Model Free Reinforcement Learning Algorithm for Stationary Mean field Equilibrium for Multiple Types of Agents

Arnob Ghosh  
arnob.ghosh@imperial.ac.uk  
Imperial College of London  
London, UK

Vaneet Aggarwal  
vaneet@purdue.edu  
Purdue University  
West Lafayette, IN, USA

ABSTRACT

We consider a multi-agent Markov strategic interaction over an infinite horizon where agents can be of multiple types. We model the strategic interaction as a mean-field game in the asymptotic limit when the number of agents of each type becomes infinite. Each agent has a private state; the state evolves depending on the distribution of the state of the agents of different types and the action of the agent. Each agent wants to maximize the discounted sum of rewards over the infinite horizon which depends on the state of the agent and the distribution of the state of the leaders and followers. We seek to characterize and compute a stationary multi-type Mean field equilibrium (MMFE) in the above game. We characterize the conditions under which a stationary MMFE exists. Finally, we propose a Reinforcement learning (RL) based algorithm using policy gradient approach to find the stationary MMFE when the agents are unaware of the dynamics. We, numerically, evaluate how such kind of interaction can model the cyber attacks among defenders and adversaries, and show how RL based algorithm can converge to an equilibrium.

1 INTRODUCTION

In many real world applications, such as in cyber-physical system, intelligent transportation, cyber-security, smart grid, and Internet of Things (IoT), strategic agents (e.g., smart devices, robots, rational humans) interact with each other. In most of these applications, the agents interact repeatedly and the ‘reward’ or the utility of an agent not only depends on her own action but also on the actions of other agents.

Finding the optimal strategy of the agents becomes more challenging when the agents are unaware of the exact environment and reward. Multi-agent Reinforcement learning (MARL) has been developed to model the uncertainty [10]. In MARL, similar to a Markov stochastic game, the environment of an agent is embedded as a state. The state and reward obtained by an agent depends on the current state or action of each individual agent. However, due to the curse of dimensionality, analyzing MARL with a large number of players is challenging when the agents interact for a longer time.

Mean-field game (MFG) has recently been popular to study interaction among a large number of statistically similar agents. MFG studies the game in the asymptotic limit where the number of agents is considered to be infinite. When the number of agents becomes large, only the distribution of the states (or, actions) of all the agents rather than the individual state (or, action) along with her own state and action only impacts the state and reward of an individual agent in a MFG. Thus, in a MFG, each agent only needs to consider the distribution of the agents’ states (or, actions) rather than individual state or action of the other agents. MFG along with the Reinforcement learning setting has been of recent interest to analyze strategic interactions among a large number of agents in an unknown environment.

In the traditional MFG, all the players are statistically similar. However, in a lot of scenario players can be of different types. In cyber-physical system, agents are often of different types. Further, leader-follower type of interaction where a set of agents (leaders) first take their actions and the other set of agents (followers) follow in a MFG setting is ubiquitous. For example, in cyber-security, cyber-physical system, agents are often of different types. Further, the agents are not statistically similar any more since there are now two different classes of agents: leaders and followers. In a MFG, mean-field equilibrium (MFE) policy is defined as an equilibrium policy where no agent has any incentive to deviate from its own strategy. However, when agents are of different types the equilibrium needs to be optimal in a leader-follower game with large number of agents, we need to compute a MFE for followers for any policy of a leader, then we need to determine the MFE policy for the leaders which must also consider the MFE policy for the followers. The analysis becomes more challenging when the agents are unaware of the reward functions and the transition probability kernels.

1.1 Contribution

We first, develop theoretical tools for modeling and analyzing a simultaneous Markov MFG (Section 2.2) where agents of different types interact simultaneously over a long period of time. We propose a Multi-type Mean-field-equilibrium (MMFE) concept in the MFG setting (Section 2.3). In a MMFE, there are two policies—one for each type of agents. In a MMFE, the policy at a given state of an agent must be optimal given the actions of the agents of the same type as well as the actions of the agents of different types. Further, the evolution of the distribution of the states of both the types of agents must be consistent with the policies.

We consider a stationary MMFE where the population distribution of the agents, and the policies of the agents do not change over time (Section 2.4). Thus, in the stationary MMFE, the population distribution at the next time must remain the same even when the agents of each type take their optimal actions based on individual
We characterize the conditions on the policy parameters such that we propose a novel Random-Horizon policy gradient mechanism which as stationary MMFE exists which are easy to verify. Further, dynamics.

At time step $t$, the agent $i$ of type $j$ gets a reward based on her own state and action, the other agents’ states of type $j$, and the states of agents of type $k \neq j$. We denote the reward of any agent $i$ of type $j$ as $r^i(x_{1,t}, x_{2,t}, a^i_{j,t})$. Here, $x_{i,j}$ is the state vector of all the agents of type $j$. All the agents take actions simultaneously.¹

The state of agent $i$ of type $j$ at time $t+1$ evolves depending on the transition probability kernel $r^i(x_{1,t}, x_{2,t}, a^i_{j,t})$. The transition probability of agent $i$ of the type $j$ depends on the state vector of all the agents of both the types and action of the agent $i, a^i_{j,t}$. Note that inherently, the transition probability depends on the state of the agent $i$ which is included in the joint state space of the agents.

Remark 1. The reward and the transition probabilities may also depend on the joint actions of agents of different types. We can extend our analysis to the above scenario and we will discuss it in Section 6.

