Mean Field Multi-Agent Reinforcement Learning

Yaodong Yang [1] **Rui Luo** [1] **Minne Li** [1] **Ming Zhou** [2] **Weinan Zhang** [2] **Jun Wang** [1]

Abstract

Existing multi-agent reinforcement learning methods are limited typically to a small number of agents. When the agent number increases largely, the learning becomes intractable due to the curse of the dimensionality and the exponential growth of agent interactions. In this paper, we present *Mean Field Reinforcement Learning* where the interactions within the population of agents are approximated by those between a single agent and the average effect from the overall population or neighboring agents; the interplay between the two entities is mutually reinforced: the learning of the individual agent's optimal policy depends on the dynamics of the population, while the dynamics of the population change according to the collective patterns of the individual policies. We develop practical mean field Q-learning and mean field Actor-Critic algorithms and analyze the convergence of the solution to Nash equilibrium. Experiments on Gaussian squeeze, Ising model, and battle games justify the learning effectiveness of our mean field approaches. In addition, we report the first result to solve the Ising model via model-free reinforcement learning methods.

1. Introduction

Multi-agent reinforcement learning (MARL) is concerned with a set of autonomous agents that share a common environment (Busoniu et al., 2008). Learning in MARL is fundamentally difficult since agents not only interact with the environment but also with each other. Independent Q-learning (Tan, 1993) that considers other agents as a part of the environment often fails as the multi-agent setting breaks the theoretical convergence guarantee and makes the learning unstable: changes in the policy of one agent will affect that of the others, and vice versa (Matignon et al., 2012).

[1]University College London, London, U.K. [2]Shanghai Jiao Tong University, Shanghai, China. Correspondence to: Yaodong Yang <yaodong.yang@cs.ucl.ac.uk>, Jun Wang <j.wang@cs.ucl.ac.uk>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

Instead, accounting for the extra information from *conjecturing* the policies of other agents is beneficial to each single learner (Foerster et al., 2017; Lowe et al., 2017a). Studies show that an agent who learns the effect of joint actions has better performance than those who do not in many scenarios, including cooperative games (Panait & Luke, 2005), zero-sum stochastic games (Littman, 1994), and general-sum stochastic games (Littman, 2001; Hu & Wellman, 2003).

The existing equilibrium-solving approaches, although principled, are only capable of solving a handful of agents (Hu & Wellman, 2003; Bowling & Veloso, 2002). The computational complexity of directly solving (Nash) equilibrium would prevent them from applying to the situations with a large group or even a population of agents. Yet, in practice, many cases do require strategic interactions among a large number of agents, such as the gaming bots in Massively Multiplayer Online Role-Playing Game (Jeong et al., 2015), the trading agents in stock markets (Troy, 1997), or the online advertising bidding agents (Wang et al., 2017).

In this paper, we tackle MARL when a large number of agents co-exist. We consider a setting where each agent is directly interacting with a finite set of other agents; through a chain of direct interactions, any pair of agents is interconnected globally (Blume, 1993). The scalability is solved by employing Mean Field Theory (Stanley, 1971) – the interactions within the population of agents are approximated by that of a single agent played with the average effect from the overall (local) population. The learning is mutually reinforced between two entities rather than many entities: the learning of the individual agent's optimal policy is based on the dynamics of the agent population, meanwhile, the dynamics of the population is updated according to the individual policies. Based on such formulation, we develop practical mean field Q-learning and mean field Actor-Critic algorithms, and discuss the convergence of our solution under certain assumptions. Our experiment on a simple multi-agent resource allocation shows that our mean field MARL is capable of learning over many-agent interactions when others fail. We also demonstrate that with temporal-difference learning, mean field MARL manages to learn and solve the Ising model without even explicitly knowing the energy function. At last, in a mixed cooperative-competitive battle game, we show that the mean field MARL achieves high winning rates against other baselines previously reported for many agent systems.

2. Preliminary

MARL intersects between reinforcement learning and game theory. The marriage of the two gives rise to the general framework of *stochastic game* (Shapley, 1953).

2.1. Stochastic Game

An N-agent (or, N-player) stochastic game Γ is formalized by the tuple $\Gamma \triangleq \big(\mathcal{S}, \mathcal{A}^1, \ldots, \mathcal{A}^N, r^1, \ldots, r^N, p, \gamma\big)$, where $\mathcal{S}$ denotes the state space, and $\mathcal{A}^j$ is the action space of agent $j \in \{1, \ldots, N\}$. The reward function for agent j is defined as $r^j : \mathcal{S} \times \mathcal{A}^1 \times \cdots \times \mathcal{A}^N \to \mathbb{R}$, determining the immediate reward. The transition probability $p : \mathcal{S} \times \mathcal{A}^1 \times \cdots \times \mathcal{A}^N \to \Omega(\mathcal{S})$ characterizes the stochastic evolution of states in time, with $\Omega(\mathcal{S})$ being the collection of probability distributions over the state space $\mathcal{S}$. The constant $\gamma \in [0, 1)$ represents the reward discount factor across time. At time step t, all agents take actions simultaneously, each receives the immediate reward r_t^j as a consequence of taking the previous actions.

The agents choose actions according to their policies, also known as strategies. For agent j, the corresponding policy is defined as $\pi^j : \mathcal{S} \to \Omega(\mathcal{A}^j)$, where $\Omega(\mathcal{A}^j)$ is the collection of probability distributions over agent j's action space $\mathcal{A}^j$. Let $\boldsymbol{\pi} \triangleq [\pi^1, \ldots, \pi^N]$ denote the joint policy of all agents; we assume, as one usually does, $\boldsymbol{\pi}$ to be time-independent, which is referred to be *stationary*. Provided an initial state s, the value function of agent j under the joint policy $\boldsymbol{\pi}$ is written as the expected cumulative discounted future reward:

$$v_{\boldsymbol{\pi}}^j(s) = v^j(s; \boldsymbol{\pi}) = \sum_{t=0}^{\infty} \gamma^t \mathbb{E}_{\boldsymbol{\pi}, p}\big[r_t^j | s_0 = s, \boldsymbol{\pi}\big]. \quad (1)$$

The Q-function (or, the action-value function) can then be defined within the framework of N-agent game based on the Bellman equation given the value function in Eq. (1) such that the Q-function $Q_{\boldsymbol{\pi}}^j : \mathcal{S} \times \mathcal{A}^1 \times \cdots \times \mathcal{A}^N \to \mathbb{R}$ of agent j under the joint policy $\boldsymbol{\pi}$ can be formulated as

$$Q_{\boldsymbol{\pi}}^j(s, \boldsymbol{a}) = r^j(s, \boldsymbol{a}) + \gamma \mathbb{E}_{s' \sim p}[v_{\boldsymbol{\pi}}^j(s')]\,, \quad (2)$$

where s' is the state at the next time step. The value function $v_{\boldsymbol{\pi}}^j$ can be expressed in terms of the Q-function in Eq. (2) as

$$v_{\boldsymbol{\pi}}^j(s) = \mathbb{E}_{\boldsymbol{a} \sim \boldsymbol{\pi}}\big[Q_{\boldsymbol{\pi}}^j(s, \boldsymbol{a})\big]. \quad (3)$$

The Q-function for N-agent game in Eq. (2) extends the formulation for single-agent game by considering the joint action taken by all agents $\boldsymbol{a} \triangleq [a^1, \ldots, a^N]$, and by taking the expectation over the joint action in Eq. (3).

We formulate MARL by the stochastic game with a discrete-time non-cooperative setting, *i.e.* no explicit coalitions are considered. The game is assumed to be incomplete but to have perfect information (Littman, 1994), *i.e.* each agent knows neither the game dynamics nor the reward functions of others, but it is able to observe and react to the previous actions and the resulting immediate rewards of other agents.

2.2. Nash Q-learning

In MARL, the objective of each agent is to learn an optimal policy to maximize its value function. Optimizing the $v_{\boldsymbol{\pi}}^j$ for agent j depends on the joint policy $\boldsymbol{\pi}$ of all agents, the concept of *Nash equilibrium* in stochastic games is therefore of great importance (Hu & Wellman, 2003). It is represented by a particular joint policy $\boldsymbol{\pi}_* \triangleq [\pi_*^1, \ldots, \pi_*^N]$ such that for all $s \in \mathcal{S},\ j \in \{1, \ldots, N\}$ and all valid π^j, it satisfies

$$v^j(s; \boldsymbol{\pi}_*) = v^j(s; \pi_*^j, \boldsymbol{\pi}_*^{-j}) \geq v^j(s; \pi^j, \boldsymbol{\pi}_*^{-j}).$$

Here we adopt a compact notation for the joint policy of all agents except j as $\boldsymbol{\pi}_*^{-j} \triangleq [\pi_*^1, \ldots, \pi_*^{j-1}, \pi_*^{j+1}, \ldots, \pi_*^N]$.

In a Nash equilibrium, each agent acts with the *best response* π_*^j to others, provided that all other agents follow the policy $\boldsymbol{\pi}_*^{-j}$. It has been shown that, for a N-agent stochastic game, there is at least one Nash equilibrium with stationary policies (Fink et al., 1964). Given a Nash policy $\boldsymbol{\pi}_*$, the Nash value function $\boldsymbol{v}^{\texttt{Nash}}(s) \triangleq [v_{\boldsymbol{\pi}_*}^1(s), \ldots, v_{\boldsymbol{\pi}_*}^N(s)]$ is calculated with all agents following $\boldsymbol{\pi}_*$ from the initial state s onward.

Nash Q-learning (Hu & Wellman, 2003) defines an iterative procedure with two alternating steps for computing the Nash policy: 1) solving the Nash equilibrium of the current stage game defined by $\{\boldsymbol{Q}_t\}$ using the Lemke-Howson algorithm (Lemke & Howson, 1964), 2) improving the estimation of the Q-function with the new Nash equilibrium value. It can be proved that under certain assumptions, the Nash operator $\mathscr{H}^{\texttt{Nash}}$ defined by the following expression

$$\mathscr{H}^{\texttt{Nash}}\boldsymbol{Q}(s, \boldsymbol{a}) = \mathbb{E}_{s' \sim p}\left[\, \boldsymbol{r}(s, \boldsymbol{a}) + \gamma \boldsymbol{v}^{\texttt{Nash}}(s') \,\right] \quad (4)$$

forms a contraction mapping, where $\boldsymbol{Q} \triangleq [Q^1, \ldots, Q^N]$, and $\boldsymbol{r}(s, \boldsymbol{a}) \triangleq [r^1(s, \boldsymbol{a}), \ldots, r^N(s, \boldsymbol{a})]$. The Q-function will eventually converge to the value received in a Nash equilibrium of the game, referred to as the *Nash Q-value*.

3. Mean Field MARL

The dimension of joint action $\boldsymbol{a}$ grows proportionally *w.r.t.* the number of agents N. As all agents act strategically and evaluate simultaneously their value functions based on the joint actions, it becomes infeasible to learn the standard Q-function $Q^j(s, \boldsymbol{a})$. To address this issue, we factorize the Q-function using only the pairwise local interactions:

$$Q^j(s, \boldsymbol{a}) = \frac{1}{N^j} \sum_{k \in \mathcal{N}(j)} Q^j(s, a^j, a^k)\,, \quad (5)$$

where $\mathcal{N}(j)$ is the index set of the neighboring agents of agent j with the size $N^j = |\mathcal{N}(j)|$ determined by the settings of different applications. It is worth noting that the pairwise approximation of the agent and its neighbors, while significantly reducing the complexity of the interactions among agents, still preserves global interactions between any pair of agents implicitly (Blume, 1993). Similar approaches can be found in factorization machine (Rendle, 2012) and learning to rank (Cao et al., 2007).

