibrium for an MFG is a profile σ^* such that $\sigma^* \in br(\mu^*)$, where μ^* is induced by σ^*.

A distribution μ_i is said to be induced by σ_i, denoted as $\mu_i^{\sigma_i}$, following the *Forward Equation*, that is, given initial distribution $\mu_{i,0}$, for $t \in [0, T-1]$ and all $x_i' \in X$,

$$\mu_{i,t+1}^{\sigma_i}(x_i') = \sum_{x_i, a_i \in X, A} \mu_{i,t}^{\sigma_i}(x_i) \sigma_{i,t}(a_i \mid x_i) p(x_i' \mid x_i, a_i), \qquad (1)$$

where $p : X \times A \to \Delta X$ is the transition function.

The representative player of population i's *regret* in profile σ given distributions μ in game $\mathcal{G}$ is given by

$$\rho_i^{\mathcal{G}}(\sigma_i, \mu) = \max_{s_i \in S_i} u_i(s_i, \mu) - u_i(\sigma_i, \mu).$$

Regret captures the maximum the representative player of population i can gain in expectation by unilaterally deviating from its mixed strategy in σ to an alternative strategy in S_i, given distributions μ. A NE strategy profile has zero regret for each representative player. A profile is said to be an ϵ-Nash equilibrium (ϵ-NE) if no representative player can gain more than ϵ by unilateral deviation. The regret of a strategy profile σ is defined as the sum over representative players' regrets:

$$\rho^{\mathcal{G}}(\sigma, \mu) = \sum_{i \in [N_p]} \rho_i^{\mathcal{G}}(\sigma_i, \mu).$$

A *restricted game* $\mathcal{G}_{S \downarrow X}$ is a projection of full game $\mathcal{G}$, in which populations choose from restricted strategy sets $\Lambda_i \subseteq S_i$. An *empirical game* $\hat{\mathcal{G}}$ is a model of true game $\mathcal{G}$ where payoffs are estimated through simulation. Thus, $\hat{\mathcal{G}}_{S \downarrow X} = (N, (\Lambda_i), (\hat{u}_i))$ denotes an empirical game model where $\hat{u}$ is an estimated projection of u onto the strategy space Λ.

4 ITERATIVE EGTA FOR MFGS

4.1 Framework

In finite games, an iterative approach for computing NE is presented by the well-known Double Oracle (DO) algorithm [10], which alternates between analysis of a current empirical game and adding strategies that best respond to a current equilibrium. We extend DO to MFGs and provide a game-solving algorithm for MP-MFGs in Algorithm 1 as our primary EGTA method for MFGs.

In Algorithm 1, for each population i, the representative player is initialized with a strategy $s_{i,0}$ and an initial distribution $\mu_{i,0}$. At each iteration τ, an equilibrium σ^e of the current empirical game is computed by a subroutine (e.g., FP or RD) and then distributions μ^e are induced by σ^e through the forward equation. Then for each population i, the representative player computes an exact/approximate best response strategy $s_{i,\tau}$ to the distribution μ^e and adds it to the corresponding strategy set in the empirical game. This process repeats until no beneficial deviation strategy could be found or certain stopping criterion is satisfied.

Algorithm 1 Iterative EGTA for MP-MPGs (primary method)

```
Input: an initial strategy $s_{i,0}$ and an initial distribution $\mu_{i,0}$ for all population $i \in [N_p]$.
for EGTA iteration $\tau \in \{1, \ldots, \mathcal{T}\}$ do
    Compute $(\sigma^e, \mu^e)$ of the current empirical game by a subroutine
    for $i \in \{1, \ldots, N_p\}$ do
        Compute a best response strategy $s_{i,\tau} \in S_i$ to the empirical equilibrium distribution $\mu^e$
        Add $s_{i,\tau}$ to the strategy set of population $i$: $\Lambda_i \longleftarrow \Lambda_i \bigcup s_{i,\tau}$
    end for
end for
Return $(\sigma^e, \mu^e)$
```

4.2 Analyzing an Empirical MFG

In EGTA, analyzing an intermediate empirical game is crucial for generating effective strategies.[1] Despite an MFG model reduces the analysis to the study of interactions between two parties (i.e., a representative player and the population), computing a NE of an empirical MFG is more than constructing an explicit payoff matrix and then applying a game-solver for two parties as in finite games. Proposition 1 shows that MFGs with a restricted strategy set can only be solved with an explicit payoff matrix under certain restrictive conditions.

Consider an empirical MFG with an explicit payoff matrix representation shown in Table 1. In the empirical game, there are 4 strategies $\Lambda = \{s_0, s_1, s_2, s_3\}$ in the restricted strategy set. Since the game is fully symmetric, the population would act following distributions $\mu = \{\mu_0, \mu_1, \mu_2, \mu_3\}$ induced by corresponding strategies in Λ. In the payoff matrix, the value in entry (j, k), where $j, k \in \{0, 1, 2, 3\}$, is $u(s_j, \mu_k)$, the utility of the representative player playing s_j against the distributions μ_k induced by s_k.

	s_0	s_1	s_2	s_3
s_0	$u(s_0, \mu_0)$	$u(s_0, \mu_1)$	$u(s_0, \mu_2)$	$u(s_0, \mu_3)$
s_1	$u(s_1, \mu_0)$	$u(s_1, \mu_1)$	$u(s_1, \mu_2)$	$u(s_1, \mu_3)$
s_2	$u(s_2, \mu_0)$	$u(s_2, \mu_1)$	$u(s_2, \mu_2)$	$u(s_2, \mu_3)$
s_3	$u(s_3, \mu_0)$	$u(s_3, \mu_1)$	$u(s_3, \mu_2)$	$u(s_3, \mu_3)$

Table 1: Single-population payoff matrix.