A Markovian game restricts the admissible policies for a player to be of markovian in nature. The policy for an agent $i$ of type $j$ is given by $\pi_t^{i,j} : X_j^N \times X_j^N \rightarrow \mathcal{P}(\mathcal{A}_j)$, where $\mathcal{P}(\mathcal{A}_j)$ is the probability space of the action space of agents of type $j$. Thus, the policy of an agent depicts the action which the agents takes given the state vector of all the agents. Note that the policy can be randomized where an agent can randomize over multiple actions. A policy can also be deterministic where the agent takes a certain action for a specific joint state distribution of leaders and followers.

The accumulated reward for an agent $i$ (a.k.a. value function) of type $j$, starting from an initial state vector $x_{0,1}, x_{0,2}$ and the policy sequence $(\pi^1, \pi^2) = (\pi^1, \pi^2)_{t=0}^\infty$ with $\pi^1_t = (\pi^1_{1,t}, \ldots, \pi^1_{N_j,t})$ and $\pi^2_t = (\pi^2_{1,t}, \ldots, \pi^2_{N_j,t})$ is given by

$$V^i_t(x_1, x_2, \pi^1, \pi^2) = \mathbb{E}[\sum_{t=0}^\infty \gamma^t r^i(x_{1,t}, x_{2,t}, a^i_{j,t}) | x_{0,1} = x, y_{0,1} = y]$$

where $0 < \gamma < 1$ is a discount factor, $a^i_{j,t} \sim \pi^i_{j,t}(x_{1,t}, x_{2,t})$ and $x_{i,j,t+1} \sim \pi^i_{j,t}(x_{1,t}, x_{2,t}, a^i_{j,t})$. Each agent $i$ of type $j$ would want to maximize her own value function over the policies $\pi^i_{j,t}$. The expectation here is both on the action profile of all the agents and the transition probabilities based on the action profiles of agents.

1 It is equivalent to the setting where the agents do not observe the actions of the other agents.
2.2 Mean Field Game for different types of agents

In general, Markovian game where the agents are of single type is difficult to analyze for finite number of players. Mean-field-game (MFG), pioneered by [12] and [14] in the continuous setting and latter developed for the discrete setting [2, 16, 22], provides a tractable approach for analyzing the finite player Markovian game. The idea of MFG is simple: Assume that all the players are identical, interchangeable, and indistinguishable from one another, when the number of player $N \to \infty$, we can view the limit of player states as a population state distribution.

In the MFG of multiple types of agents, similar to the MFG where agents are of single type, we consider that both $N_1 \to \infty$ and $N_2 \to \infty$. The agents of type $j$ are identical and indistinguishable from each other. However, the agents of types $1$ and $2$ are different. For each agent’s perspective, the distribution of the states of all the other agents of its own type, states of the other type, and her own state and action would impact the states and rewards.

We define the limit of the state distribution of the agents of type $j$ as $z_{j,t}$

$$z_{j,t}(s) = \lim_{N_j \to \infty} \frac{\sum_{i=1}^{N_j} \mathbb{1}_{x_{j,t,s}}}{N_j}$$

where $\mathbb{1}_{x_{j,t,s}}$ indicates that the state of agents $i$ of type $j$ at time $t$ is $s$ (the value is 1 if the state of the agent $i$ of type $j$ at time $t$ is $s$ and is zero otherwise). This is also known as type $j$’s population distribution.

The transition probability kernel of an agent $i$ of type $j$ when she is in state $x_{j,t}$ and takes an action $a_{j,t}$ is now represented as $r^j(\cdot|x_{j,t},z_{1,t},z_{2,t},a_{j,t})$. The reward function of agent $i$ of type $j$ when she is in state $x_{j,t}$ and takes action $a_{j,t}$, is represented by $r^j(x_{j,t},a_{j,t},z_{1,t},z_{2,t})$. Hence, the transition probability and the reward both depend on the current state, action, and the population distribution of the agents of both the types. Note that the reward and the transition probabilities are identical for each agent of type $j$. However, the reward and the transition probabilities are different for agents of different types. Note that the agents of similar types may have greater influence on the agent compared to the agents of different type. Hence, the transition probability and rewards of agents of type $j$ may be influenced by $z_{1,t}$ compared to $z_{2,t}$ for $k \neq j$.

The policy of an agent of type $j$ is now a function from state $x_{j,t}$, the population distribution of type $j$’s agents $z_{j,t} \in \mathbb{Z}_j$, the population distribution of the types $k \neq j$ agents $z_{k,t} \in \mathbb{Z}_k k \neq j$ to the probability space over the actions. It is a mapping from $\pi^j_{t} : \mathbb{Z}_j \times \mathbb{Z}_2 \times X_j \to P(A_j)$

**Remark 2.** We can represent $z_t = \{z_{1,t}, z_{2,t}\}$ as the population state distribution of both types of agents.

Due to the homogeneity among the agents of particular type, we consider a representative agent who wants to maximize the value function starting from an initial state $x$:

$$V^j(x, \pi^1, \pi^2, z_{1}, z_{2}) = E[\sum_{t=0}^{\infty} r^j(x_{j,t},z_{1,t},z_{2,t},a_{j,t})|x_{j,0} = x],$$

where $z_{j}$ is the population distribution sequences of type $j$, i.e., $z_{j} = \{z_{j,t}\}_{t=0}^{\infty}$. Here the expectation is over both the transition kernel, and the policies $\pi^1$ and $\pi^2$.

2.3 Multi-agent Mean Field Equilibrium

Now, we define the multi-agent mean field equilibrium (MMFE).