3.1. Mean Field Approximation

The pairwise interaction $Q^j(s, a^j, a^k)$ as in Eq. (5) can be approximated using the mean field theory (Stanley, 1971). Here we consider discrete action spaces, where the action a^j of agent j is a discrete categorical variable represented as the one-hot encoding with each component indicating one of the D possible actions: $a^j \triangleq [a_1^j, \dots, a_D^j]$. We calculate the *mean* action $\bar{a}^j$ based on the neighborhood $\mathcal{N}(j)$ of agent j, and express the one-hot action a^k of each neighbor k in terms of the sum of $\bar{a}^j$ and a small fluctuation $\delta a^{j,k}$ as

$$a^k = \bar{a}^j + \delta a^{j,k}, \qquad \text{where } \bar{a}^j = \frac{1}{N^j} \sum_k a^k, \tag{6}$$

where $\bar{a}^j \triangleq [\bar{a}_1^j, \dots, \bar{a}_D^j]$ can be interpreted as the empirical distribution of the actions taken by agent j's neighbors. By Taylor's theorem, the pairwise Q-function $Q^j(s, a^j, a^k)$, if twice-differentiable *w.r.t.* the action a^k taken by neighbor k, can be expended and expressed as

$$\begin{aligned}
Q^j(s, \boldsymbol{a}) &= \frac{1}{N^j} \sum_k Q^j(s, a^j, a^k) \\
&= \frac{1}{N^j} \sum_k \Big[Q^j(s, a^j, \bar{a}^j) + \nabla_{\bar{a}^j} Q^j(s, a^j, \bar{a}^j) \cdot \delta a^{j,k} \\
&\qquad + \frac{1}{2} \delta a^{j,k} \cdot \nabla^2_{\tilde{a}^{j,k}} Q^j(s, a^j, \tilde{a}^{j,k}) \cdot \delta a^{j,k} \Big] \\
&= Q^j(s, a^j, \bar{a}^j) + \nabla_{\bar{a}^j} Q^j(s, a^j, \bar{a}^j) \cdot \Big[\frac{1}{N^j} \sum_k \delta a^{j,k} \Big] \\
&\qquad + \frac{1}{2N^j} \sum_k \Big[\delta a^{j,k} \cdot \nabla^2_{\tilde{a}^{j,k}} Q^j(s, a^j, \tilde{a}^{j,k}) \cdot \delta a^{j,k} \Big] \qquad (7) \\
&= Q^j(s, a^j, \bar{a}^j) + \frac{1}{2N^j} \sum_k R^j_{s,a^j}(a^k) \;\approx\; Q^j(s, a^j, \bar{a}^j), \qquad (8)
\end{aligned}$$

where $R^j_{s,a^j}(a^k) \triangleq \delta a^{j,k} \cdot \nabla^2_{\tilde{a}^{j,k}} Q^j(s, a^j, \tilde{a}^{j,k}) \cdot \delta a^{j,k}$ denotes the Taylor polynomial's remainder with $\tilde{a}^{j,k} = \bar{a}^j + \epsilon^{j,k} \delta a^{j,k}$ and $\epsilon^{j,k} \in [0, 1]$. In Eq. (7), $\sum_k \delta a^k = 0$ by Eq. (6) such that the first-order term is dropped. From the perspective of agent j, the action a^k in the second-order remainders $R^j_{s,a^j}(a^k)$ is chosen based on the external action distribution of agent k, $R^j_{s,a^j}(a^k)$ is thus essentially a random variable. In fact, one can further prove that the remainder $R^j_{s,a^j}(a^k)$ is bounded within a symmetric interval $[-2M, 2M]$ under the mild condition of the Q-function $Q^j(s, a^j, a^k)$ being M-smooth (*e.g.* the linear function); as a result, $R^j_{s,a^j}(a^k)$ acts as a small fluctuation near zero. To stay self-contained, the derivation of the bound is put in the Appendix B. With the assumptions of homogeneity and locality on all agents within the neighborhood, the remainders tend to cancel each other, leading to the left term of $Q^j(s, a^j, \bar{a}^j)$ in Eq. (8).

As illustrated in Fig. 1, with the mean field approximation, the pairwise interactions $Q^j(s, a^j, a^k)$ between agent j and each neighboring agent k are simplified as that between j, the *central* agent, and the virtual *mean* agent, that is abstracted by the mean effect of all neighbors within

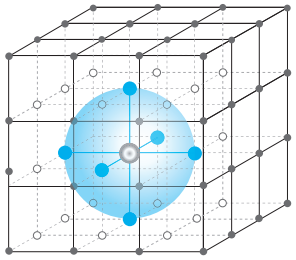


Figure 1: Mean field approximation. Each agent is represented as a node in the grid, which is only affected by the **mean** effect from its **neighbors** (the blue area). Many-agent interactions are effectively converted into two-agent interactions.

j's neighborhood. The interaction is thus simplified and expressed by the mean field Q-function $Q^j(s, a^j, \bar{a}^j)$ in Eq. (8). During the learning phase, given an experience $e = \big(s, \{a^k\}, \{r^j\}, s'\big)$, the mean field Q-function is updated in a recurrent manner as

$$Q^j_{t+1}(s, a^j, \bar{a}^j) = (1 - \alpha) Q^j_t(s, a^j, \bar{a}^j) + \alpha [r^j + \gamma v^j_t(s')], \tag{9}$$

where α_t denotes the learning rate, and $\bar{a}^j$ is the mean action of all neighbors of agent j as defined in Eq. (6). The mean field value function $v^j_t(s')$ for agent j at time t in Eq. (9) is

$$v^j_t(s') = \sum_{a^j} \pi^j_t(a^j | s', \bar{a}^j) \mathbb{E}_{\bar{a}^j(\boldsymbol{a}^{-j}) \sim \boldsymbol{\pi}_t^{-j}} \Big[Q^j_t(s', a^j, \bar{a}^j) \Big], \tag{10}$$

As shown in Eqs. (9) and (10), with the mean field approximation, the MARL problem is converted into that of solving for the central agent j's best response π^j_t *w.r.t.* the mean action $\bar{a}^j$ of all j's neighbors, which represents the action distribution of all neighboring agents of the central agent j.

We introduce an iterative procedure in computing the best response π^j_t of each agent j. In the stage game $\{\boldsymbol{Q}_t\}$, the mean action $\bar{a}^j$ of all j's neighbors is first calculated by averaging the actions a^k taken by j's N^j neighbors from the policies π^k_t parametrized by their previous mean actions $\bar{a}^k_-$

$$\bar{a}^j = \frac{1}{N^j} \sum_k a^k, \; a^k \sim \pi^k_t(\cdot | s, \bar{a}^k_-), \tag{11}$$

With each $\bar{a}^j$ calculated as in Eq. (11), the policy π^j_t changes consequently due to the dependence on the current $\bar{a}^j$. The new Boltzmann policy is then determined for each j that

$$\pi^j_t(a^j | s, \bar{a}^j) = \frac{\exp\big(\beta Q^j_t(s, a^j, \bar{a}^j)\big)}{\sum_{a^{j'} \in \mathcal{A}^j} \exp\big(\beta Q^j_t(s, a^{j'}, \bar{a}^j)\big)}. \tag{12}$$

By iterating Eqs. (11) and (12), the mean actions $\bar{a}^j$ and the corresponding policies π^j_t for all agents improves alternatively. In spite of lacking an intuitive impression of being stationary, in the following subsections, we will show that the mean action $\bar{a}^j$ will be equilibrated at an unique point after several iterations, and hence the policy π^j_t converges.

To distinguish from the Nash value function $\boldsymbol{v}^{\texttt{Nash}}(s)$ in Eq. (4), we denote the mean field value function in Eq. (10) as $\boldsymbol{v}^{\texttt{MF}}(s) \triangleq [v^1(s), \dots, v^N(s)]$. With $\boldsymbol{v}^{\texttt{MF}}$ assembled, we now define the mean field operator $\mathscr{H}^{\texttt{MF}}$ in the form of

$$\mathcal{H}^{\texttt{MF}}\boldsymbol{Q}(s,\boldsymbol{a}) = \mathbb{E}_{s'\sim p}\left[\, \boldsymbol{r}(s,\boldsymbol{a}) + \gamma \boldsymbol{v}^{\texttt{MF}}(s') \,\right]. \qquad (13)$$

In fact, we can prove that $\mathcal{H}^{\texttt{MF}}$ forms a contraction mapping; that is, one updates $\boldsymbol{Q}$ by iteratively applying the mean field operator $\mathcal{H}^{\texttt{MF}}$, the mean field Q-function will eventually converge to the Nash Q-value under certain assumptions.

3.2. Implementation

We can implement the mean field Q-function in Eq. (8) by universal function approximators such as neural networks, where the Q-function is parameterized with the weights ϕ. The update rule in Eq. (9) can be reformulated as weights adjustment. For off-policy learning, we exploit either standard Q-learning (Watkins & Dayan, 1992) for discrete action spaces or DPG (Silver et al., 2014) for continuous action spaces. Here we focus on the former, which we call MF-Q.

In MF-Q, agent j is trained by minimizing the loss function

$$\mathcal{L}(\phi^j) = \left(y^j - Q_{\phi^j}(s, a^j, \bar{a}^j)\right)^2,$$

where $y^j = r^j + \gamma\, v^{\texttt{MF}}_{\phi^j_-}(s')$ is the target mean field value calculated with the weights ϕ^j_-. Differentiating $\mathcal{L}(\phi^j)$ gives

$$\nabla_{\phi^j}\mathcal{L}(\phi^j) = \Big(y^j - Q_{\phi^j}(s, a^j, \bar{a}^j)\Big)\nabla_{\phi^j} Q_{\phi^j}(s, a^j, \bar{a}^j), \quad (14)$$

which enables the gradient-based optimizers for training.

Instead of setting up Boltzmann policy using the Q-function as in MF-Q, we can explicitly model the policy by neural networks with the weights θ, which leads to the on-policy actor-critic method (Konda & Tsitsiklis, 2000) that we call MF-AC. The policy network π_{θ^j}, *i.e.* the actor, of MF-AC is trained by the sampled policy gradient:

$$\nabla_{\theta^j}\mathcal{J}(\theta^j) \approx \nabla_{\theta^j}\log\pi_{\theta^j}(s) Q_{\phi^j}(s, a^j, \bar{a}^j)\Big|_{a=\pi_{\theta^j}(s)}.$$

The critic of MF-AC follows the same setting for MF-Q with Eq. (14). During the training of MF-AC, one needs to alternatively update ϕ and θ until convergence. We illustrate the MF-Q iterations in Fig. 2, and present the pesudocode for both MF-Q and MF-AC in Appendix A.