Proposition 1. *The NE of the aforementioned payoff matrix will not generally be a NE of the empirical mean field game unless the utility function is linear in μ.*

[1] For discussion simplicity, we assume that players are fully symmetric (i.e., $N_p = 1$) for illustrating the representation issue. So the population index i is dropped when the context is clear. The analysis can be easily extended to MP-MFGs.

Proof. Assume σ is a NE computed from the payoff matrix. According to the definition of NE, we have

$$\sum_{j\in[|\Lambda|]}\sum_{k\in[|\Lambda|]}\sigma(s_j)\sigma(s_k)u(s_j,\mu_k)\geq\sum_{k\in[|\Lambda|]}\sigma(s_k)u(s',\mu_k),\forall s'\in\Lambda. \tag{2}$$

According to the definition of NE σ^* in MFG, we have

$$\sum_{j\in[|\Lambda|]}\sigma^*(s_j)u(s_j,\mu^{\sigma^*})\geq u(s',\mu^{\sigma^*}),\forall s'\in\Lambda, \tag{3}$$

where μ^{σ^*} is induced by σ^*.

By comparing inequalities 2 and 3, to make the NE σ an MFG NE σ^*, the following condition should hold

$$\sum_{k\in[|\Lambda|]}\sigma(s_k)u(s_j,\mu_k)=u(s_j,\mu^{\sigma}),\forall s_j\in\Lambda,$$

indicating the requirement of linearity in μ at least at the equilibrium point.

□

Since the utility function in MFGs will not generally be linear in μ, analysis of an empirical game based on an explicit payoff matrix is impractical. Instead, we rely on query-based methods. We propose FP and RD as two subroutines for solving empirical games and query utility information through simulation as needed.

4.2.1 *FP for Empirical MFGs.*

FP for MFGs has been studied by Elie et al. [4] and Perrin et al. [15]. We adapt it to analyzing an empirical game with a restricted set of strategies. In Algorithm 2, we demonstrate how to apply FP to an empirical game. Specifically, starting from the uniform strategy $\bar{\sigma}$ over the strategies in the restricted strategy set and its induced distribution $\bar{\mu}$, at each iteration $j\in[1,J]$, the representative player of a population i computes a best response strategy $s_{i,j}\in\Lambda_i$ against populations $\bar{\mu}$. The probability in the mixed strategy of playing $s_{i,j}$ is updated by the frequency of $s_{i,j}$ appearing as a best response, that is, the count of being a best response $n_{s_{i,j}}$ divided by the total count up to the j^{th} iteration. Mathematically, for all $i\in[N_p]$ and $k\in[|\Lambda_i|]$, the count of $s_{i,k}$ being a best response $n_{s_{i,k}}$ is incremented by 1 if $s_{i,k}$ is the best response at the current iteration (i.e., $s_{i,k}=s_{i,j}$) and remains the same otherwise.

$$n_{s_{i,k}}=\begin{cases}n_{s_{i,k}}+1 & s_{i,k}=s_{i,j}\\ n_{s_{i,k}} & s_{i,k}\neq s_{i,j}\end{cases}$$

Then we update the probability of every strategy in the mixed strategy by

$$\bar{\sigma}_i(s_{i,k})=\frac{n_{s_{i,k}}}{|\Lambda_i|+\sum_{k\in[|\Lambda_i|]}n_{s_{i,k}}},\forall s_i\in\Lambda_i,i\in[N_p]. \tag{4}$$

To induce a corresponding distribution $\bar{\mu}_i$, we first build a weighted average strategy that is equivalent to the mixed strategy $\bar{\sigma}_i$. Specifically, consider a mixed strategy $\sigma\in\Delta(\Lambda)$ defined on the empirical game with a restricted strategy set Λ. An equivalent strategy $\bar{s}$ is defined as, for each population i,

$$\bar{s}_{i,t}(a\mid x)=\frac{\sum_{k=1}^{|\Lambda_i|}\sigma_i(s_{i,k,t})\mu_t^{s_{i,k,t}}(x)s_{i,k,t}(a\mid x)}{\sum_{k=1}^{|\Lambda_i|}\sigma_i(s_{i,k,t})\mu_t^{s_{i,k,t}}(x)},\forall t\in[0,T-1],$$

Algorithm 2 Fictitious Play as a Subroutine

```
Input: an empirical game. An initial averaged strategy $\bar{\sigma}_i$ over
    strategies in the restricted set $\Lambda_i = (s_{i,1},\dots,s_{i,\tau})$ of population
    $i, \forall i \in [N_p]$ and distributions $\bar{\mu}$ that are induced by $\bar{\sigma}$.
for FP iteration $j \in \{1,\dots,J\}$ do
    for $i \in \{1,\dots,N_p\}$ do
        Compute a best response strategy $s_{i,j} \in \Lambda_i$ against $\bar{\mu}$
        $n_{s_{i,j}} \leftarrow n_{s_{i,j}} + 1$
    end for
    for $i \in \{1,\dots,N_p\}$ do
        Update $\bar{\sigma}_i$ according to Equation 4 and induce $\bar{\mu}_i$
    end for
end for
Return $(\bar{\sigma}, \bar{\mu})$
```

where $s_{i,k,t}$ is the k^{th} strategy of population i at time step t. Then the induced distribution is computed through Equation 1 or estimated through various approaches (e.g., empirical density estimation [15] or generative models [14]).