**Definition 1.** A player-population profile $(\pi^1, \pi^2, z_{1}, z_{2})$ is called a MMFE if the following hold:

- **MFE** for each agent of type $j$, $j = 1, 2$: The policy must be MFE for the representative agent of type $j$, i.e., it must be optimal for a given reaction policy of agents of other type $k \neq j$, $\pi^k$, i.e., for all state $x$ and $\forall \pi^{j'}$,

$$V^j(x, \pi^j, \pi^k, z_{1}, z_{2}) \geq V^j(x, \pi^{j'}, \pi^k, z_{1}, z_{2}) \quad (3)$$

- **Population Consistency:** We must have for all $t \geq 0$ and for $j = 1, 2$

$$z_{j,t+1}(x) = \sum_{x_j,a_j} \tau^j(\tilde{x}|x_j,a_j,z_{1,t},z_{2,t})\pi^j_t(a_j|x_j,z_{1,t},z_{2,t})z_{j,t}(x) \quad (4)$$

where $z_{1,0}$ and $z_{2,0}$ are initial population distributions of type $1$ and $2$ respectively.

The first part of the definition specifies that $\pi^j_t$ must be optimal for agents of type $j$ for any given state given the policy of other agents of other type $k \neq j$ for each type $j$. The population consistency guarantees that the state distribution of the type $j$ $j = 1, 2$ is consistent with the state distribution evolution of the agents of type $j$ given the state distributions and policies of the agents for other type $k \neq j$. This part is unique for a MFG setting.

We represent (4) as

$$z_{j,t+1} = \phi^j(\pi^j_t, z_{1,t}, z_{2,t})$$

where $\phi^j$ depends on the transition probability $\tau^j$, the policy $\pi^j$ and $z_{1,t}$ as evident from (4).

**Remark 3.** If we consider that the reward and transition probability also depend on the action distributions of the agents, the action distribution also needs to be consistent with the MFE policy. Hence, we would need additional conditions, given as follows for $j = 1, 2$:

$$eq: action_{z_{j,t+1}}(a_j) = \sum_{x_j} z_{j,t+1}(x_j)\pi^{j}_{t+1}(a_j|x_j,z_{1,t+1},z_{2,t+1}) \quad (6)$$

Note that here the action distribution implicitly depends on the actions of the agents of type $k \neq j$, since the transition probability and reward both becomes functions of the action distributions of the agents of type $k$ as well.

2.4 Stationary MMFE

In a stationary MMFE, $z_{j,t} = z_{j}$, and $\pi^j_t = \pi^j \forall t$, for $j = 1, 2$. Thus, for a stationary MMFE, we have,

**Definition 2.** A player-population profile $(\pi^1, \pi^2, z_{1}, z_{2})$ is called a stationary MMFE if the following hold:

- **MFE** for agents of type $j$: The stationary policy must be MFE for the representative agent of type $j$, i.e., for all state $x$ and $\forall \pi^{j'}$, for $j = 1, 2$, and $k \in \{1, 2\}$, $k \neq j$,

$$V^j(x_j, \pi^j, \pi^k, z_{1}, z_{2}) \geq V^j(x_j, \pi^{j'}, \pi^k, z_{1}, z_{2}) \quad (7)$$
2.5 Bellman Equation for a stationary MMFE

In this section, we represent the Bellman equation corresponding to the value function which we use in our proposed Algorithm and the corresponding analysis in Section 4.

Let us assume the following Assumption 1. There exist a constant $d_1$ such that for any $z'_1, z'_2$

\[
\begin{align*}
D(\Gamma^1(z_1, z_2), \Gamma^1(z'_1, z'_2)) &\leq d_1 W_1((z_1, z_2), (z'_1, z'_2)) \\
D(\Gamma^2(z_1, z_2), \Gamma^2(z'_1, z'_2)) &\leq d_1 W_1((z_1, z_2), (z'_1, z'_2))
\end{align*}
\]

where

\[
D(\pi, \pi') = \sup_y W_1(\pi(y), \pi'(y))
\]

and $W_1$ is the $l_1$-Wasserstein distance between two probability measures.

Step 2. Based on the analysis on Step 1, update the sequence $z_j$ to $z'_j$ for $j = 1, 2$ according to population dynamics in (5).

Accordingly, for any admissible policy of an agent of type $j$ $\pi^j$ and the joint population distribution $(z_1, z_2)$, define the mapping $\Gamma : \Pi^1 \times \Pi^2 \times P(X_1 \times X_2) \rightarrow P(X_1 \times X_2)$, $\Gamma$ operates on the optimal policy of the agents of both the types for a given population distribution and outputs a population distribution according to (5) based on the policies of the agents of both the types and the current population distribution.

We assume the following Assumption 2. There exist constants $d_2, d_3 \geq 0$, such that for any admissible policies $\pi^1, \pi^1'$ and population distributions $z_1, z'_1, z_2, z'_2$.

\[
\begin{align*}
W_1(\Gamma(\pi^1, \pi^2, (z_1, z_2)), \Gamma(\pi^1', \pi^2', (z_1, z_2))) &\leq d_2 max(D(\pi^1, \pi^1'), D(\pi^2, \pi^2')) \\
W_1(\Gamma(\pi^1, \pi^2, (z_1, z_2)), \Gamma(\pi^1', \pi^2', (z'_1, z'_2))) &\leq d_3 W_1((z_1, z_2), (z'_1, z'_2))
\end{align*}
\]

Step 3. Repeat Steps 1 and 2 until both $z_1 = z'_1$ and $z_2 = z'_2$. The following theorem shows that the prescribed methodology indeed converges to stationary MMFE.

\[
\text{Theorem 1. Given Assumptions 1 and 2, and assume } d_1 d_2 + d_3 < 1, \text{ then there exists a stationary MMFE.}
\]