3.3. Proof of Convergence

We now prove the convergence of $\boldsymbol{Q}_t \triangleq [Q^1_t, \ldots, Q^N_t]$ to the Nash Q-value $\boldsymbol{Q}_* = [Q^1_*, \ldots, Q^N_*]$ as the iterations of MF-Q is applied. The proof is presented by showing that the mean field operator $\mathcal{H}^{\texttt{MF}}$ in Eq. (13) forms a contraction mapping with the fixed point at $\boldsymbol{Q}_*$ under the main assumptions. We start from introducing the assumptions:

Assumption 1. *Each action-value pair is visited infinitely often, and the reward is bounded by some constant K.*

Assumption 2. *Agent's policy is Greedy in the Limit with Infinite Exploration (GLIE). In the case with the Boltzmann policy, the policy becomes greedy* w.r.t. *the Q-function in the limit as the temperature decays asymptotically to zero.*

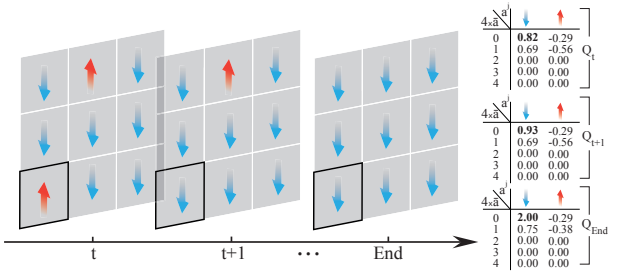


Figure 2: MF-Q iterations on a 3×3 stateless toy example. The goal is to coordinate the agents to an agreed direction. Each agent has two choices of actions: *up* $\uparrow$ or *down* $\downarrow$. The reward of each agent's staying in the same direction as its $[0, 1, 2, 3, 4]$ neighbors are $[-2.0, -1.0, 0.0, 1.0, 2.0]$, respectively. The neighbors are specified by the four directions on the grid with cyclic structure on all directions, *e.g.* the first row and the third row are adjacent. The reward for the highlighted agent j on the bottom left at time $t+1$ is 2.0, as all neighboring agents stay down in the same time. We listed the Q-tables for agent j at three time steps where $\bar{a}^j$ is the percentage of neighboring ups. Following Eq. 9, we have $Q^j_{t+1}(\uparrow, \bar{a}^j = 0) = Q^j_t(\uparrow, \bar{a}^j = 0) + \alpha[r^j - Q^j_t(\uparrow, \bar{a}^j = 0)] = 0.82 + 0.1 \times (2.0 - 0.82) = 0.93$. The rightmost plot shows the convergent scenario where the Q-value of staying down is 2.0, which is the largest reward in the environment.

Assumption 3. *For each stage game $[Q^1_t(s), ..., Q^N_t(s)]$ at time t and in state s in training, for all t, s, $j \in \{1, \ldots, N\}$, the Nash equilibrium $\boldsymbol{\pi}_* = [\pi^1_*, \ldots, \pi^N_*]$ is recognized either as 1) the* global optimum *or 2) a* saddle point *expressed as:*

1. $\mathbb{E}_{\boldsymbol{\pi}_*}[Q^j_t(s)] \geq \mathbb{E}_{\boldsymbol{\pi}}[Q^j_t(s)],\ \forall \boldsymbol{\pi} \in \Omega\big(\prod_k \mathcal{A}^k\big)$;
2. $\mathbb{E}_{\boldsymbol{\pi}_*}[Q^j_t(s)] \geq \mathbb{E}_{\pi^j}\mathbb{E}_{\boldsymbol{\pi}^{-j}_*}[Q^j_t(s)],\ \forall \pi^j \in \Omega\big(\mathcal{A}^j\big)$ *and* $\mathbb{E}_{\boldsymbol{\pi}_*}[Q^j_t(s)] \leq \mathbb{E}_{\pi^j_*}\mathbb{E}_{\boldsymbol{\pi}^{-j}}[Q^j_t(s)],\ \forall \boldsymbol{\pi}^{-j} \in \Omega\big(\prod_{k\neq j}\mathcal{A}^k\big)$.

Note that Assumption 3 imposes a strong constraint on every single stage game encountered in training. In practice, however, we find this constraint appears not to be a necessary condition for the learning algorithm to converge. This is in line with the empirical findings in Hu & Wellman (2003).

Our proof is also built upon the two lemmas as follows:

Lemma 1. *Under Assumption 3, the Nash operator $\mathcal{H}^{\texttt{Nash}}$ in Eq.* (4) *forms a contraction mapping on the complete metric space from $\mathbb{Q}$ to $\mathbb{Q}$ with the fixed point being the Nash Q-value of the entire game,* i.e. *$\mathcal{H}^{\texttt{Nash}}_t \boldsymbol{Q}_* = \boldsymbol{Q}_*$.*

Proof. See Theorem 17 in Hu & Wellman (2003). □

Lemma 2. *The random process $\{\Delta_t\}$ defined in $\mathbb{R}$ as*

$$\Delta_{t+1}(x) = (1 - \alpha_t(x))\Delta_t(x) + \alpha_t(x)F_t(x) \qquad (15)$$

converges to zero with probability 1 (w.p.1) *when*

1. $0 \leq \alpha_t(x) \leq 1$, $\sum_t \alpha_t(x) = \infty$, $\sum_t \alpha^2_t(x) < \infty$;

2. $x \in \mathfrak{X}$, *the set of possible states, and* $|\mathfrak{X}| < \infty$;
3. $\|\mathbb{E}[F_t(x)|\mathcal{F}_t]\|_W \leq \gamma\|\Delta_t\|_W + c_t$, *where* $\gamma \in [0,1)$ *and* c_t *converges to zero* w.p.1;
4. $\mathbf{var}[F_t(x)|\mathcal{F}_t] \leq K(1+\|\Delta_t\|_W^2)$ *with constant* $K > 0$.

Here $\mathcal{F}_t$ *denotes the filtration of an increasing sequence of* σ*-fields including the history of processes;* $\alpha_t, \Delta_t, F_t \in \mathcal{F}_t$ *and* $\|\cdot\|_W$ *is a weighted maximum norm (Bertsekas, 2012).*

Proof. See Theorem 1 in Jaakkola et al. (1994) and Corollary 5 Szepesvári & Littman (1999) for detailed derivation. We include it here to stay self-contained. □

By subtracting $\boldsymbol{Q}_*(s,\boldsymbol{a})$ on both sides of Eq. (9), we present the relation from the comparison with Eq. (15) such that

$$\begin{aligned}\boldsymbol{\Delta}_t(x) &= \boldsymbol{Q}_t(s,\boldsymbol{a}) - \boldsymbol{Q}_*(s,\boldsymbol{a}),\\ \boldsymbol{F}_t(x) &= \boldsymbol{r}_t + \gamma \boldsymbol{v}_t^{\texttt{MF}}(s_{t+1}) - \boldsymbol{Q}_*(s_t,\boldsymbol{a}_t),\end{aligned} \tag{16}$$

where $x \triangleq (s_t,\boldsymbol{a}_t)$ denotes the visited state-action pair at time t. In Eq. (15), $\alpha(t)$ is interpreted as the learning rate with $\alpha_t(s',\boldsymbol{a}') = 0$ for any $(s',\boldsymbol{a}') \neq (s_t,\boldsymbol{a}_t)$; this is because that each agent only updates the Q-function with the state s_t and actions $\boldsymbol{a}_t$ visited at time t. Lemma 2 suggests $\Delta_t(x)$'s convergence to zero, which means, if it holds, the sequence of Q's will asymptotically tend to the Nash Q-value $\boldsymbol{Q}_*$.

One last piece to establish the main theorem is the below:

Proposition 1. *Let the metric space be* $\mathbb{R}^N$ *and the metric be* $d(\boldsymbol{a},\boldsymbol{b}) = \sum_j |a^j - b^j|$, *for* $\boldsymbol{a} = [a^j]_1^N, \boldsymbol{b} = [b^j]_1^N$. *If the Q-function is K-Lipschitz continuous* w.r.t. a^j, *then the operator* $\mathcal{B}(a^j) \triangleq \pi^j(a^j|s,\bar{a}^j)$ *in Eq.* (12) *forms a contraction mapping under sufficiently low temperature* β.

Proof. See details in Appendix D due to the space limit. □

Theorem 1. *In a finite-state stochastic game, the* $\boldsymbol{Q}_t$ *values computed by the update rule of MF-Q in Eq.* (9) *converges to the Nash Q-value* $\boldsymbol{Q}_* = [Q_*^1,\ldots,Q_*^N]$, *if Assumptions 1, 2 & 3, and Lemma 2's first and second conditions are met.*

Proof. Let $\mathcal{F}_t$ denote the σ-field generated by all random variables in the history of the stochastic game up to time t: $(s_t,\alpha_t,\boldsymbol{a}_t,r_{t-1},...,s_1,\alpha_1,\boldsymbol{a}_1,\boldsymbol{Q}_0)$. Note that $\boldsymbol{Q}_t$ is a random variable derived from the historical trajectory up to time t. Given the fact that all $\boldsymbol{Q}_\tau$ with $\tau < t$ are $\mathcal{F}_t$-measurable, both $\boldsymbol{\Delta}_t$ and $\boldsymbol{F}_{t-1}$ are therefore also $\mathcal{F}_t$-measurable, which satisfies the measurability condition of Lemma 2.

To apply Lemma 2, we need to show that the mean field operator $\mathscr{H}^{\texttt{MF}}$ meets Lemma 2's third and fourth conditions. For Lemma 2's third condition, we begin with Eq. (16) that

$$\begin{aligned}\boldsymbol{F}_t(s_t,\boldsymbol{a}_t) &= \boldsymbol{r}_t + \gamma \boldsymbol{v}_t^{\texttt{MF}}(s_{t+1}) - \boldsymbol{Q}_*(s_t,\boldsymbol{a}_t)\\ &= \boldsymbol{r}_t + \gamma \boldsymbol{v}_t^{\texttt{Nash}}(s_{t+1}) - \boldsymbol{Q}_*(s_t,\boldsymbol{a}_t)\\ &\quad + \gamma[\boldsymbol{v}_t^{\texttt{MF}}(s_{t+1}) - \boldsymbol{v}_t^{\texttt{Nash}}(s_{t+1})]\\ &= \left[\boldsymbol{r}_t + \gamma \boldsymbol{v}_t^{\texttt{Nash}}(s_{t+1}) - \boldsymbol{Q}_*(s_t,\boldsymbol{a}_t)\right] + C_t(s_t,\boldsymbol{a}_t)\\ &= \boldsymbol{F}_t^{\texttt{Nash}}(s_t,\boldsymbol{a}_t) + \boldsymbol{C}_t(s_t,\boldsymbol{a}_t).\end{aligned} \tag{17}$$

Note the fact that $\boldsymbol{F}_t^{\texttt{Nash}}$ in Eq. (17) is essentially the $\boldsymbol{F}_t$ in Lemma 2 in proving the convergence of the Nash Q-learning algorithm. From Lemma 1, it is straightforward to show that $\boldsymbol{F}_t^{\texttt{Nash}}$ forms a contraction mapping with the norm $\|\cdot\|_\infty$ being the maximum norm on $\boldsymbol{a}$. We thus have for all t that

$$\|\mathbb{E}[\boldsymbol{F}_t^{\texttt{Nash}}(s_t,\boldsymbol{a}_t)|\mathcal{F}_t]\|_\infty \leq \gamma\|\boldsymbol{Q}_t - \boldsymbol{Q}_*\|_\infty = \gamma\|\boldsymbol{\Delta}_t\|_\infty.$$

In meeting the third condition, we obtain from Eq. (17) that

$$\begin{aligned}\|\mathbb{E}[\boldsymbol{F}_t(s_t,\boldsymbol{a}_t)|\mathcal{F}_t]\|_\infty &\leq \|\boldsymbol{F}_t^{\texttt{Nash}}(s_t,\boldsymbol{a}_t)|\mathcal{F}_t\|_\infty + \|\boldsymbol{C}_t(s_t,\boldsymbol{a}_t)|\mathcal{F}_t\|_\infty\\ &\leq \gamma\|\boldsymbol{\Delta}_t\|_\infty + \|\boldsymbol{C}_t(s_t,\boldsymbol{a}_t)|\mathcal{F}_t\|_\infty.\end{aligned} \tag{18}$$

We are left to prove that $c_t = \|\boldsymbol{C}_t(s_t,\boldsymbol{a}_t)|\mathcal{F}_t\|$ converges to zero *w.p.*1. With Assumption 3, for each stage game, all the globally optimal equilibrium(s) share the same Nash value, so does the saddle-point equilibrium(s). Each of the two following results is essentially associated with one of the two mutually exclusive scenarios in Assumption 3:

1. For globally optimal equilibriums, all players obtain the joint maximum values that are unique and identical for all equilibriums according to the definition;
2. Suppose that the stage game $\{\boldsymbol{Q}_t\}$ has two saddle-point equilibriums, $\boldsymbol{\pi}$ and $\boldsymbol{\rho}$. It holds for agent j that
$$\begin{aligned}\mathbb{E}_{\pi^j}\mathbb{E}_{\boldsymbol{\pi}^{-j}}[Q_t^j(s)] &\geq \mathbb{E}_{\rho^j}\mathbb{E}_{\boldsymbol{\pi}^{-j}}[Q_t^j(s)],\\ \mathbb{E}_{\rho^j}\mathbb{E}_{\boldsymbol{\rho}^{-j}}[Q_t^j(s)] &\leq \mathbb{E}_{\rho^j}\mathbb{E}_{\boldsymbol{\pi}^{-j}}[Q_t^j(s)].\end{aligned}$$
By combing the above inequalities, we obtain
$$\mathbb{E}_{\pi^j}\mathbb{E}_{\boldsymbol{\pi}^{-j}}[Q_t^j(s)] \geq \mathbb{E}_{\rho^j}\mathbb{E}_{\boldsymbol{\rho}^{-j}}[Q_t^j(s)].$$
By the definition of saddle points, the above inequality still holds by reversing the order of $\boldsymbol{\pi}$ and $\boldsymbol{\rho}$; hence, the equilibriums for agent i at both saddle points are the same such that $\mathbb{E}_{\pi^j}\mathbb{E}_{\boldsymbol{\pi}^{-j}}[Q_t^j(s)] = \mathbb{E}_{\rho^j}\mathbb{E}_{\boldsymbol{\rho}^{-j}}[Q_t^j(s)]$.