The empirical game analysis stops until certain stopping criterion is satisfied (e.g., reaching a fixed number of iterations or the difference between the averaged strategies in two consecutive iterations is tiny). FP has been proved for convergence to NE in MFGs with a single population [4, 15]. However, FP will not generally converge to a NE (or even a CE) in MFGs with multiple populations.

4.2.2 *RD for Empirical MFGs.*

Replicator dynamics describes an evolving trajectory of mixed profiles and is commonly employed as a heuristic equilibrium search algorithm in finite games [17, 19]. In Algorithm 3, we adapt RD to our MFG model and propose it as a practical subroutine for empirical game analysis. Similar to RD in finite games, at each iteration the update of a strategy's probability in population i is in proportion to the deviation payoff of that strategy from the average fitness, weighted by its probability from the previous iteration and a learning rate. Theoretically, RD has not been proved for convergence as FP. However, we show that RD exhibits empirical convergence with an even more stable learning manner than FP in our experiments.

Algorithm 3 Replicator Dynamics as a Subroutine

```
Input: an empirical game. An initial averaged strategy $\bar{\sigma}_i$ over
    strategies in the restricted set $\Lambda_i = (s_{i,1},\dots,s_{i,\tau})$ of population
    $i, \forall i \in [N_p]$ and $\bar{\mu}$ is induced by $\bar{\sigma}$. A learning rate $dt$.
for RD iteration $j \in \{1,\dots,J\}$ do
    for $i \in \{1,\dots,N_p\}$ do
        Compute the average fitness $F_i = u_i(\bar{\sigma}_i, \bar{\mu})$
        for $s_i \in \Lambda_i$ do
            Evaluate $u_i(s_i, \bar{\mu})$
            Update $\bar{\sigma}'_i(s_i) \leftarrow \bar{\sigma}_i(s_i) + dt * \bar{\sigma}_i(s_i)[u_i(s_i, \bar{\mu}) - F_i]$
        end for
    end for
    $\bar{\sigma} = \bar{\sigma}'$
    Induce new distribution $\bar{\mu}$ using updated $\bar{\sigma}$.
end for
Return $(\bar{\sigma}, \bar{\mu})$
```

4.3 Convergence to NE

We analyze the theoretical properties of iterative EGTA for MFGs from two aspects: the existence of NE in mean field empirical games and the convergence of iterative ETGA to NE of the full game. We assume that players are fully symmetric (i.e., a single population) and prove that the NE exists in an empirical game under one mild assumption.

Assumption 1. *The utility function $u(s, \mu)$ is continuous in the distribution μ.*

Theorem 1. *Under Assumption 1, for single-population MFGs with finite state and action spaces, there exists a Nash equilibrium in an empirical game.*

According to Theorem 1, an empirical NE exists and hence it can be obtained through some theoretically proven equilibrium search subroutines (e.g., FP). The remaining problem is whether iterative EGTA converges to the NE of the full game.

Theorem 2. *For single-population MFGs with finite state and action spaces, suppose the empirical NE is the best response target at each iteration of iterative EGTA and an exact best response oracle is available, then the empirical NE converges to the NE of the full game.*

We provide the proofs of both theorems in the Appendix (available in the arxiv version: https://arxiv.org/pdf/2112.00900.pdf).

5 GAME MODEL LEARNING AND REGULARIZATION FOR IMPROVED SAMPLE EFFICIENCY

One crucial factor that affects the practicality of our primary iterative EGTA method is sample efficiency. Sample efficiency here refers to the number of simulations needed for estimating the utilities in the equilibrium search at each iteration of EGTA. In finite games, the utilities of any mixed strategy profile can be computed by taking an expectation of utilities across pure strategy profiles in the support, so utility information of pure strategy profiles can be re-used.

Unlike finite games, Proposition 1 shows that the utility $u(s, \mu^{\sigma})$ of playing strategy s against population distributions μ^{σ} needs to be evaluated for different μ^{σ} and generally cannot be computed by first computing $u(s, \mu^{s'})$ for each pure strategy s' in the support of σ and then taking an expectation as in finite games. Moreover, since it is very unlikely to encounter a same distribution μ^{σ} across EGTA iterations, it is also not useful to store the corresponding utility information. Due to these characteristics, instead of storing utility information and re-using them, the iterative EGTA approaches up to this point (including our primary method and the version by Muller et al. [11]) need to compute or simulate $u(s, \mu^{\sigma})$ whenever μ^{σ} changes in the equilibrium computation (i.e., query-based implementation), which results in a low sample efficiency. To improve the sample efficiency, we introduce GML and regularization to our primary method. For discussion purposes in this section, we assume that MFGs are single-population (i.e., $N = 1$) and our approach can be readily extended to MFGs with multiple populations.

5.1 Game Model Learning

Our GML approach is a form of regression that learns the utility function based on utility information collected over previous EGTA iterations. With a game model (i.e., a learned utility function), sample efficiency can be improved by querying the game model rather than running simulations as long as the game model is able to provide high-fidelity predictions on these queries. In the following discussion, we first discuss where a game model fits in the EGTA framework and then elaborate the method for learning a game model given utility samples.