Proof. By the definition of stationary MMFE, $(\pi^1, \pi^2, z_1, z_2)$ is a stationary MMFE, if and only if $z_1 = \Gamma(z_1, z_2)$ = $\Gamma(\Gamma^1(z_1, z_2), \Gamma^2(z_1, z_2), (z_1, z_2))$ and $\pi^1 = \Gamma^1(z_1, z_2)$ where $\Gamma(z_1, z_2) = \Gamma^1(z_1, z_2), \Gamma^2(z_1, z_2))$. This indicates for any two population distributions $z'_1, z_1, z'_2, z_2$

\[
\begin{align*}
W_1(\Gamma(z_1, z_2), \Gamma(z'_1, z'_2)) &\leq W_1(\Gamma(z_1, z_2), \Gamma^1(z'_1, z'_2), \Gamma^2(z'_1, z'_2), (z'_1, z'_2)) \\
&\leq W_1(\Gamma(z_1, z_2), \Gamma^1(z'_1, z'_2), \Gamma^2(z'_1, z'_2), (z'_1, z'_2)) + W_1(\Gamma(z_1, z_2), \Gamma^1(z'_1, z'_2), \Gamma^2(z'_1, z'_2), (z'_1, z'_2))
\end{align*}
\]

\[
\leq d_2 d_1 W_1((z_1, z_2), (z'_1, z'_2)) + d_3 W_1((z_1, z_2), (z'_1, z'_2))
\]

We must have for all $t$ and $j = 1, 2$

\[
z_j = \phi^j(\pi^j, z_1, z_2)
\]

In general, computing a non-stationary MMFE is challenging since one needs to find a fixed point at every time step by considering the sequences of population distributions. Thus, in non-stationary MMFE, a finite horizon is generally used. Stationary MMFE is computationally simple. Thus, in the following we focus on the conditions under which a MMFE exists and how to compute a stationary MMFE.
We propose a policy gradient based algorithm to find a MMFE. The state and action space is convex, similar to [1] which considers only one type of agents. When the action space is finite Assumption 2 can written in the following lemma.

**Lemma 1.** Suppose that max \( j (x^{j'}|x_j,a_j,z_1,z_2) \leq c_1 \) for all \( z_1,z_2,x_j,a_j \), and \( \tau^j, j = 1,2 \) are\( c_2 \)-Lipschitz in \( W_1 \), i.e., \( \forall x_j,x_j',a_j \)
\[
|\tau^j(x_j'|x_j,a_j,z_1,z_2)| \leq c_2 W_1((z_1,z_2),(z_1',z_2'))
\]
then in Assumption 2, \( d_2 \) and \( d_3 \) are
\[
d_2 = \max_j \text{diam}(X_j) \max_j |X_j| c_1, \quad d_3 = \frac{\max_j \text{diam}(X_j) c_2}{2}
\]
where \( d_{min}(\mathcal{A}_j) = \min_{a_j,a_j'} ||a_j-a_j'||_2 \), \( \text{diam}(X_j) = \max_{x_j,x_j' \in X_j} ||x_j-x_j'||_2 \) which are non-zero for finite action and state space.

Note that since we need \( d_1 d_2 + d_3 < 1 \), thus, the value of \( c_2 \) must be small. Intuitively, if the transition probabilities do not change much with the change in the population distribution, the optimal policy will also not change much which leads to a convergence.

Further, if there is a gap between any two actions, the optimal policy would not change much even when the population distributions change.

4 REINFORCEMENT LEARNING BASED ALGORITHM

In this section, we provide a reinforcement learning based algorithm to find policies for agents when they are unaware of the reward and the transition probability dynamics.

4.1 Proposed Algorithm

We propose a policy gradient based algorithm to find a MMFE. The policy gradient mechanism is a model free approach. Compared to [24], we characterize the conditions, which are easier to verify for a policy gradient mechanism to converge. We, first, characterize a simulator which is essential for the algorithm.

**Simulator:** We assume that the algorithm has access to a simulator which will give samples of next states according to the transition probability \( \tau^j(x_j,a_j,z_1,z_2) \) and the corresponding rewards for both \( j = 1,2 \). This is a standard assumption in the literature for computing optimal RL algorithm.

In the policy gradient mechanism, we represent the policies of agents of type \( j \), \( \pi^j \) as parameterized by the parameter \( \theta_j \). Specifically, \( \pi^j \) gives a choice of action for an agent of type \( j \) for every state. One of the classical examples is Boltzmann policy. In the Boltzmann policy, \( \pi^j \) gives the probability of taking action \( a_j \) when the state is \( a_j \) via the following function
\[
h(x_j,a_j,\theta_j) = \sum_{a_k} e^{h(x_j,a_k,\theta_j)}\]
where
\[
h(x_j,a_k,\theta_j) = \sum_i \theta_j(i) f_i^j(x_j,a_k)\]
and \( f_i^j \) is a function of the state \( x_j \) and action \( a_k \) for agents of type \( j \).

Now, we specify an algorithm (Algorithm 2) to provide an unbiased estimator of estimating the \( Q \)-function for the agents which we then use to define our policy gradient mechanism. Algorithm 2 provides an unbiased estimator of \( J^j(x,a,\pi^j_0,z_j) \). It gives an unbiased estimator because of the random horizon setting via Monte-carlo rollout [30]. Note that since the horizon \( T \) is chosen randomly from a geometric distribution of parameter \( 1 - \gamma^{1/2} \), the probability that the horizon is of length at least \( t \) is given by \( \gamma^{t/2} \). Now, combined with the fact that the reward is multiplied by \( \gamma^{t/2} \) in the algorithm, we obtain the unbiased estimator of the \( Q \)-function. While in practice, usually finite deterministic horizon roll-outs are used to estimate infinite \( Q \)-function. However, it would create bias in estimating \( Q \)-function and hence it would end-up in creating biases in policy gradient method. In [24], authors also relied on unbiased estimator of \( Q \)-function without explicitly stating how to find it.