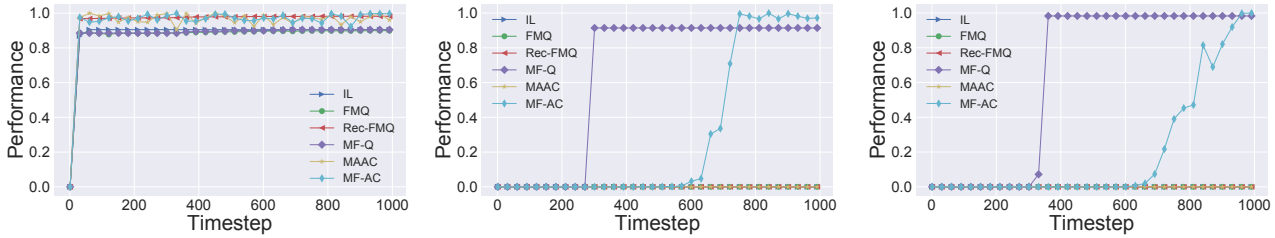
Given Proposition 1 that the policy based on the mean field Q-function forms a contraction mapping, and that all optimal/saddle points share the same Nash value in each stage game, with the homogeneity of agents, $\boldsymbol{v}^{\texttt{MF}}$ will asymptotically converges to $\boldsymbol{v}^{\texttt{Nash}}$, the third condition is thus satisfied.

For the fourth condition, we exploit the conclusion that is proved above that $\mathscr{H}^{\texttt{MF}}$ forms a contraction mapping, *i.e.* $\mathscr{H}^{\texttt{MF}}\boldsymbol{Q}_* = \boldsymbol{Q}_*$, and it follows that

$$\begin{aligned}\mathbf{var}[\boldsymbol{F}_t(s_t,\boldsymbol{a}_t)|\mathcal{F}_t] &= \mathbb{E}[(\boldsymbol{r}_t + \gamma \boldsymbol{v}_t^{\texttt{MF}}(s_{t+1}) - \boldsymbol{Q}_*(s_t,\boldsymbol{a}_t))^2]\\ &= \mathbb{E}[(\boldsymbol{r}_t + \gamma \boldsymbol{v}_t^{\texttt{MF}}(s_{t+1}) - \mathscr{H}^{\texttt{MF}}(\boldsymbol{Q}_*))^2]\\ &= \mathbf{var}[\boldsymbol{r}_t + \gamma \boldsymbol{v}_t^{\texttt{MF}}(s_{t+1})|\mathcal{F}_t]\\ &\leq K(1+\|\boldsymbol{\Delta}_t\|_W^2).\end{aligned} \tag{19}$$

In the last step of Eq. (19), we employ Assumption 1 that the reward $\boldsymbol{r}_t$ is always bounded by some constant. Finally, with all conditions met, it follows Lemma 2 that $\boldsymbol{\Delta}_t$ converges to zero *w.p.*1, *i.e.* $\boldsymbol{Q}_t$ converges to $\boldsymbol{Q}_*$ *w.p.*1. □

Apart from being convergent to the Nash Q-value, MF-Q is also *Rational* (Bowling & Veloso, 2001; 2002). We leave the corresponding discussion in Appendix D for details.



(a) $N = 100$ (b) $N = 500$ (c) $N = 1000$

Figure 3: Learning with N agents in the GS environment with $\mu = 400$ and $\sigma = 200$.

4. Related Work

We continue our discussion on related work from Introduction and make comparisons with existing techniques in a greater scope. Our work follows the same direction as Littman (1994); Hu & Wellman (2003); Bowling & Veloso (2002) on adapting a Stochastic Game (van der Wal et al., 1981) into the MARL formulation. Specifically, Littman (1994) addressed two-player zero-sum stochastic games by introducing a "minimax" operator in Q-learning, whereas Hu & Wellman (2003) extended it to the general-sum case by learning a Nash equilibrium in each stage game and considering a mixed strategy. Nash-Q learning is guaranteed to converge to Nash strategies under the (strong) assumption that there exists an equilibrium for every stage game. In the situation where agents can be identified as either "friends" or "foes" (Littman, 2001), one can simply solve it by alternating between fully cooperative and zero-sum learning. Considering the convergence speed, Littman & Stone (2005) and de Cote & Littman (2008) draw on the *folk theorem* and acquired a polynomial-time Nash equilibrium algorithm for repeated stochastic games, while Bowling & Veloso (2002) tried varying the learning rate to improve the convergence.

The recent treatment of MARL was using deep neural networks as the function approximator. In addressing the non-stationary issue in MARL, various solutions have been proposed including neural-based opponent modeling (He & Boyd-Graber, 2016), policy parameters sharing (Gupta et al., 2017), *etc*. Researchers have also adopted the paradigm of *centralized training with decentralized execution* for multi-agent policy-gradient learning: BICNET (Peng et al., 2017), COMA (Foerster et al., 2018) and MADDPG (Lowe et al., 2017a), which allows the centralized critic Q-function to be trained with the actions of other agents, while the actor needs only local observation to optimize agent's policy.

The above MARL approaches limit their studies mostly to tens of agents. As the number of agents grows larger, not only the input space of Q grows exponentially, but most critically, the accumulated noises by the exploratory actions of other agents make the Q-function learning no longer feasible. Our work addresses the issue by employing the mean field approximation (Stanley, 1971) over the joint action space. The parameters of the Q-function is independent of the number of agents as it transforms multiple agents interactions into two entities interactions (single agent *v.s.* the distribution of the neighboring agents). This would effectively alleviate the problem of the exploratory noise (Colby et al., 2015) caused by many other agents, and allow each agent to determine which actions are beneficial to itself.

Our work is also closely related to the recent development of mean field games (MFG) (Lasry & Lions, 2007; Huang et al., 2006; Weintraub et al., 2006). MFG studies population behaviors resulting from the aggregations of decisions taken from individuals. Mathematically, the dynamics are governed by a set of two stochastic differential equations that model the backward dynamics of individual's value function, and the forward dynamics of the aggregate distribution of agent population. Despite that the backward equation equivalently describes what Bellman equation indicates in the MDP, the primarily goal for MFG is rather for a model-based planning and to infer the movements of the individual density through time. The mean field approximation (Stanley, 1971) in also employed in physics, but our work is different in that we focus on a model-free solution of learning optimal actions when the dynamics of the system and the reward function are unknown. Very recently, Yang et al. (2017) built a connection between MFG and reinforcement learning. Their focus is, however, on the inverse RL in order to learn both the reward function and the forward dynamics of the MFG from the policy data, whereas our goal is to form a computable Q-learning algorithm under the framework of temporal difference learning.

5. Experiments

We analyze and evaluate our algorithms in three different scenarios, including two stage games: the Gaussian Squeeze and the Ising Model, and the *mixed cooperative-competitive* battle game.

5.1. Gaussian Squeeze

Environment. In the Gaussian Squeeze (GS) task (Holmes-Parker et al., 2014), N homogeneous agents determine their individual action a^j to jointly optimize the most appropriate summation $x = \sum_{j=1}^{N} a^j$. Each agent has 10 action choices – integers 0 to 9. The system objective is defined as $G(x) = xe^{\frac{-(x-\mu)^2}{\sigma^2}}$, where μ and σ are the pre-defined mean and variance of the system. In the scenario of traffic congestion, each agent is one traffic controller trying to send

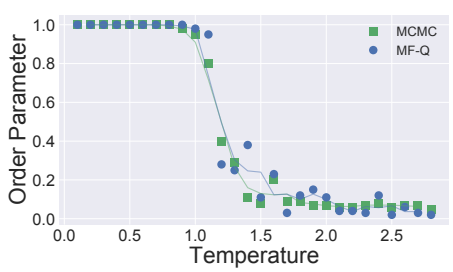
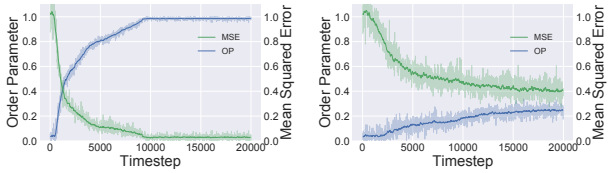


Figure 4: The *order parameter* at equilibrium *v.s.* temperature in the Ising model with 20×20 grid.



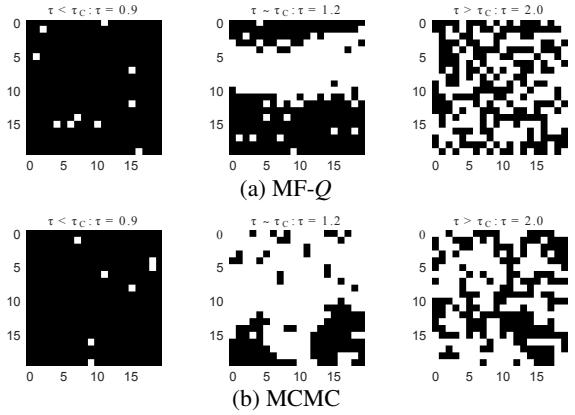
(a) $\tau = 0.8$ (b) $\tau = 1.2$

Figure 5: Training performance of MF-Q in the Ising model with 20×20 grid.

a^j vehicles into the main road. Controllers are expected to coordinate with each other to make the full use of the main route while avoiding congestions. The goal of each agent is to learn to allocate system resources efficiently, avoiding either over-use or under-use. The GS problem here sits ideally as an ablation study on the impact of multi-agent exploratory noises toward the learning (Colby et al., 2015).

Model Settings. We implement MF-Q and MF-AC following the framework of centralized training (shared critic) with decentralized execution (independent actor). We compare against 4 baseline models: (1) Independent Learner (IL), a traditional Q-Learning algorithm that does not consider the actions performed by other agents; (2) Frequency Maximum Q-value (FMQ) (Kapetanakis & Kudenko, 2002), a modified IL which increases the Q-values of actions that frequently produced good rewards in the past; (3) Recursive Frequency Maximum Q-value (Rec-FMQ) (Matignon et al., 2012), an improved version of FMQ that recursively computes the occurrence frequency to evaluate and then choose actions; (4) Multi-agent Actor-Critic (MAAC), a variant of MADDPG architecture for the discrete action space (see Eq. (4) in Lowe et al. (2017b)). All models use the multi-layer perception as the function approximator. The detailed settings of the implementation are in the Appendix C.1.