5.1.1 *Applying GML to EGTA.*

To implement GML in iterative EGTA, one key step is to periodically update the game model based on collected utilities and apply the game model to running a subroutine for equilibrium computation of the current empirical game. We select RD as a subroutine for illustration purposes.

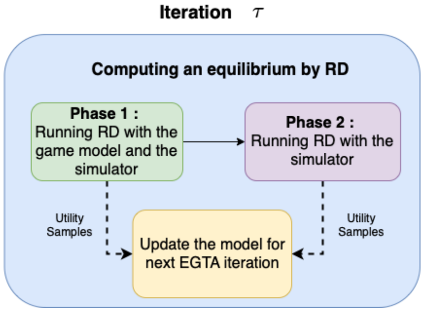

Iteration τ
Computing an equilibrium by RD
Phase 1 : Running RD with the game model and the simulator
Phase 2 : Running RD with the simulator
Utility Samples
Utility Samples
Update the model for next EGTA iteration

Figure 1: RD with a game model.

Denote the current EGTA iteration as iteration τ. Our object is to approximate the NE of the current empirical game with the strategy set Λ_{τ}, using RD and the game model $\hat{u}_{\tau-1}$ learned based on $\Lambda_{\tau-1}$ (i.e., the game model learned from previous iterations). Since the model $\hat{u}_{\tau-1}(s, \mu^{\sigma})$ only contains utility information of $s \in \Lambda_{\tau-1}$ and $\sigma \in \Delta\Lambda_{\tau-1}$ from previous iterations, while RD requires utilities $u(s, \mu^{\sigma}), \forall s \in \Lambda_{\tau}, \forall \sigma \in \Delta\Lambda_{\tau}$ for the current iteration, we cannot directly apply the model $\hat{u}_{\tau-1}$ due to the lack of information of s_{τ}. To handle this issue, we interleave utility approximation with simulations for different strategies.

In particular, for $s \in \Lambda_{\tau-1}$ and $\sigma \in \Delta\Lambda_{\tau-1}$, we directly apply the model to predict $u(s, \mu^{\sigma})$. For $s \in \Lambda_{\tau-1}$ and $\sigma \in \Delta\Lambda_{\tau}$, we first project σ onto $\Delta\Lambda_{\tau-1}$ by a projection operator $P_{\tau-1}(\sigma) = \text{argmin}_{\sigma' \in \Delta\Lambda_{\tau-1}} ||\sigma' - \sigma||_2$ and approximate $u(s, \mu^{\sigma})$ by $\hat{u}_{\tau-1}(s, \mu^{P(\sigma)})$. Our assumption here is that if $P(\sigma)$ is close to σ, then $u_{\tau-1}(s, \mu^{\sigma})$ is close to $u_{\tau-1}(s, \mu^{P(\sigma)})$ and the model is valid for estimating $u(s, \mu^{\sigma})$. The scenario in the assumption often holds at the start of running RD when RD is initialized with the equilibrium strategy from last iteration. Since the update of strategy in RD is controlled by a small step size, $P(\sigma)$ will be close to σ within first few RD iterations. For $s = s_{\tau}$, we query the simulator (e.g., a noiseless simulator) to obtain the exact utility $u(s, \mu^{\sigma})$ because the game model does not contain its utility information. We refer to this procedure as the first phase.

The number of RD iterations for the first phase is determined by the quality of utility estimations given by the projection. To measure this quality, we set a threshold for the L-2 distance between σ and its projection $P(\sigma)$. If the distance goes beyond the threshold, it means that the game model with projection becomes less accurate on the utility predictions. So we should stop using the model and switch to the simulator (i.e., the second phase).

In the second phase, we directly run RD with the simulator for all $s \in \Lambda_\tau$ and $\sigma \in \Delta\Lambda_\tau$ for a fixed number of iterations. At the end the second phase, utilities collected from the simulator at both phases are used to fine-tune the current game model. Note that the sampling of these utilities is guided by RD to avoid sampling the whole strategy space, where the latter will hurt our motivation of improving sample efficiency. The overall framework is depicted in Figure 1.

5.1.2 Learning Utility Functions. Given an empirical game at iteration τ, we aim at learning a game model (i.e., a utility function) $\hat{u}_\tau(\sigma, \mu^{\sigma'}), \forall \sigma, \sigma' \in \Delta\Lambda_\tau$ given the model from previous iterations $\hat{u}_{\tau-1}$ (which can be a random function for $\tau = 1$) and new utility data generated at the current iteration τ. To obtain the utility $\hat{u}(\sigma, \mu^{\sigma'})$ defined on mixed strategies $\sigma \in \Delta\Lambda_\tau$, it is sufficient to learn a game model $\hat{u}(s, \mu^{\sigma'})$ on pure strategies $s \in \Lambda_\tau$ and take an expectation. We focus on a general setup of MFGs where a strategy s and distributions μ are generally both time-dependent (i.e., $s = (s_t)_{t\in[0,T-1]}$ and $\mu = (\mu_t)_{t\in[0,T]}$). Explicitly encoding them as inputs to a learner (either a neural network or other regression models) will result in a high-dimensional input vector and entail an impractically complex regression setup.