Note that in the MMFE, the policy must be optimal in the value function starting from each initial state. However, it is computationally difficult to verify whether a policy is optimal for starting from each initial state. Rather, we characterize the MMFE which provides an optimal policy for agents for the expected value function where the expectation is taken with respect to the population dynamics. Thus, we seek to compute policy such that

- \( \pi^j \) maximizes \( J^j(\theta_j,z_1,z_2) = \mathbb{E}_{x_j,z_j}[V^j(x_j,\pi^j_0,z_1,z_2)] \)
  for \( j = 1,2 \)
- \( (z_1,z_2) \) both satisfy (8).

Since \( \pi^j \) maximizes \( J^j \), thus, from the first order of stationary condition, \( \nabla J^j(\theta_0,z_j) = 0 \) and \( \nabla J^j(\theta_0,z_1,z_2) = 0 \).
From the policy-gradient method, we can represent the gradient of $J^j$ in the following form for $j = 1, 2$

$$\nabla J^j(\theta_j, z_1, z_2) = \sum_{y_0} z_j(y_0) \sum_{t=0}^{\infty} \sum_{y} p^j(y_t = y|y_0, \pi_{\theta_j}, z_1, z_2)$$

$$\sum_{a} \nabla \pi_{\theta_j}(a|y)Q^j(y, a, \pi_{\theta_j}, z_1, z_2)$$

$$= \sum_{y_0} z_j(y_0) \sum_{t=0}^{\infty} \sum_{y} p^j(y_t = y|y_0, \pi_{\theta_j}, z_1, z_2)$$

$$\sum_{a} \pi_{\theta_j}(a|y)\nabla \log \pi_{\theta_j}(a|y)Q^j(y, a, \pi_{\theta_j}, z_1, z_2)$$

(16)

where

$$p^j(y_t = y|y_0, \pi_{\theta_j}, z_1, z_2) = \sum_{k=0}^{t-1} \pi_{\theta_j}(b^k|y_k)\tau^j(y_{k+1}|y_k, b^k, z_1, z_2)$$

i.e., the probability that the state is $y$ at time $t$ for agent of type $j$ under the policy $\pi_{\theta_j}$ and when it starts from the state $y_0$. The first expression comes from the standard policy gradient mechanism.

The second equality comes from the fact that $\nabla \log(x) = \frac{1}{x}$.

Now, we describe the policy gradient mechanism in order to find policies for leaders and followers. As an extension of the result of [30], we obtain

**Lemma 2.** For a given $z_1, z_2$, let

$$\nabla J^j(\theta_j) = \frac{1}{1 - \gamma} \hat{Q}^j(x_j|T, a_j, T, \pi_{\theta_j}, z_1, z_2) \nabla \log(\pi_{\theta_j}(a_T|x_T))$$

for $j = 1, 2$. Then, for any $\theta_j$, for $j = 1, 2$,

$$\mathbb{E}[\nabla J^j(\theta_j, z_1, z_2)] = \nabla J^j(\theta_j)$$

(18)

where the expectation is taken over the random sample from $z_1$ and $z_2$, random horizon $T$, the trajectory along $(x_1,0, a_{1,0}, x_2,0, a_{2,0}, \ldots, x_{1,T-1}, a_{1,T-1}, x_{2,T-1}, a_{2,T})$.

The above lemma indicates that $\nabla J^j$ for $j = 1, 2$ is an unbiased estimator of $\nabla J^j$. Intuitively, from Algorithm 1 note that $\hat{Q}^j(x_j|T, a_j, T, \pi_{\theta_j}, z_1, z_2)$ gives an unbiased estimator of $Q^j(x_j, a_j, T, \pi_{\theta_j}, z_1, z_2)$. Thus, from (16) it follows that $\nabla J^j$ gives unbiased estimators of $\nabla J^j$ for $j = 1, 2$.

In Algorithm 2, we first fix $z_1, z_2$ and try to find the optimal value functions for leaders and followers for the given $z_1, z_2$ similar to Step 1 in Section 3. The parameters are updated using these unbiased estimators in Algorithm 2 and thus, it would converge when a stationary point exists. Once the inner loop converges, we compute new $z'_1$ and $z'_2$ based on the optimal policies acted upon $z_1$ and $z_2$ similar to Step 2. When $z'_1$ and $z'_2$ become close to $z_1, z_2$ respectively, we stop the algorithm similar to Step 3.

In order to find optimal policy for a given $z_1, z_2$, note that first a random horizon is selected with Geometric distribution and given the policy and population simulator, next state and action are obtained both for leader and follower. This is done to get an unbiased estimate of $p^j p^j(y_t = y|y_0, \pi_{\theta_j}, z_1, z_2)$. After that the algorithm obtains the unbiased estimator of $Q$-function from Algorithm 1. Finally, the parameters are updated using the unbiased estimator of the gradient.