Results. Figure. 3 illustrates the results for the GS environment of $\mu = 400$ and $\sigma = 200$ with three different numbers of agents ($N = 100, 500, 1000$) that stand for 3 levels of congestions. In the smallest GS setting of Fig. 3a, all models show excellent performance. As the agent number increases, Figs. 3b and 3c show MF-Q and MF-AC's capabilities of learning the optimal allocation effectively after a few iterations, whereas all four baselines fail to learn at all. We believe this advantage is due to the awareness of other agents' actions under the mean field framework;



(a) MF-Q

(b) MCMC

Figure 6: The spins of the Ising model at equilibrium under different temperatures.

such mechanism keeps the interactions among agents manageable while reducing the noisy effect of the exploratory behaviors from the other agents. Between MF-Q and MF-AC, MF-Q converges faster. Both FMQ and Rec-FMQ fail to reach pleasant performance, it might be because agents are essentially unable to distinguish the rewards received for the same actions, and are thus unable to update their own Q-values *w.r.t.* the actual contributions. It is worth noting that MAAC is surprisingly inefficient in learning when the number of agents becomes large; it simply fails to handle the non-aggregated noises due to the agents' explorations.

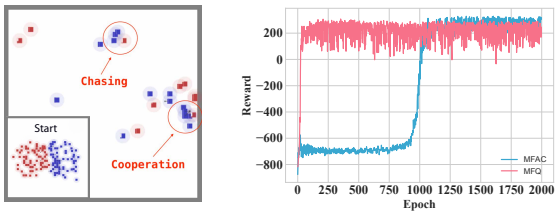
5.2. Model-free MARL for Ising Model

Environment. In statistical mechanics, the Ising model is a mathematical framework to describe ferromagnetism (Ising, 1925). It also has wide applications in sociophysics (Galam & Walliser, 2010). With the energy function explicitly defined, mean field approximation (Stanley, 1971) is a typical way to solve the Ising model for every spin j, *i.e.* $\langle a^j \rangle = \sum_a a^j P(a)$. See the Appendix C.2 for more details.

To fit into the MARL setting, we transform the Ising model into a stage game where the reward for each spin/agent is defined by $r^j = h^j a^j + \frac{\lambda}{2} \sum_{k \in \mathcal{N}(j)} a^j a^k$; here $\mathcal{N}(j)$ is the set of nearest neighbors of spin j, $h^j \in \mathbb{R}$ is the external field affecting the spin j, and $\lambda \in \mathbb{R}$ is an interaction coefficient that determines how much the spins are motivated to stay aligned. Unlike the typical setting in physics, here each spin does not know the energy function, but aims to understand the environment, and to maximize its reward by learning the optimal policy of choosing the spin state: up or down.

In addition to the reward, the *order parameter* (OP) (Stanley, 1971) is a traditional measure of purity for the Ising model. OP is defined as $\xi = \frac{|N_\uparrow - N_\downarrow|}{N}$, where $N_\uparrow$ represents the number of up spins, and $N_\downarrow$ for the down spins. The closer the OP is to 1, the more orderly the system is.

Model Settings. To validate the correctness of the MF-Q learning, we implement MCMC methods (Binder et al.,



(a) Battle game scene. (b) Learning curve.

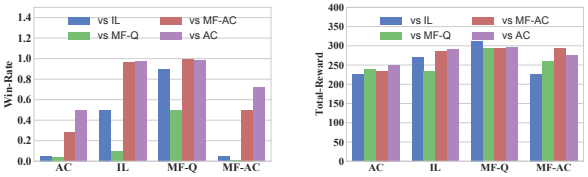
Figure 7: The battle game: 64 *v.s.* 64.

1993) to simulate the same Ising model and provide the ground truth for comparison. The full settings of MCMC and MF-Q for Ising model are provided in the Appendix C.2. One of the learning goals is to obtain the accurate approximation of $\langle a^j \rangle$. Notice that agents here do not know exactly the energy function, but rather use the temporal difference learning to approximate $\langle a^j \rangle$ during the learning procedure. Once this is accurately approximated, the Ising model as a whole should be able to converge to the same simulation result suggested by MCMC.

Correctness of MF-Q. Figure. 4 illustrates the relationship between the order parameter at equilibrium under different system temperatures. MF-Q converges nearly to the exact same plot as MCMC, this justifies the correctness of our algorithms. Critically, MF-Q finds a similar Curie temperature (the phase change point) as MCMC that is $\tau = 1.2$. As far as we know, this is the first work that manages to solve the Ising model via model-free reinforcement learning methods. Figure. 5 illustrates the mean squared error between the learned Q-value and the reward target. MF-Q is shown in Fig. 5a to be able to learn the target well under low temperature settings. When it comes to the Curie temperature, the environment enters into the phase change when the stochasticity dominates, resulting in a lower OP and higher MSE observed in Fig. 5b. We visualize the equilibrium in Fig. 6. The equilibrium points from MF-Q in fact match MCMC's results under three types of temperatures. The spins tend to stay aligned under a low temperature ($\tau = 0.9$). As the temperature rises ($\tau = 1.2$), some spins become volatile and patches start to form as spontaneous magnetization. This phenomenon is mostly observed around the Curie temperature. After passing the Curie temperature, the system becomes unstable and disordered due to the large thermal fluctuations, resulting in random spinning patterns.

5.3. Mixed Cooperative-Competitive Battle Game

Environment. The Battle game in the Open-source MAgent system (Zheng et al., 2018) is a *Mixed Cooperative-Competitive* scenario with two armies fighting against each other in a grid world, each empowered by a different RL algorithm. In the setting of Fig. 7a, each army consists of 64 homogeneous agents. The goal of each army is to get more rewards by collaborating with teammates to destroy all the opponents. Agent can takes actions to either move to or attack nearby grids. Ideally, the agents army should learn



(a) Average wining rate. (b) Average total reward.

Figure 8: Performance comparisons in the battle game.

skills such as chasing to hunt after training. We adopt the default reward setting: -0.005 for every move, 0.2 for attacking an enemy, 5 for killing an enemy, -0.1 for attacking an empty grid, and -0.1 for being attacked or killed.

Model Settings. Our MF-Q and MF-AC are compared against the baselines that are proved successful on the MAgent platform. We focus on the battles between mean field methods (MF-Q, MF-AC) and their non-mean field counterparts, independent Q-learning (IL) and advantageous actor critic (AC). We exclude MADDPG/MAAC as baselines, as the framework of centralized critic cannot deal with the varying number of agents for the battle (simply because agents could die in the battle). Also, as we demonstrated in the previous experiment of Fig. 3, MAAC tends to scale poorly and fail when the agent number is in hundreds.

Results and Discussion. We train all four models by 2000 rounds *self-plays*, and then use them for comparative battles. During the training, agents can quickly pick up the skills of chasing and cooperation to kill in Fig. 7a. The Fig. 8 shows the result of winning rate and the total reward over 2000 rounds cross-comparative experiments. It is evident that on all the metrics mean field methods, MF-Q largely outperforms the corresponding baselines, *i.e.* IL and AC respectively, which shows the effectiveness of the mean field MARL algorithms. Interestingly, IL performs far better than AC and MF-AC (2nd block from the left in Fig. 8a), although it is worse than the mean field counterpart MF-Q. This might imply the effectiveness of off-policy learning with shuffled buffer replay in many-agent RL towards a more stable learning process. Also, the Q-learning family tends to introduce a positive bias (Hasselt, 2010) by using the maximum action value as an approximation for the maximum expected action value, and such overestimation can be beneficial for each single agent to find the best response to others even though the environment itself is still changing. On the other hand, On-policy methods need to comply with the GLIE assumption (Assumption 2 in Sec 3.3) so as to converge properly to the optimal value (Singh et al., 2000), which is in the end a greedy policy as off-policy methods. Figure. 7b further shows the self-play learning curves of MF-AC and MF-Q. MF-Q presents a faster convergence speed than MF-AC, which is consistent with the findings in the Gaussian Squeeze task (see Fig. 3b & 3c). Apart from 64, we further test the scenarios when the agent size is 8, 144, 256, the comparative results keep the same relativity as Fig. 8; we omit the presentations for clarity.

6. Conclusions

In this paper, we developed mean field reinforcement learning methods to model the dynamics of interactions in the multi-agent systems. MF-Q iteratively learns each agent's best response to the mean effect from its neighbors; this effectively transform the many-body problem into a two-body problem. Theoretical analysis on the convergence of the MF-Q algorithm to Nash Q-value was provided. Three types of tasks have justified the effectiveness of our approaches. In particular, we report the first result to solve the Ising model using model-free reinforcement learning methods.

Acknowledgement

We sincerely thank Ms. Yi Qu for her generous help on the graphic design.

References

Bertsekas, D. P. Weighted sup-norm contractions in dynamic programming: A review and some new applications. *Dept. Elect. Eng. Comput. Sci., Massachusetts Inst. Technol., Cambridge, MA, USA, Tech. Rep. LIDS-P-2884*, 2012.

Binder, K., Heermann, D., Roelofs, L., Mallinckrodt, A. J., and McKay, S. Monte carlo simulation in statistical physics. *Computers in Physics*, 7(2):156–157, 1993.

Blume, L. E. The statistical mechanics of strategic interaction. *Games and economic behavior*, 5(3):387–424, 1993.

Bowling, M. and Veloso, M. Rational and convergent learning in stochastic games. In *International joint conference on artificial intelligence*, volume 17, pp. 1021–1026. Lawrence Erlbaum Associates Ltd, 2001.

Bowling, M. and Veloso, M. Multiagent learning using a variable learning rate. *Artificial Intelligence*, 136(2): 215–250, 2002.

Busoniu, L., Babuska, R., and De Schutter, B. A comprehensive survey of multiagent reinforcement learning. *IEEE Trans. Systems, Man, and Cybernetics, Part C*, 38 (2):156–172, 2008.

Cao, Z., Qin, T., Liu, T.-Y., Tsai, M.-F., and Li, H. Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pp. 129–136. ACM, 2007.

Colby, M. K., Kharaghani, S., HolmesParker, C., and Tumer, K. Counterfactual exploration for improving multiagent learning. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pp. 171–179. International Foundation for Autonomous Agents and Multiagent Systems, 2015.

de Cote, E. M. and Littman, M. L. A polynomial-time nash equilibrium algorithm for repeated stochastic games. In McAllester, D. A. and Myllymäki, P. (eds.), *UAI 2008*, pp. 419–426. AUAI Press, 2008. ISBN 0-9749039-4-9.

Fink, A. M. et al. Equilibrium in a stochastic n-person game. *Journal of science of the hiroshima university, series ai (mathematics)*, 28(1):89–93, 1964.

Foerster, J. N., Chen, R. Y., Al-Shedivat, M., Whiteson, S., Abbeel, P., and Mordatch, I. Learning with opponent-learning awareness. *CoRR*, abs/1709.04326, 2017.

Foerster, J. N., Farquhar, G., Afouras, T., Nardelli, N., and Whiteson, S. Counterfactual multi-agent policy gradients. In McIlraith & Weinberger (2018).

Galam, S. and Walliser, B. Ising model versus normal form game. *Physica A: Statistical Mechanics and its Applications*, 389(3):481–489, 2010.

Gupta, J. K., Egorov, M., and Kochenderfer, M. Cooperative multi-agent control using deep reinforcement learning. In *AAMAS*, pp. 66–83. Springer, 2017.