To handle this issue, we apply a *coarse coding* scheme from a separate work. Specifically, let $I : \Lambda_\tau \rightarrow \mathbb{Z}_+$ be an indexing function that assigns a unique index to each strategy $s \in \Lambda$. Let $\sigma \in \Delta\Lambda_\tau$ be the mixed strategy that induces the distribution μ^σ. Since the distribution induction function (Eq. 1) is deterministic, the distributions μ^σ are uniquely determined by a fixed initial distribution $\mu_0 \in \Delta(X)$ and a strategy σ.[2] Rather than learning $\hat{u}(s, \mu^\sigma)$ with time-dependent inputs, we learn a game model $\hat{u} : I(\Lambda_\tau) \times \Delta(\Lambda_\tau) \rightarrow \mathbb{R}$ with equivalent representations $I(s)$ and σ of s and μ^σ.

Based on coarse coding, a utility data point is constructed to include an index of a pure strategy $I(s)$, a mixed strategy σ, and a utility target $u(s, \mu^\sigma)$. Since the object is to predict the true utility $u(s, \mu^\sigma)$ by $\hat{u}(I(s), \sigma)$, we fine-tune $\hat{u}_{\tau-1}$ by minimizing the mean square error $E[u(s, \mu^\sigma) - \hat{u}(I(s), \sigma))^2]$. Our regression is based on neural networks and the structure of the neural networks is shown in Figure 2.

5.2 Regularization

Our regularization method improves sample efficiency through reducing the number of iterations of RD in each EGTA iteration. In current iterative EGTA approaches, the number of iterations for the subroutines (e.g., RD and FP) is set to be large enough so that they can approximately converge to the current equilibrium of the empirical game. This mimics the classic double oracle method, in which a best response to the current equilibrium is computed at each iteration.

In a separate work [23], we have shown that properly regularizing the best response target (i.e., not best-responding to an exact equilibrium) will lead to improved learning performance for EGTA. Regularization can be achieved by early stopping RD when the regret of the current profile with respect to the empirical game exceeds a pre-defined regret threshold. For our purposes, early stopping of RD means less utility queries from the simulator and improved sample efficiency if the overall learning performance would not decline. We apply this approach to our EGTA framework by replacing the regret threshold with a fixed number of RD iterations since the prediction error given by the game model could affect the regret estimation for reaching a threshold.

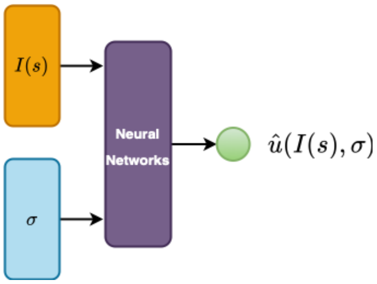


Figure 2: Neural network structure for coarse coding.

5.3 Algorithms

We show the complete EGTA framework with GML and regularization for single-population MFGs in Algorithm 4 and Algorithm 5. Compared to our primary method (Alg.1), the main differences are the introduction of a game model $\hat{u}$ in Alg. 4 and how the model $\hat{u}$ is updated and applied to RD in Alg. 5 (discussed in Section 5.1). Note that, in Alg. 5, RD is initialized with the equilibrium σ^e from last EGTA iteration, perturbed by a function δ to guarantee a full support (make sure every strategy can be played with non-zero probability and hence can be updated by RD). Regularization is achieved by controlling the maximal number of RD iteration J.