---

**Algorithm 2 RHPG-MMFE: Random Horizon Policy gradient to find MMFE**

**Input:** Population simulator

**Output:** $\pi_{\theta_j}, j = 1, 2.$

**Initialization:** $z_{1,0}, z_{2,0}$: Initial Population distribution uniform distribution

$m \leftarrow 0, z_{1,m} \leftarrow z_{1,0}, z_{2,m} \leftarrow z_{2,0}$

**while** Until Convergence **do**

**Initialization** $\theta_{j,k}$ set at $0, k \leftarrow 0$, for $j = 1, 2$

**while** Until Convergence do

Sample $x_{j,k}$ from $z_{j,m}$ for $j = 1, 2$

$x_{j,0} \leftarrow x_{j,k}$ for $j = 1, 2$

Draw $T_{k+1}$ from Geometric Distribution with parameter $(1 - \gamma), a_{j,0} \sim \pi_{\theta_{j,k}}(a_{j|x_{j,0}})$ for $j = 1, 2.$

for $t = 0, 1, \ldots, T_{k+1} - 1$ do

Simulate the next state $x_{j,t+1} \sim \tau^j(x_{j,t}, a_{j,t}|z_{1,m}, z_{2,m})$ for $j = 1, 2$ from the simulator.

Simulate the action $a_{j,t+1} \sim \pi_{\theta_{j,k}}(a_{j|x_{j,t+1}})$ for $j = 1, 2$.

end for

Obtain an Estimate of $Q^j(x_{j,T_{k+1}}, a_{j,T_{k+1}}, \pi_{\theta_{j,k}}, z_{1,m}, z_{2,m})$ for $j = 1, 2$ using Algorithm 1, i.e.,

$\hat{Q}^j(x_{1,T_{k+1}}, a_{1,T_{k+1}}, \pi_{\theta_{1,k}}, z_{1,m}, z_{2,m}) = \text{EST}$.

$Q(x_{1,T_{k+1}}, a_{1,T_{k+1}}, \pi_{\theta_{1,k}}, z_{1,m}, z_{2,m}) = \text{EST}$.

$\hat{Q}^j(x_{2,T_{k+1}}, a_{2,T_{k+1}}, \pi_{\theta_{2,k}}, z_{1,m}, z_{2,m}) = \text{EST}$.

$Q(x_{2,T_{k+1}}, a_{2,T_{k+1}}, \pi_{\theta_{2,k}}, z_{1,m}, z_{2,m}) = \text{EST}$.

$\theta_{j,k+1} \leftarrow \theta_{j,k} + a_{j} \hat{Q}^j(\cdot) \nabla \log(\pi_{\theta_{j,k}}(a_{j|x_{j,T_{k+1}}})) \text{ for } j = 1, 2$.

end for

Update iteration $k \leftarrow k + 1$

end while

**until** convergence

for $j = 1, 2$ do

for $l = 1, \ldots, N_j$ do

Sample $x_{j,l}$ from $z_{j,m}$ take action $a_j \sim \pi_{\theta_{j,k}}(a_{j|x_{j,l}})$ simulate the next state $x_{j,m+1}$.

$z_{j,m+1}(x) = \frac{1}{N_j} \sum\{x_{j,l} = x\}$ for $j = 1, 2$

end for

end for

Update outer iteration counter $m \leftarrow m + 1$.

end while

---

**4.2 Convergence of Policy Gradient**

In this section, we prove the convergence of the policy gradient mechanism to a MMFE under certain additional assumptions.

**Assumption 3.**

- The reward function is bounded
- $\pi_{\theta_j}$ is differentiable with respect to $\theta_j$ for $j = 1, 2$. Further, for any $(x_j, a_j)$ for $j = 1, 2$

$$||\nabla \log(\pi_{\theta_j}(a_j|x_j)) - \nabla \log(\pi_{\theta_j}(a_j|x_j))|| \leq L_\theta ||\theta_j^2 - \theta_j^0||$$

for any $\theta_j^0, \theta_j^1

$$||\nabla \log(\pi_{\theta_{\text{theta}}}(a_j|x_j))|| \leq B_\theta$$
A higher state means more vulnerable. Each defender can take an action in order to achieve higher reward in the future. The attacker can not successfully attack a defender. Thus, one takes actions with certainty. When the state is less than a certain value, the probability of taking any action is negligible for both attacker and defender. On the other hand, it makes the virus propagation from one defender to another defender is successfully attack a defender if the attacker's state is lower than the defender. The defender can take an action $b_t = 0$ if it wants to become extremely powerful, i.e., the next state would be 0.

The defender $i$'s state evolves as the following

$$y_{t+1} = \begin{cases} y_t + w_{x_i} & \text{if } b_t^i = 1 \\ 0 & \text{otherwise} \end{cases}$$

where $w_{x_i}$ is a random variable with uniform mass on $\{0, 1/n, \ldots, 1\}$.

Now, the reward model can be described as the following for a defender.

$$r^i(x_t, a_t, z_{1,t}, z_{2,t}) = -g_1y_t - g_2y_t(\max \{E(z_{1,t}) - E(z_{2,t}), 0\}) - (1 - a_t)\lambda_1$$

$\lambda_1$ is the cost of securing the node. Note that a leader will be more vulnerable when its state is high. Further, if the mean value of the states of the followers is small it means that there are more powerful attackers which can attack the defenders easily. Thus, a defender can also be infected if the mean state of the attackers (i.e., $\mathbb{E}(z_{2,t})$) is smaller compared to the mean state of the defenders ($\mathbb{E}(z_{1,t})$).

Likewise, the reward model for an attacker is

$$r^2(y_t, b_t, z_{1,t}, z_{2,t}) = -g_1y_t - g_2y_t(\max \{E(z_{2,t}) - E(z_{1,t}), 0\}) - (1 - b_t)\lambda_2$$

$\lambda_2$ is the cost for attacker to become most powerful. When the state of the follower is high, it would attack a defender; however, it would have smaller chance to breach the defender. On the other hand, it makes the mean state of the defenders is smaller compared to the attackers, the attackers may need to attack more number of defender since the virus propagation from one defender to another defender is small which depletes the energies of the attackers.