Hasselt, H. V. Double q-learning. In *NIPS*, pp. 2613–2621, 2010.

He, H. and Boyd-Graber, J. L. Opponent modeling in deep reinforcement learning. In Balcan, M. and Weinberger, K. Q. (eds.), *ICML*, volume 48, pp. 1804–1813. JMLR.org, 2016.

HolmesParker, C., Taylor, M., Zhan, Y., and Tumer, K. Exploiting structure and agent-centric rewards to promote coordination in large multiagent systems. In *Adaptive and Learning Agents Workshop*, 2014.

Hu, J. and Wellman, M. P. Nash q-learning for general-sum stochastic games. *Journal of machine learning research*, 4(Nov):1039–1069, 2003.

Huang, M., Malhamé, R. P., Caines, P. E., et al. Large population stochastic dynamic games: closed-loop mckean-vlasov systems and the nash certainty equivalence principle. *Communications in Information & Systems*, 6(3): 221–252, 2006.

Ising, E. Beitrag zur theorie des ferromagnetismus. *Zeitschrift für Physik*, 31(1):253–258, 1925.

Jaakkola, T., Jordan, M. I., and Singh, S. P. Convergence of stochastic iterative dynamic programming algorithms. In *NIPS*, pp. 703–710, 1994.

Jeong, S. H., Kang, A. R., and Kim, H. K. Analysis of game bot's behavioral characteristics in social interaction networks of mmorpg. In *ACM SIGCOMM Computer Communication Review*, volume 45, pp. 99–100. ACM, 2015.

Kapetanakis, S. and Kudenko, D. Reinforcement learning of coordination in cooperative multi-agent systems. In *NCAI*, pp. 326–331, Menlo Park, CA, USA, 2002. ISBN 0-262-51129-0.

Konda, V. R. and Tsitsiklis, J. N. Actor-critic algorithms. In Solla, S. A., Leen, T. K., and Müller, K. (eds.), *Advances in Neural Information Processing Systems 12*, pp. 1008–1014. MIT Press, 2000.

Kreyszig, E. *Introductory functional analysis with applications*, volume 1. wiley New York, 1978.

Lasry, J.-M. and Lions, P.-L. Mean field games. *Japanese journal of mathematics*, 2(1):229–260, 2007.

Lemke, C. E. and Howson, Jr, J. T. Equilibrium points of bimatrix games. *Journal of the Society for Industrial and Applied Mathematics*, 12(2):413–423, 1964.

Littman, M. L. Markov games as a framework for multi-agent reinforcement learning. In *ICML*, volume 157, pp. 157–163, 1994.

Littman, M. L. Friend-or-foe q-learning in general-sum games. In *ICML*, volume 1, pp. 322–328, 2001.

Littman, M. L. and Stone, P. A polynomial-time nash equilibrium algorithm for repeated games. *Decision Support Systems*, 39(1):55–66, 2005.

Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, O. P., and Mordatch, I. Multi-agent actor-critic for mixed cooperative-competitive environments. In *NIPS*, pp. 6382–6393, 2017a.

Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., and Mordatch, I. Multi-agent actor-critic for mixed cooperative-competitive environments. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R. (eds.), *NIPS*, pp. 6382–6393, 2017b.

Matignon, L., Laurent, G. J., and Le Fort-Piat, N. Independent reinforcement learners in cooperative markov games: a survey regarding coordination problems. *The Knowledge Engineering Review*, 27(1):1–31, 2012.

McIlraith, S. A. and Weinberger, K. Q. (eds.). *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, Louisiana, USA, February 2-7, 2018*, 2018. AAAI Press.

Melo, F. S., Meyn, S. P., and Ribeiro, M. I. An analysis of reinforcement learning with function approximation. In *Proceedings of the 25th international conference on Machine learning*, pp. 664–671. ACM, 2008.

Panait, L. and Luke, S. Cooperative multi-agent learning: The state of the art. *AAMAS*, 11(3):387–434, 2005.

Peng, P., Yuan, Q., Wen, Y., Yang, Y., Tang, Z., Long, H., and Wang, J. Multiagent bidirectionally-coordinated nets for learning to play starcraft combat games. *CoRR*, abs/1703.10069, 2017.

Rendle, S. Factorization machines with libfm. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 3 (3):57, 2012.

Shapley, L. S. Stochastic games. *Proceedings of the national academy of sciences*, 39(10):1095–1100, 1953.

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., and Riedmiller, M. Deterministic policy gradient algorithms. In *ICML*, pp. 387–395, 2014.

Singh, S., Jaakkola, T., Littman, M. L., and Szepesvári, C. Convergence results for single-step on-policy reinforcement-learning algorithms. *Machine learning*, 38(3):287–308, 2000.

Stanley, H. E. Phase transitions and critical phenomena. *Clarendon, Oxford*, 9, 1971.

Szepesvári, C. and Littman, M. L. A unified analysis of value-function-based reinforcement-learning algorithms. *Neural computation*, 11(8):2017–2060, 1999.

Tan, M. Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proceedings of the tenth international conference on machine learning*, pp. 330–337, 1993.

Troy, C. A. Envisioning stock trading where the brokers are bots. *New York Times*, 16, 1997.

van der Wal, J., van der Wal, J., van der Wal, J., Mathématicien, P.-B., van der Wal, J., and Mathematician, N. *Stochastic Dynamic Programming: successive approximations and nearly optimal strategies for Markov decision processes and Markov games*. Mathematisch centrum, 1981.

Wang, J., Zhang, W., Yuan, S., et al. Display advertising with real-time bidding (rtb) and behavioural targeting. *Foundations and Trends® in Information Retrieval*, 11 (4-5):297–435, 2017.

Watkins, C. J. and Dayan, P. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

Weintraub, G. Y., Benkard, L., and Van Roy, B. Oblivious equilibrium: A mean field approximation for large-scale dynamic games. In *Advances in neural information processing systems*, pp. 1489–1496, 2006.

Yang, J., Ye, X., Trivedi, R., Xu, H., and Zha, H. Deep mean field games for learning optimal behavior policy of large populations. *CoRR*, abs/1711.03156, 2017.

Zheng, L., Yang, J., Cai, H., Zhou, M., Zhang, W., Wang, J., and Yu, Y. Magent: A many-agent reinforcement learning platform for artificial collective intelligence. In McIlraith & Weinberger (2018).

A. Detailed mean field reinforcement learning algorithms

We published the code at `https://github.com/mlii/mfrl`.

Algorithm 1 Mean Field Q-learning (MF-Q)

Initialise Q_{ϕ^j}, $Q_{\phi^j_-}$, and $\bar{a}^j$ for all $j \in \{1,\dots,N\}$
while training not finished **do**
  for $m = 1, ..., M$ **do**
    For each agent j, sample action a^j from Q_{ϕ^j} by Eq. (12), with the current mean action $\bar{a}^j$ and the exploration rate β
    For each agent j, compute the new mean action $\bar{a}^j$ by Eq. (11)
  Take the joint action $\boldsymbol{a} = [a^1,\dots,a^N]$ and observe the reward $\boldsymbol{r} = [r^1,\dots,r^N]$ and the next state s'
  Store $\langle s, \boldsymbol{a}, \boldsymbol{r}, s', \bar{\boldsymbol{a}}\rangle$ in replay buffer $\mathcal{D}$, where $\bar{\boldsymbol{a}} = [\bar{a}^1,\dots,\bar{a}^N]$
  for $j = 1$ to N **do**
    Sample a minibatch of K experiences $\langle s, \boldsymbol{a}, \boldsymbol{r}, s', \bar{\boldsymbol{a}}\rangle$ from $\mathcal{D}$
    Sample action a^j_- from $Q_{\phi^j_-}$ with $\bar{a}^j_- \leftarrow \bar{a}^j$
    Set $y^j = r^j + \gamma\, v^{\mathtt{MF}}_{\phi^j_-}(s')$ by Eq. (10)
    Update the Q-network by minimizing the loss $\mathcal{L}(\phi^j) = \frac{1}{K}\sum \left(y^j - Q_{\phi^j}(s^j, a^j, \bar{a}^j)\right)^2$
  Update the parameters of the target network for each agent j with learning rate τ:

$$\phi^j_- \leftarrow \tau\phi^j + (1-\tau)\phi^j_-$$

Algorithm 2 Mean Field Actor-Critic (MF-AC)

Initialize Q_{ϕ^j}, $Q_{\phi^j_-}$, π_{θ^j}, $\pi_{\theta^j_-}$, and $\bar{a}^j$ for all $j \in \{1,\dots,N\}$
while training not finished **do**
  For each agent j, sample action $a^j = \pi_{\theta^j}(s)$; compute the new mean action $\bar{\boldsymbol{a}} = [\bar{a}^1,\dots,\bar{a}^N]$
  Take the joint action $\boldsymbol{a} = [a^1,\dots,a^N]$ and observe the reward $\boldsymbol{r} = [r^1,\dots,r^N]$ and the next state s'
  Store $\langle s, \boldsymbol{a}, \boldsymbol{r}, s', \bar{\boldsymbol{a}}\rangle$ in replay buffer $\mathcal{D}$
  for $j = 1$ to N **do**
    Sample a minibatch of K experiences $\langle s, \boldsymbol{a}, \boldsymbol{r}, s', \bar{\boldsymbol{a}}\rangle$ from $\mathcal{D}$
    Set $y^j = r^j + \gamma\, v^{\mathtt{MF}}_{\phi^j_-}(s')$ by Eq. (10)
    Update the critic by minimizing the loss $\mathcal{L}(\phi^j) = \frac{1}{K}\sum \left(y^j - Q_{\phi^j}(s, a^j, \bar{a}^j)\right)^2$
    Update the actor using the sampled policy gradient:

$$\nabla_{\theta^j} \mathcal{J}(\theta^j) \approx \frac{1}{K}\sum \nabla_{\theta^j} \log \pi_{\theta^j}(s') Q_{\phi^j_-}(s', a^j_-, \bar{a}^j_-)\Big|_{a^j_- = \pi_{\theta^j_-}(s')}$$

  Update the parameters of the target networks for each agent j with learning rates τ_ϕ and τ_θ:

$$\phi^j_- \leftarrow \tau_\phi\phi^j + (1-\tau_\phi)\phi^j_-$$
$$\theta^j_- \leftarrow \tau_\theta\theta^j + (1-\tau_\theta)\theta^j_-$$

B. Proof of the bound for the remainder term in Eq. 7

Recall Eq. (8) that we approximate the action a^k taken by the neighboring agent k with the mean action $\bar{a}$ calculated from the neighborhood $\mathcal{N}(j)$. The state s and the action a^j of the central agent j can be considered as fixed parameters; the indices j,k of agents are essentially irrelevant to the derivation. With those omitted for simplicity, We rewrite the expression of the pairwise Q-function as $Q(a) \triangleq Q^j(s,a^j,a^k)$.

Suppose that Q is M-smooth, where its gradient ∇Q is Lipschitz-continuous with constant M such that for all $a,\bar{a}$

$$\|\nabla Q(a) - \nabla Q(\bar{a})\|_2 \leq M\|a-\bar{a}\|_2, \tag{20}$$

where $\|\cdot\|_2$ indicates the ℓ_2-norm.