Algorithm 4 Iterative EGTA with GML and regularization

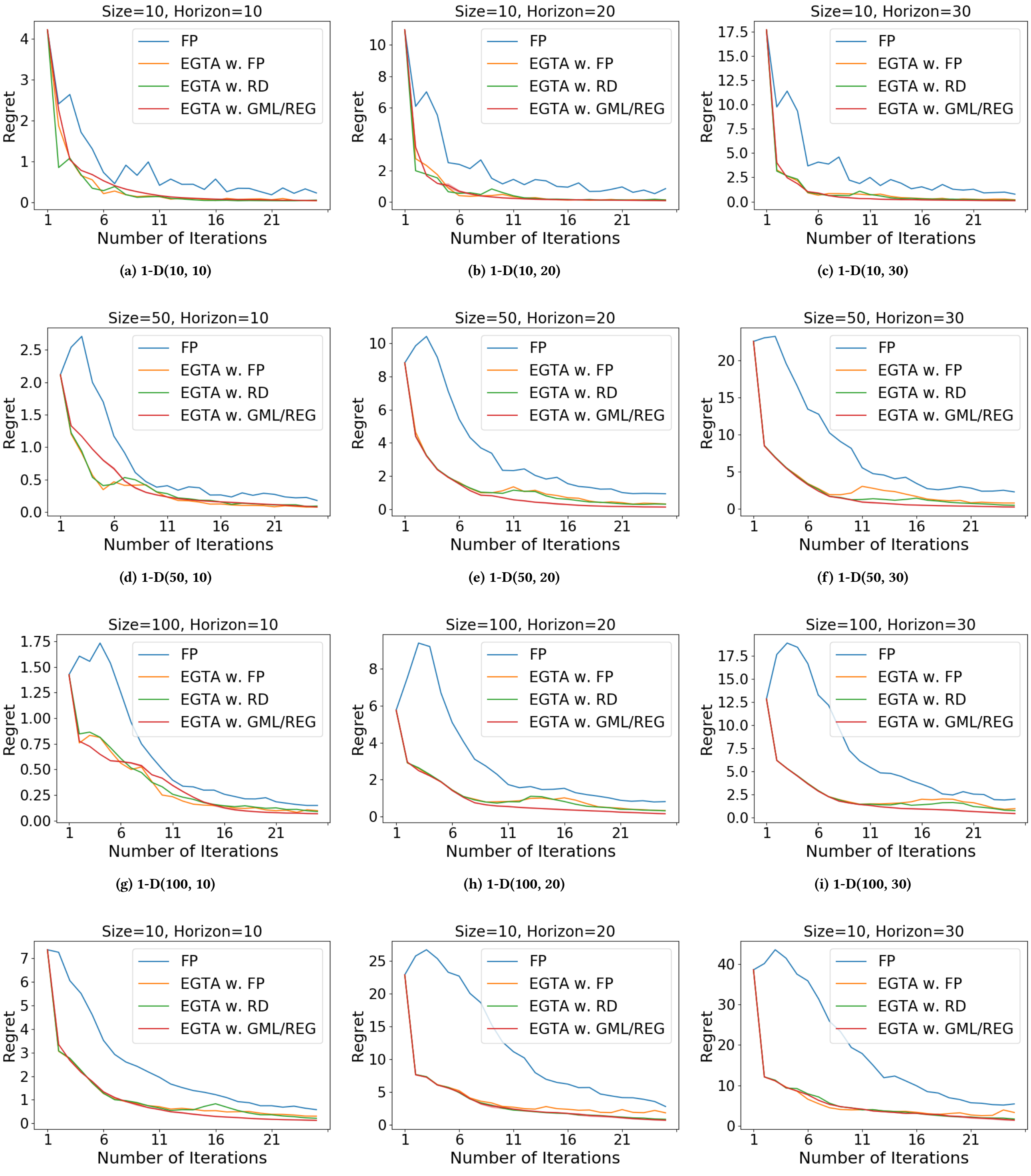
```
Input: An initial strategy $\Lambda_0 = \{s_0\}$ and an initial distribution $\mu_0$.
    A neural network $\hat{u}$.
$\sigma^e \leftarrow s_0$
Initialize $\mu^e$ by Eq. 1 using $s_0$
for EGTA iteration $\tau \in \{1, \ldots, \mathcal{T}\}$ do
    Compute a best response strategy $s_\tau$ to the empirical equilibrium distribution $\mu^e$
    Add $s_\tau$ to the strategy set of population $i$: $\Lambda_\tau \leftarrow \Lambda_{\tau-1} \bigcup s_\tau$
    Compute $\sigma^e, \mu^e, \hat{u} \leftarrow$ a subroutine $\Psi(\mathcal{G}_{S\downarrow\Lambda_\tau}, \hat{u}, \sigma^e, \mu^e)$
end for
Return $(\sigma^e, \mu^e)$
```

6 EXPERIMENTAL RESULTS

6.1 The 1-D Beach Bar

We consider a simplified version of Santa Fe bar problem [1, 5] and adopt the model by Perrin et al. [15]. Specifically, a beach bar problem for a single-population MFG is a Markov Decision Process with

[2] We assume the initial distributions μ_0 are fixed, so it is sufficient for σ to determine μ^σ.



(a) 1-D(10, 10)

(b) 1-D(10, 20)

(c) 1-D(10, 30)

(d) 1-D(50, 10)

(e) 1-D(50, 20)

(f) 1-D(50, 30)

(g) 1-D(100, 10)

(h) 1-D(100, 20)

(i) 1-D(100, 30)

(j) 2-D(100, 10)

(k) 2-D(100, 20)

(l) 2-D(100, 30)

Figure 3: Experimental results of 1-D and 2-D beach bar problems.

Algorithm 5 Regularized Replicator Dynamics as a Subroutine Ψ

```
Input: An empirical game $\mathcal{G}_{S\downarrow\Lambda_\tau}$. A learned utility simulator $\hat{u}$.
    Equilibrium strategy $\sigma^e$, and distribution $\mu^e$.
    Parameters: A distance threshold $\gamma$. A maximal number of
    iterations for applying the model $M$. A learning rate $dt$.
Initialize a strategy $\bar{\sigma} \leftarrow \delta(\sigma^e)$
for RD iteration $j \in \{1, \dots, J\}$ do
    $\bar{\sigma}^p \leftarrow P_{\tau-1}(\bar{\sigma})$
    if $||\bar{\sigma}, \bar{\sigma}^p||_2 < \gamma$ and $j < M$ then
        Approximate $u(s, \mu^{\bar{\sigma}})$ by $\hat{u}(I(s), \bar{\sigma}^p), \forall s \in \Lambda_{\tau-1}$
        Simulate $u(s_\tau, \mu^{\bar{\sigma}})$
    else
        Simulate $u(s, \mu^{\bar{\sigma}}), \forall s \in \Lambda_\tau$
    end if
    Save new data points $I(s)$, $\bar{\sigma}$, and $u(s, \mu^{\bar{\sigma}})$
    Compute fitness $F = \sum_{s\in\Lambda} \bar{\sigma}(s) u(s, \mu^{\bar{\sigma}})$
    for $s \in \Lambda_\tau$ do
        $\bar{\sigma}(s) \leftarrow \bar{\sigma}(s) + dt * \bar{\sigma}(s)[u(s, \mu^{\bar{\sigma}}) - F]$
    end for
end for
Fine-tune $\hat{u}$ with all new data points
Compute the induced distributions $\mu^{\bar{\sigma}}$ by Eq. 1 using $\bar{\sigma}$
Return $\bar{\sigma}$, $\mu^{\bar{\sigma}}$, and $\hat{u}$
```