5.1.2 Set-up. In our evaluation, we consider that there are $N_1 = 100$ number of defenders and $N_2 = 100$ number of attackers. We set $n = 10$. We set $g_1 = 0.2, g_2 = 0.1, \lambda_1 = \lambda_2 = 0.5$. The parameterized policy we consider is the Boltzman policy where action 0 is taken with probability $1/(1 + \exp(\theta_{1,1}x_t + \theta_{1,2}(1 - x_t)))$ for defender at state $x$. Note that as $\theta_{1,2}$ decreases, defenders take action 0 with a higher probability as the state value $x$ increases.

Similar to the defender, the attacker’s policy is also considered to be of the following form - at state $y$, action 0 is taken with probability $1/(1 + \exp(\theta_{2,1}y + \theta_{2,2}(1 - y)))$.

5.1.3 Results. Fig. 1 shows the probabilities of taking action $a_t = 0$ ($b_t = 0$, resp.) by defender (attacker, resp.) as a function of state. As the state increases, the probability increases. This is because higher state indicates a lower reward for both defender and attacker. A defender will be more vulnerable to the attacker if the state is higher. On the other hand, higher state of an attacker indicates that the attacker can not successfully attack a defender. Thus, one takes action in order to achieve higher reward in the future. Fig. 1 shows a threshold type of behavior for both attacker and defender. When the state is less than 0.3, the probability of taking any action is negligible for both attacker and defender. On the other hand, when the state exceeds 0.5 both the attacker and defender take actions with certainty. When the state is 0.3, the attacker takes an action with slightly higher probability because of slightly higher mean value of the state of attackers (3.997) compared to the
6 DISCUSSION

6.1 MMFE with both state and action distribution

As we have discussed in Remarks in 4 and 3, we can extend our analysis when the reward, and transition probability both depend on the action distribution of the players as well. We only need to replace $z_{j,t}$ with joint state and action distribution of the agents of type $j$ ($z_{j,t}, v_{j,t}$) for $j = 1, 2$. The algorithm also converges under additional condition with replacing $z_j$ with ($z_j, v_j$).

6.2 More than two types of agents

We can set the extending with more than two types of agents where each type has infinite number of players. For example, when there are more than two types of agents, the reward function and transition probabilities depend on state distribution functions of each type of agents and the individual state and action. Hence, in the stationary MMFE, one can compute the optimal policy for each type of agents while considering the other agents’ distributions as fixed. After the optimal policy is found for a fixed set of population distribution for each set of agents, we update the population state distributions and we repeat until we reach the convergence.

6.3 Non-stationary MMFE

Similar to the stationary MMFE, a non-stationary MMFE can also be obtained. First, we fix the sequence of population state distributions $z_{j,t}$ for $j = 1, 2$ and $t = 0, \cdots, \infty$ and obtain the optimal policies for the agents of both the types (Step 1). We then update the population state distribution $z'_{j,t}$ across all the time-horizon for $j = 1, 2$. If $z'_{1,t}$ and $z'_{2,t}$ do not match with $z_{1,t}$ and $z_{2,t}$ respectively for every $t$, we then again obtain policies for the updated population state distributions $z'_{j,t}$. (Step 2). We repeat the process until $z'_{j,t}$ becomes equal to $z_{j,t}$ for $j = 1, 2$. Computing a non-stationary MMFE is computationally challenging. The characterization of a computationally efficient RL algorithm to compute a non-stationary MMFE is left for the future.

7 CONCLUSIONS

We study a multi-agent multi-type Markov strategic interaction over a finite time steps for large number of agents of each type. A MMFE is defined as an equilibrium concept. An algorithm for known system dynamics is proposed that achieves a stationary MMFE. The condition under which a stationary MMFE exists is also characterized. A policy gradient based RL algorithm is proposed to obtain the stationary MMFE when the players are unaware of the dynamics.

A PROOF OF THEOREM 3

First, we prove the theorem by assuming that the population simulator gives exactly $z_{l,m+1}$ and $z_{f,m+1}$ from $z_{l,m}$ and $z_{f,m}$, respectively, using $\pi$ and $w$.

Consider $m = M$-th iteration,

From Theorem 2, there exists a $K$ such that $||\theta_{j,K} - \theta_{j,0}|| \leq \delta$ a such that $||\pi_{j,K} - \pi_{j,0}|| \leq \epsilon$. Let $(z_{j,1}^*, z_{j,2}^*)$ are the MFE population distributions. Recall that $\Gamma_1$ and $\Gamma_2$ are the optimal for leaders and followers respectively. Let us denote the policy of follower and leader at $m = M$-th iteration after $K$ number of inner iterations as $\pi_{j,1,K}(z_{l,m}, z_{2,m})$. Now,

\begin{align*}
W_1((z_{l,m+1}, z_{2,m+1}),(z_{1,*}, z_{2,*})) &= W_1(G_2(\pi_{\theta_{1,K}}(z_{l,m}, z_{2,m}), \pi_{\theta_{2,K}}(z_{l,m}, z_{2,m})), \Gamma_2(G_1(z_{1,*}, z_{2,*}, z_{l,m}, z_{2,m})))
&\leq W_1(G_2(G_1(z_{1,*}, z_{2,*}, z_{l,m}, z_{2,m})), \Gamma_2(z_{1,*}, z_{2,*})))
&\leq dW_1((z_{l,m}, z_{2,m}),(z_{1,*}, z_{2,*})) + d_2\epsilon
\end{align*}

where $d = d_1 + d_2$. Hence, we have

\begin{align*}
W_1((z_{l,m+1}, z_{2,m+1}),(z_{1,*}, z_{2,*})) &\leq d^m W_1((z_{l,0}, z_{2,0}), (z_{1,*}, z_{2,*})) + \frac{d_2\epsilon (1 - d^m)}{1 - d}
\end{align*}