With the Lagrange's mean value theorem, we have

$$\nabla Q(a) - \nabla Q(\bar{a}) = \nabla Q(\bar{a}+1\cdot(a-\bar{a})) - \nabla Q(\bar{a}) = \nabla^2 Q(\bar{a}+\epsilon\cdot(a-\bar{a}))\cdot(a-\bar{a}), \quad \text{where } \epsilon\in[0,1]. \tag{21}$$

Take the ℓ_2-norm on the both sides of the above equation, it follows from the smoothness condition that

$$\|\nabla Q(a) - \nabla Q(\bar{a})\|_2 = \|\nabla^2 Q(\bar{a}+\tau\cdot(a-\bar{a}))\cdot(a-\bar{a})\|_2 \leq M\|a-\bar{a}\|_2. \tag{22}$$

Define $\delta a \triangleq a-\bar{a}$ and the normalized vector $\delta\hat{a} \triangleq {}^{a-\bar{a}}/_{\|a-\bar{a}\|_2}$ with $\|\delta\hat{a}\|_2 = 1$, it follows from the above inequality

$$\|\nabla^2 Q(a+\tau\cdot\delta a)\cdot\delta\hat{a}\|_2 \leq M. \tag{23}$$

By arbitrary choice of (the unnormalized vector) δa such that the magnitude $\|\delta a\|_2 \to 0$, it follows from above that

$$\|\nabla^2 Q(a)\cdot\delta\hat{a}\|_2 \leq M. \tag{24}$$

By aligning (the normalized vector) $\delta\hat{a}$ in the direction of the eigenvectors of the Hessian matrix $\nabla^2 Q$, we can obtain for any eigenvalue λ of $\nabla^2 Q$ that

$$\|\nabla^2 Q(a)\cdot\delta\hat{a}\|_2 = \|\lambda\cdot\delta\hat{a}\|_2 = |\lambda|\cdot\|\delta\hat{a}\|_2 \leq M, \tag{25}$$

which indicates that all eigenvalues of $\nabla^2 Q$ can be bounded in the symmetric interval $[-M,M]$.

As the Hessian matrix $\nabla^2 Q$ is real symmetric and hence diagonalizable, there exist an orthogonal matrix U such that $U^\top[\nabla^2 Q]U = \Lambda \triangleq \mathrm{diag}[\lambda_1,\ldots,\lambda_D]$. It then follows that

$$\delta a\cdot\nabla^2 Q\cdot\delta a = [U\delta a]^\top\Lambda[U\delta a] = \sum_{i=1}^{D}\lambda_i[U\delta a]_i^2, \quad \text{with} \quad -M\|U\delta a\|_2 \leq \sum_{i=1}^{D}\lambda_i[U\delta a]_i^2 \leq M\|U\delta a\|_2 \tag{26}$$

Recall the definition $\delta a = a-\bar{a}$ in Eq. (6), where a is the one-hot encoding for D actions, and $\bar{a}$ is a D-dimensional multinomial distribution. It can be shown that

$$\|U\delta a\|_2 = \|\delta a\|_2 = (a-\bar{a})^\top(a-\bar{a}) = a^\top a + \bar{a}^\top\bar{a} - \bar{a}^\top a - a^\top\bar{a} = 2(1-\bar{a}_i) \leq 2, \tag{27}$$

where i represents the specific action a has represented such that $a_{i'} = 0$ for $i'\neq i$.

With all elements assembled, we have proved that each single remainder term $R^j_{s,a^j}(a^k)$ in Eq. (8) is bounded in $[-2M,2M]$.

C. Experiment details

C.1. Gaussian Squeeze

IL, FMQ, Rec-FMQ and MF-Q all use a three-layer MLP to approximate Q-value. All agents share the same Q-network for each experiment. The shared Q-network takes an agent embedding as input and computes Q-value for each candidate action. For MF-Q, we also feed in the action approximation $\bar{a}$. We use the Adam optimizer with a learning rate of 0.00001 and ϵ-greedy exploration unless otherwise specified. For FMQ, we set the exponential decay rate $s = 0.006$, start temperature *max_temp=1000* and FMQ heuristic $c = 5$. For Rec-FMQ, we set the frequency learning rate $\alpha_f = 0.01$.

MAAC and MF-AC use the Adam optimizer with a learning rate of 0.001 and 0.0001 for Critics and Actors respectively, and $\tau = 0.01$ for updating the target networks. We share the Critic among all agents in each experiment and feed in an agent embedding as extra input. Actors are kept separate. The discounted factor γ is set to be 0.95 and the mini-batch size is set to be 200. The size of the replay buffer is 10^6 and we update the network parameters after every 500 samples added to the replay buffer.

For all models, we use the performance of the joint-policy learned up to that point if learning and exploration were turned off (*i.e.*, take the greedy action w.r.t. the learned policy) to compare our method with the above baseline models.

C.2. Ising Model

An Ising model is defined as a stateless system with N homogeneous sites on a finite square lattice. Each site determines their individual spin a^j to interact with each other and aims to minimize the system energy for a more stable environment. The system energy is defined as

$$E(a,h) = -\sum_j (h^j a^j + \frac{\lambda}{2} \sum_{k \in \mathcal{N}(j)} a^j a^k) \tag{28}$$

where $\mathcal{N}(j)$ is the set of nearest neighbors of site j, $h^j \in \mathbb{R}$ is the external field affecting site j, and $\lambda \in \mathbb{R}$ is an interaction term determines how much the sites tend to align in the same direction. The system is said to reach an equilibrium point when the system energy is minimized, with the probability

$$P(a) = \frac{\exp\left(-E(a,h)/\tau\right)}{\sum_a \exp(-E(a,h)/\tau)}, \tag{29}$$

where τ is the system temperature. When the temperature rises beyond a certain point (the Curie temperature), the system can no longer keep a stable form and a phase transition happens. As the ground-truth is known, we would be able to evaluate the correctness of the Q-function learning when there is a large body of agents interacted.

The mean field theory provides an approximate solution to $\langle a^j \rangle = \sum_a a^j P(a)$ through a set of self-consistent mean field equations

$$\langle a^j \rangle = \frac{\exp\left(-[h^j a^j + \lambda \sum_{k \in \mathcal{N}(j)} \langle a^k \rangle]/\tau\right)}{1 + \exp\left(-[h^j a^j + \lambda \sum_{k \in \mathcal{N}(j)} \langle a^k \rangle]/\tau\right)}. \tag{30}$$

which can be solved iteratively by

$$\langle a^j \rangle^{(t+1)} = \frac{\exp\left(-[h^j a^j + \lambda \sum_{k \in \mathcal{N}(j)} \langle a^k \rangle^{(t)}]/\tau\right)}{1 + \exp\left(-[h^j a^j + \lambda \sum_{k \in \mathcal{N}(j)} \langle a^k \rangle^{(t)}]/\tau\right)}, \tag{31}$$

where t represents the number of iterations.

To learn an optimal joint policy $\boldsymbol{\pi}^*$ for Ising model, we use the stateless Q-learning with mean field approximation (MF-Q), defined as

$$Q^j(a^j, \bar{a}^j) \leftarrow Q^j(a^j, \bar{a}^j) + \alpha[r^j - Q^j(a^j, \bar{a}^j)], \tag{32}$$

Algorithm 3 MCMC in Ising Model

initialize spin state $\boldsymbol{a} \in \{-1, 1\}^N$ for N sites
while training not finished **do**
  randomly choose site $j \in \mathcal{N}(j)$
  flip the spin state for site j: $a^j_- \leftarrow -a^j$
  compute neighbor energy $E(a, h) = -\sum_j (h^j a^j + \frac{\lambda}{2} \sum_{k \in \mathcal{N}(j)} a^j a^k)$ for a^j and a^j_-
  randomly choose $\epsilon \sim U(0, 1)$
  if $\exp((E(a^j, h) - E(a^j_-, h))/\tau) > \epsilon$ **then**
    $a^j \leftarrow a^j_-$

where the mean $\bar{a}^j$ is given as the mean $\langle a^j \rangle$ from the last time step, and the individual reward is

$$r^j = h^j a^j + \frac{\lambda}{2} \sum_{k \in \mathcal{N}(j)} a^j a^k. \tag{33}$$

To balance the trade-off between exploration and exploitation under low temperature settings, we use a policy with Boltzmann exploration and a decayed exploring temperature. The temperature for Boltzmann exploration of MF-Q is multiplied by a decay factor exponentially through out the training process.

Without lost of generality, we assume $\lambda > 0$, thus neighboring sites with the same action result in lower energy (observe higher reward) and are more stable. Each site should also align with the sign of external field h^j to reduce the system energy. For simplification, we eliminate the effect of external fields and assume the model to be discrete, *i.e.*, $\forall j \in N, h^j = 0, a^j \in \{-1, 1\}$.

We simulate the Ising model using Metropolis Monte Carlo methods (MCMC). After initialization, we randomly change a site's spin state and calculate the energy change, select a random number between 0 and 1, and accept the state change only if the number is less than $e^{\frac{(E^j - E^j_-)}{\tau}}$. This is called the Metropolis technique, which saves computation time by selecting the more probable spin states.

C.3. Battle Game

IL and MF-Q have almost the same hyper-parameters settings. The learning rate is $\alpha = 10^{-4}$, and with a dynamic exploration rate linearly decays from $\gamma = 1.0$ to $\gamma = 0.05$ during the 2000 rounds training. The discounted factor γ is set to be 0.95 and the mini-batch size is 128. The size of replay buffer is 5×10^5.

AC and MF-AC also have almost the same hyper-parameters settings. The learning rate is $\alpha = 10^{-4}$, the temperature of soft-max layer in $actor$ is $\tau = 0.1$. And the coefficient of entropy in the total loss is 0.08, the coefficient of value in the total loss is 0.1.

D. Further details towards the theoretical guarantee of MF-Q

Proposition 1. *Let the metric space be $\mathbb{R}^N$ and the metric be $d(\boldsymbol{a},\boldsymbol{b}) = \sum_j |a^j - b^j|$, for $\boldsymbol{a} = [a^j]_1^N, \boldsymbol{b} = [b^j]_1^N$. If the Q-function is K-Lipschitz continuous* w.r.t. *a^j, then the operator $\mathcal{B}(a^j) \triangleq \pi^j(a^j|s,\bar{a}^j)$ in Eq.* (12) *forms a contraction mapping under sufficiently low temperature β.*

Proof. Following the contraction mapping theorem (Kreyszig, 1978), in order to be a contraction, the operator has to satisfy:

$$d(\mathcal{B}(\boldsymbol{a}),\mathcal{B}(\boldsymbol{b})) \leq \alpha d(\boldsymbol{a},\boldsymbol{b}), \quad \forall \boldsymbol{a},\boldsymbol{b}$$

where $0 \leq \alpha < 1$ and $\mathcal{B}(\boldsymbol{a}) \triangleq [\mathcal{B}(a^1),\ldots,\mathcal{B}(a^N)]$.

Here we start from binomial case and then adapt to the multinomial case in general. We first rewrite $\mathcal{B}(a^j)$ as

$$\begin{aligned}\mathcal{B}(a^j) = \pi^j(a^j|s,\bar{a}^j) =& \frac{\exp\big(-\beta Q_t^j(s,a^j,\bar{a}^j)\big)}{\exp\big(-\beta Q_t^j(s,a^j,\bar{a}^j)\big) + \exp\big(-\beta Q_t^j(s,\neg a^j,\bar{a}^j)\big)} \\ =& \frac{1}{1+\exp\left(-\beta\cdot \Delta Q(s,a^j,\bar{a})\right)}\,, \end{aligned} \tag{34}$$

where $\Delta Q(s,a^j,\bar{a}) = Q(s,a^{\neg j},\bar{a}) - Q(s,a^j,\bar{a})$.