$|X|$ states on a one-dimensional torus ($X = 0, \dots, |X| - 1$). Without loss of generality, we designate a bar to the state 0. Positions of players are initialized according to a uniform distribution. Players can keep still ($a_t = 0$) or move left ($a_t = -1$) or right ($a_t = 1$) at time step t on the torus to get as close as possible to the bar, while avoiding the crowded areas. The transition function is given by:

$$x_{t+1} = x_t + a_t + \epsilon_t$$

where a_t is an action of the representative player at time t and ϵ_t represents the randomness of the environment. The immediate reward function is given by:

$$r(x_t, a_t, \mu_t) = \tilde{r}(x_t) - \frac{|a_t|}{|X|} - \log(\mu_t(x_t))$$

where $\tilde{r}(x_t)$ measures the closeness to the bar from state x_t, $-\frac{|a_t|}{|X|}$ is the running cost and $-\log(\mu_t(x_t))$ represents the aversion of players to the crowded areas.

We test the performance of iterative EGTA in the 1-D beach bar problem with various configurations, which are determined by the Cartesian product of $|X| \in \{10, 50, 100\}$ and $T \in \{10, 20, 30\}$, denoted by 1-D($|X|$, T). In Figure 3a-3i, we plot the regret curves of iterative EGTA (with FP and RD as subroutines respectively) against directly applying FP to MFGs [15], where x-axis being the EGTA iterations. Since the game size supports exact best response calculation and exact strategy evaluation given a fixed initialization and parameters of the randomness of the environment (i.e., the simulator is noiseless), the regret curves can be exactly computed and hence no error bar is reported in the plots.

From Figure 3a-3i, we observe that the performance of our primary EGTA method (orange and green curves) dominates FP in all instances. Moreover, as the instance becomes complex (i.e., with more states and longer horizon), the performance gap between our primary method and FP becomes more apparent. For the subroutine selection, we observe that RD in some cases exhibits a more stable learning manner (e.g., Fig 3f) than FP while in most cases their performances are almost indistinguishable. For EGTA with GML and regularization (abbr. REG in the plots) (red curve), we observe that its performance also almost coincides with our primary method, but becomes even most stable as time horizon increases due to regularization. Thanks to GML and regularization, we obtain this performance with only 1/6 of the total utility queries compared to our primary method, which demonstrates a significant improvement on the sample efficiency. We show the number of simulations needed across EGTA iterations in Figure 4.

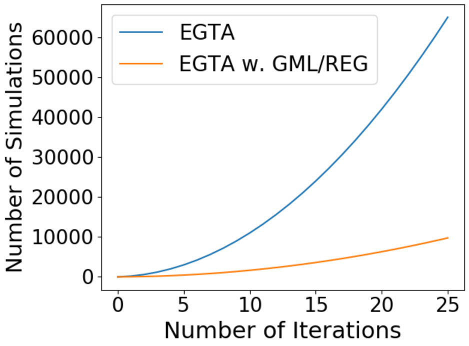


Figure 4: The number of utility samples across EGTA iterations.

6.2 The 2-D Beach Bar

We test the performance of our primary EGTA method in the 2-D beach bar problem with fixed $|X| = 100$ and different time horizon $T \in \{10, 20, 30\}$, denoted by 2-D($|X|$, T). Compared to the 1-D beach bar problem, the 2-D beach bar problem includes extra actions (i.e., up, down, left, right, stay) yielding a more complex environment. From Figure 3j-3l, we observe the same phenomenon as in the 1-D problem, that is, iterative EGTA dominates FP and as the game instance becomes larger, the advantage of iterative EGTA becomes apparent. EGTA with GML and regularization again exhibits similar performance compared to the primary method but only consumes 1/6 of the utility queries.

6.3 Multi-population Chasing

For MP-MFGs, we test the performance of iterative EGTA in a 3-population chasing problem [12] and observe that iterative EGTA also exhibits faster convergence than FP in the 3-population MFG. We report the environment definition and our experimental results in the Appendix.

7 CONCLUSION

We propose an iterative EGTA framework for computing NE in MFGs and a sample-efficient version by combining GML and regularization. We demonstrate the efficacy of our approaches in various MFGs. Theoretically, we prove the existence of NE in empirical MFGs and the convergence of the iterative EGTA framework.

ACKNOWLEDGMENTS

This work was supported in part by funding from the US Army Research Office (MURI grant W911NF-18-1-0208), and a grant from the Effective Altruism Foundation.

REFERENCES

[1] W. Brian Arthur. 1994. Inductive reasoning and bounded rationality. *American Economic Review* 84, 2 (1994), 406–411.

[2] George W Brown. 1951. Iterative solution of games by fictitious play. *Activity Analysis of Production and Allocation* 13, 1 (1951), 374–376.

[3] Wojciech Marian Czarnecki, Gauthier Gidel, Brendan Tracey, Karl Tuyls, Shayegan Omidshafiei, David Balduzzi, and Max Jaderberg. 2020. Real World Games Look Like Spinning Tops. In *34th Annual Conference on Neural Information Processing Systems*.

[4] Romuald Elie, Julien Perolat, Mathieu Laurière, Matthieu Geist, and Olivier Pietquin. 2020. On the convergence of model free learning in mean field games. In *34th AAAI Conference on Artificial Intelligence*. 7143–7150.