Since $d < 1$, thus, there exists $M$ such that $d^m \leq \epsilon$ for $m > M$, and $d_2(1 - d^m) < \epsilon$, hence we have

\begin{align*}
W_1((z_{l,m+1}, z_{2,m+1}),(z_{1,*}, z_{2,*})) &\leq O(\epsilon)
\end{align*}

Now, we show the result for the weak population simulator which only gives the next state of the agents a for a given state of the agent, action, and population distribution. Let us denote the actual population dynamics be $z_{l,1,m+1}^*$ and $z_{2,2,m+1}^*$ and the population dynamics returned by Algorithm 2 are $z_{l,m+1}$ and $z_{2,m+1}$, respectively.
Now, 
\[ W_1((z_1, m+1), (z_2, m+1), (z_1, z_2, z_3)) \leq \]
\[ W_1 (\Gamma(z_1, z_2, z_3) \setminus \Gamma_0(z_1, z_2, z_3)) \]
\[ + W_1((z_1, m+1), (z_2, m+1), (z_1, z_2, z_3)) \]
\[ + W_1((z_1, m+1), (z_2, m+1), (z_1, z_2, z_3)) \]
\[ = W_1 (\Gamma(z_1, z_2, z_3) \setminus \Gamma_0(z_1, z_2, z_3)) \]
\[ Γ_0(z_1, z_2, z_3) = (z_1, z_2, z_3, m) \]
\[ Γ(z_1, z_2, z_3) = (z_1, z_2, z_3) \]
\[ (23) \]

where
\[ \mathbb{E}(z_{1,m+1}(x')) = \]
\[ \sum_x \sum_a z_{1,m}(x) t(x'|x, a, z_{1,m}, z_{2,m}) \]
\[ \mathbb{E}(z_{2,m+1}(y')) = \]
\[ \sum_y \sum_a z_{2,m}(y) t(y'|y, a, z_{1,m}, z_{2,m}) \]
\[ \mathbb{E}(z_{2,m+1}(y')) = \]
\[ \sum_y \sum_a z_{2,m}(y) t(y'|y, a, z_{1,m}, z_{2,m}) \]
\[ (24) \]

From Hoeffding’s inequality, we have
\[ \Pr(|z_{1,m+1}(x') - \mathbb{E}(z_{1,m+1}(x'))| > t) \leq 2 \exp(-N_1 t^2) \]
\[ \Pr(|z_{2,m+1}(y') - \mathbb{E}(z_{2,m+1}(y'))| > t) \leq 2 \exp(-N_2 t^2) \]

Hence, we can bound the second expression in the right hand side of (23) by \( \epsilon \) with high probability for large enough \( N_1 \) and \( N_2 \). Since we have already obtained bound for the first expression in the right hand side. Hence, with a high probability, we prove the convergence as \( K \to \infty \), and \( N_1, N_2 \to \infty \). \( \square \)

REFERENCES
[1] Berkay Anahtarci, Can Deha Karlikuz, and Naci Saldi. 2019. Fitted Q-learning in mean-field games. arXiv preprint arXiv:1912.13309 (2019).
[2] Michel Benaim and Jean-Yves Le Boudec. 2008. A class of mean field interaction models for computer and communication systems. Communications in Information & Systems
[3] Lucian Buşoniu, Robert Babuška, and Bart De Schutter. 2010. Multi-agent Reinforcement Learning: An Overview. Springer Berlin Heidelberg, Berlin, Heidelberg, 183–221. https://doi.org/10.1007/978-3-642-14435-6_7
[4] René Carmona, Mathieu Laurière, and Zongjun Tan. 2019. Linear-quadratic mean-field reinforcement learning: convergence of policy gradient methods. arXiv preprint arXiv:1910.12802 (2019).
[5] René Carmona, Mathieu Laurière, and Zongjun Tan. 2019. Model-free mean-field reinforcement learning: mean-field MDP and mean-field Q-learning. arXiv preprint arXiv:1910.07498 (2019).
[6] Zuyue Fu, Zhoaoran Yang, Yongxin Chen, and Zhoaoran Wang. 2019. Actor-critic provably finds Nash equilibria of linear-quadratic mean-field games. arXiv preprint arXiv:1910.07498 (2019).
[7] Xin Guo, Anran Hu, Renyuan Xu, and Junzi Zhang. 2019. Learning mean-field games. In Advances in Neural Information Processing Systems. 4966–4976.
[8] Xin Guo, Anran Hu, Renyuan Xu, and Junzi Zhang. 2020. A General Framework for Learning Mean-Field Games. arXiv preprint arXiv:2003.06069 (2020).
[9] Christopher Harris, Philip Reny, and Arthur Robson. 1995. The existence of subgame-perfect equilibrium in continuous games with almost perfect information: A case for public randomization. Econometrica: Journal of the Econometric Society (1995), 507–544.
[10] Pablo Hernández-Leal, Bilal Kartal, and Matthew E Taylor. 2019. A survey and critique of multiagent deep reinforcement learning. Autonomous Agents and Multi-Agent Systems 33, 6 (2019), 750–797.
[11] Junling Hu, Michael P Wellman, et al. 1998. Multiagent reinforcement learning: theoretical framework and an algorithm. In ICMC. Vol. 98. Citeseer, 242–250.
[12] 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.
[13] 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.