Then we have

$$\begin{aligned}|\mathcal{B}(a^j) - \mathcal{B}(b^j)| =& \left|\frac{1}{1+e^{-\beta\cdot\Delta Q(s,a^j,\bar{a})}} - \frac{1}{1+e^{-\beta\cdot\Delta Q(s,b^j,\bar{a})}}\right| \\ =& \left|\frac{\beta e^{-\beta\Delta Q_0}}{(1+e^{-\beta\Delta Q_0})^2}\right| \left|\Delta Q(s,a^j,\bar{a}) - \Delta Q(s,b^j,\bar{a})\right| \\ \leq& \frac{1}{4T}\cdot\left|Q(s,a^{\neg j},\bar{a}) - Q(s,b^{\neg j},\bar{a}) + Q(s,b^j,\bar{a}) - Q(s,a^j,\bar{a})\right| \\ \leq& \frac{1}{4T}\cdot\left(K\cdot\left|1-a^j-(1-b^j)\right| + K\cdot\left|a^j-b^j\right|\right) \\ \leq& \frac{1}{4T}\cdot 2K\cdot\sum_j\left|a^j-b^j\right|\,. \end{aligned} \tag{35}$$

In the second equation, we apply the mean value theorem in calculus: $\exists x_0 \in [x_1,x_2]$, *s.t.*, $f(x_1)-f(x_2) = f'(x_0)(x_1-x_2)$. In the third equation we use the maximum value for $e^{-\beta\Delta Q_0}/(1+e^{-\beta\Delta Q_0})^2 = 1/4$ when $Q_0 = 0$. In the last equation we apply the Lipschitz constraint in the assumption where constant $K \geq 0$. Finally, we have:

$$\begin{aligned} d(\mathcal{B}(\boldsymbol{a}),\mathcal{B}(\boldsymbol{b})) &\leq \frac{1}{4T}\cdot 2K\cdot\sum_j\left|a^j-b^j\right| \\ &= \frac{K}{2T}d(\boldsymbol{a},\boldsymbol{b}) \end{aligned} \tag{36}$$

In order for the contraction to hold, $T > \frac{K}{2}$. In other words, when the action space is binary for each agent, and the temperature is sufficiently large, the mean field procedure converges.

This proposition can be easily extended to multinomial case by replacing binary variable a^j by a multi-dimensional binary indicator vector $\boldsymbol{a}^j$, on each dimension, the rest of the derivations would remain essentially the same. □

D.1. Discussion on Rationality

In line with (Bowling & Veloso, 2001; 2002), we argue that to better evaluate a multi-agent learning algorithm, on top of the convergence guarantee, discussion on property of Rationality is also needed.

Property 1. *(also see (Bowling & Veloso, 2001; 2002)) In an N-agent stochastic game defined in this paper, given all agents converge to stationary policies, if the learning algorithm converges to a policy that is a **best response** to the other agents' policies, then the algorithm is Rationale.*

Our mean field Q-learning is rational in that Eq. (5) converts many agents interactions into two-body interactions between a single agent and the distribution of other agents actions. When all agents follow stationary policies, their policy distribution would be stationary too. As such the two-body stochastic game becomes an MDP, and the agent would choose a policy (based on Assumption 2) which is the best response to the distribution of other stationary policies. As agents are symmetric in our case, they all show the best response to the distributions, and are therefore rational.

E. Proof of Mean Field Reinforcement Learning with Function Approximation

Previous convergence results in Theorem.1 has shown that the Mean Field Q-learning algorithm will converge when the Q function is in a tabular case. We now move onto the proof that the MF-Q algorithm will converge when the Q function is represented by some functional approximations.

An N-agent (or, N-player) stochastic game Γ is formalized by the tuple $\Gamma \triangleq \left(\mathcal{S}, \mathcal{A}^1, \ldots, \mathcal{A}^N, r^1, \ldots, r^N, p, \gamma\right)$. The state-space $\mathcal{S}$ is finite. Let $(\mathcal{S}, p_{\boldsymbol{\pi}})$ be the Markov chain induced by the joint policy $\boldsymbol{\pi}$, and we assume it to be uniformly ergodic.

Let $\mathbb{Q} = \{\boldsymbol{Q}_\theta\}$ be a family of real-valued functions defined on $\mathcal{S} \times \mathcal{A} \times \bar{\mathcal{A}}$, where $\bar{\mathcal{A}}$ is the action space for the mean actions computed from the neighbors. Assuming that the function class is linearly parameterized, for each agent j, Q can be expressed as the linear span of a fixed set of P linearly independent functions $\omega_p^j : \mathcal{S} \times \mathcal{A} \times \bar{\mathcal{A}} \to \mathbb{R}$. Given the parameter vector $\phi^j \in \mathbb{R}^P$, for each agent, the function Q_{ϕ^j} is thus defined as

$$Q_{\phi^j}(s, a^j, \bar{a}^j) = \sum_{p=1}^{P} \omega_p^j(s, a^j, \bar{a}^j)\phi^j(p) = \omega^j(s, a^j, \bar{a}^j)^\top \phi^j \tag{37}$$

In the functional approximation setting, we can apply the update rules:

$$\begin{aligned} \phi_{t+1}^j &= \phi_t^j + \alpha_t \Delta_t \nabla_{\phi^j} Q_{\phi^j}(s, a^j, \bar{a}^j) \\ &= \phi_t^j + \alpha_t \Delta_t \omega^j(s, a_t^j, \bar{a}_t^j) \end{aligned} \tag{38}$$

In the above, Δ_t is the temporal difference at time t.

$$\Delta_t = r^j + \gamma\, v_{\phi^j}^{\mathtt{MF}}(s') - Q_{\phi^j}(s, a^j, \bar{a}^j) \tag{39}$$

$$= r^j + \gamma\, \mathbb{E}_{\boldsymbol{a} \sim \boldsymbol{\pi}^{\mathtt{MF}}}\left[Q_{\phi^j}(s', a^j, \bar{a}^j)\right] - Q_{\phi^j}(s, a^j, \bar{a}^j). \tag{40}$$

And the goal is to derive the parameter vector $\boldsymbol{\phi} = \{\phi^j\}$ such that $\boldsymbol{\omega}^\top \boldsymbol{\phi}$ approximates the (local) Nash Q-values. At each time step, the learning policy $\boldsymbol{\pi}_{\phi_t}$ is the Botlzmann policy with respect to $\boldsymbol{\omega}^\top \boldsymbol{\phi}$. Give the **Proposition 1**, we know that the policy $\boldsymbol{\pi}_{\phi_t}$ is $\frac{K}{2T}$ Lipschitz continuous with respect to ϕ_t.

Similar to the framework used in the convergence proof of Q-learning with function approximation (Melo et al., 2008), we establish convergence of Eq. (38) by adopting an ordinary differentiable equation (ODE) with a globally asymptotically stable equilibrium point where the trajectories closely follow.

Theorem 2. *Given the MDP Γ, $\boldsymbol{\pi}_{\phi_t}$, $\{\boldsymbol{\omega}_p, p = 1, ..., P\}$, and the learning policy $\boldsymbol{\pi}_{\phi_t}$ that is $\frac{K}{2T}$ Lipschitz continuous with respect to ϕ_t, if the Assumptions 1, 2 & 3, and Lemma 2's first and second conditions are met, then there exists C_0 such that the algorithm in Eq.* (38) *converges w.p.1 if $\frac{K}{2T} < C_0$.*

Proof. We first re-write the Eq. (38) as on ODE:

$$\begin{aligned} \frac{d\boldsymbol{\phi}}{dt} &= \mathbb{E}_{\boldsymbol{\phi}}\left[\boldsymbol{\omega}_s^\top \left(\boldsymbol{r}(s, a, s') + \gamma \boldsymbol{\omega}_{s'}^\top \boldsymbol{\phi} - \boldsymbol{\omega}_s^\top \boldsymbol{\phi}\right)\right] \\ &= \mathbb{E}_{\boldsymbol{\phi}}\left[\boldsymbol{\omega}_s^\top (\gamma \boldsymbol{\omega}_{s'}^\top - \boldsymbol{\omega}_s^\top)\right] \boldsymbol{\phi} + \mathbb{E}_{\boldsymbol{\phi}}\left[\boldsymbol{\omega}_s^\top (\boldsymbol{r}(s, a, s'))\right] \\ &= \boldsymbol{A}_\phi \boldsymbol{\phi} + \boldsymbol{b}_\phi \end{aligned} \tag{41}$$

Notice that we use a vector for considering the updating rule for the Q function of each agent. We can easily know that necessity condition of the equilibrium is that it must follow $\boldsymbol{\phi}^* = \boldsymbol{A}_{\phi^*}^{-1} \boldsymbol{b}_{\phi^*}$. The existence of the such equilibrium has been restricted in the scenario that meets Assumption 3. In the proof of Theorem 1 we have already pointed out that under the

Assumption 3, the existing equilibrium, either in the form of global equilibrium or in the form of saddle-point equilibrium, is unique.

Let $\tilde{\boldsymbol{\phi}} = \boldsymbol{\phi}_t - \boldsymbol{\phi}^*$, we have:

$$\begin{aligned}
\frac{d}{dt}||\tilde{\boldsymbol{\phi}}||_2 &= 2\boldsymbol{\phi} \cdot \frac{d\boldsymbol{\phi}}{dt} - 2\boldsymbol{\phi}^* \cdot \frac{d\boldsymbol{\phi}}{dt} \\
&= 2\tilde{\boldsymbol{\phi}}^\top (\boldsymbol{A}_\phi \boldsymbol{\phi} + \boldsymbol{b}_\phi - \boldsymbol{A}_{\phi^*} \boldsymbol{\phi}^* - \boldsymbol{b}_{\phi^*}) \\
&= 2\tilde{\boldsymbol{\phi}}^\top (\boldsymbol{A}_{\phi^*} \boldsymbol{\phi} - \boldsymbol{A}_{\phi^*} \boldsymbol{\phi} + \boldsymbol{A}_\phi \boldsymbol{\phi} + \boldsymbol{b}_\phi - \boldsymbol{A}_{\phi^*} \boldsymbol{\phi}^* - \boldsymbol{b}_{\phi^*}) \\
&= 2\tilde{\boldsymbol{\phi}}^\top \boldsymbol{A}_{\phi^*} \tilde{\boldsymbol{\phi}} + 2\tilde{\boldsymbol{\phi}}^\top (\boldsymbol{A}_\phi - \boldsymbol{A}_{\phi^*}) \boldsymbol{\phi} + 2\tilde{\boldsymbol{\phi}}^\top (\boldsymbol{b}_\phi - \boldsymbol{b}_{\phi^*}) \\
&\leq 2\tilde{\boldsymbol{\phi}}^\top \left(\boldsymbol{A}_{\phi^*} + \sup_{\boldsymbol{\phi}} ||\boldsymbol{A}_\phi - \boldsymbol{A}_{\phi^*}||_2 + \sup_{\boldsymbol{\phi}} \frac{||\boldsymbol{b}_\phi - \boldsymbol{b}_\phi^*||_2}{||\boldsymbol{\phi} - \boldsymbol{\phi}^*||_2} \right) \tilde{\boldsymbol{\phi}}
\end{aligned} \tag{42}$$

As we know that the policy $\boldsymbol{\pi}_{\phi_t}$ is Lipschitz w.r.t ϕ_t, this implies that $\boldsymbol{A}_\phi$ and $\boldsymbol{\nu}_\phi$ are also Lipschitz continuous w.r.t to ϕ. In other words, if $\frac{K}{2T} \leq C_0$ is sufficiently small and close to zero, then the norm term of $\left(\sup_{\boldsymbol{\phi}} ||\boldsymbol{A}_\phi - \boldsymbol{A}_{\phi^*}||_2 + \sup_{\boldsymbol{\phi}} \frac{||\boldsymbol{b}_\phi - \boldsymbol{b}_\phi^*||_2}{||\boldsymbol{\phi} - \boldsymbol{\phi}^*||_2} \right)$ goes to zero. Considering near the equilibrium point ϕ^*, $\boldsymbol{A}_{\phi^*}$ is a negative definite matrix, the Eq. (42) tends to be negative definite as well, so the ODE in Eq.(41) is globally asymptotically stable and the conclusion of the theorem follows. □