[5] Julie Farago, Amy Greenwald, and Keith Hall. 2002. Fair and efficient solutions to the Santa Fe bar problem. In *Grace Hopper Celebration of Women in Computing*. Citeseer.

[6] Sevan G Ficici, David C Parkes, and Avi Pfeffer. 2008. Learning and solving many-player games through a cluster-based representation. In *In Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*. 187–195.

[7] Minyi Huang, Roland P Malhamé, Peter E Caines, et al. 2006. Large population stochastic dynamic games: Closed-loop McKean-Vlasov systems and the Nash certainty equivalence principle. *Communications in Information & Systems* 6, 3 (2006), 221–252.

[8] Marc Lanctot, Vinicius Zambaldi, Audrūnas Gruslys, Angeliki Lazaridou, Karl Tuyls, Julien Pérolat, David Silver, and Thore Graepel. 2017. A unified game-theoretic approach to multiagent reinforcement learning. In *31st Annual Conference on Neural Information Processing Systems* (Long Beach, CA). 4190–4203.

[9] Jean-Michel Lasry and Pierre-Louis Lions. 2007. Mean field games. *Japanese Journal of Mathematics* 2, 1 (2007), 229–260.

[10] H. Brendan McMahan, Geoffrey J. Gordon, and Avrim Blum. 2003. Planning in the presence of cost functions controlled by an adversary. In *20th International Conference on Machine Learning*. 536–543.

[11] Paul Muller, Mark Rowland, Romuald Elie, Georgios Piliouras, Julien Perolat, Mathieu Lauriere, Raphael Marinier, Olivier Pietquin, and Karl Tuyls. 2021. Learning Equilibria in Mean-Field Games: Introducing Mean-Field PSRO. In *21th International Conference on Autonomous Agents and Multi-Agent Systems*.

[12] Julien Perolat, Sarah Perrin, Romuald Elie, Mathieu Laurière, Georgios Piliouras, Matthieu Geist, Karl Tuyls, and Olivier Pietquin. 2022. Scaling up Mean Field Games with Online Mirror Descent. In *21th International Conference on Autonomous Agents and Multi-Agent Systems*.

[13] Sarah Perrin, Mathieu Laurière, Julien Pérolat, Romuald Élie, Matthieu Geist, and Olivier Pietquin. 2022. Generalization in Mean Field Games by Learning Master Policies. In *36th AAAI Conference on Artificial Intelligence*.

[14] Sarah Perrin, Mathieu Laurière, Julien Pérolat, Matthieu Geist, Romuald Élie, and Olivier Pietquin. 2021. Mean Field Games Flock! The Reinforcement Learning Way. In *30th International Joint Conference on Artificial Intelligence*.

[15] Sarah Perrin, Julien Pérolat, Mathieu Laurière, Matthieu Geist, Romuald Elie, and Olivier Pietquin. 2020. Fictitious play for mean field games: Continuous time analysis and applications. In *34th Annual Conference on Neural Information Processing Systems*.

[16] Julia Robinson. 1951. An iterative method of solving a game. *Annals of Mathematics* (1951), 296–301.

[17] J Maynard Smith and George R Price. 1973. The logic of animal conflict. *Nature* 246, 5427 (1973), 15–18.

[18] Samuel Sokota, Caleb Ho, and Bryce Wiedenbeck. 2019. Learning deviation payoffs in simulation-based games. In *33rd AAAI Conference on Artificial Intelligence*, Vol. 33. 2173–2180.

[19] Peter D Taylor and Leo B Jonker. 1978. Evolutionary stable strategies and game dynamics. *Mathematical biosciences* 40, 1-2 (1978), 145–156.

[20] Karl Tuyls, Julien Pérolat, Marc Lanctot, Edward Hughes, Richard Everett, Joel Z. Leibo, Csaba Szepesvári, and Thore Graepel. 2020. Bounds and dynamics for empirical game theoretic analysis. *Autonomous Agents and Multi-Agent Systems* 34 (2020), 7.

[21] Yevgeniy Vorobeychik, Michael P. Wellman, and Satinder Singh. 2007. Learning payoff functions in infinite games. *Machine Learning* 67 (2007), 145–168.

[22] Yufei Wang, Zheyuan Ryan Shi, Lantao Yu, Yi Wu, Rohit Singh, Lucas Joppa, and Fei Fang. 2019. Deep reinforcement learning for green security games with real-time information. In *33rd AAAI Conference on Artificial Intelligence*. 1401–1408.

[23] Yongzhao Wang and Michael P. Wellman. 2023. Regularization for Strategy Exploration in Empirical Game-Theoretic Analysis. arXiv:2302.04928 [cs.GT]

[24] Michael P. Wellman. 2006. Methods for Empirical Game-Theoretic Analysis (Extended Abstract). In *Twenty-First National Conference on Artificial Intelligence*. Boston, 1552–1555.

[25] Bryce Wiedenbeck, Fengjun Yang, and Michael P. Wellman. 2018. A regression approach for modeling games with many symmetric players. In *Thirty-Second AAAI Conference on Artificial Intelligence* (New Orleans). 1266–1273.

[26] Mason Wright, Yongzhao Wang, and Michael P. Wellman. 2019. Iterated Deep Reinforcement Learning in Games: History-Aware Training for Improved Stability. In *EC* (Phoenix). 617–636.