Mean-Field Sampling for Cooperative Multi-Agent Reinforcement Learning

Emile Anand[1,3], **Ishani Karmarkar**[2], **Guannan Qu**[3]

[1]School of Computer Science, Georgia Institute of Technology
[2] Institute of Computational and Mathematical Engineering, Stanford University
[3] Department of Electrical and Computer Engineering, Carnegie Mellon University
emile@gatech.edu, ishanik@stanford.edu, gqu@andrew.cmu.edu

Abstract

Designing efficient algorithms for multi-agent reinforcement learning (MARL) is fundamentally challenging due to the fact that the size of the joint state and action spaces are exponentially large in the number of agents. These difficulties are exacerbated when balancing sequential global decision-making with local agent interactions. In this work, we propose a new algorithm `SUBSAMPLE-MFQ` (**Subsample**-**M**ean-**F**ield-**Q**-learning) and a decentralized randomized policy for a system with n agents. For $k \leq n$, our algorithm system learns a policy for the system in time polynomial in k. We show that this learned policy converges to the optimal policy in the order of $\widetilde{O}(1/\sqrt{k})$ as the number of subsampled agents k increases. We validate our method empirically on Gaussian squeeze and global exploration settings.

Extended version —
https://www.arxiv.org/pdf/2412.00661

Introduction

Reinforcement Learning (RL) has become a popular learning framework to solve sequential decision making problems in unknown environments, and has achieved tremendous success in a wide array of domains such as playing the game of Go (Silver et al. 2016), robotic control (Kober, Bagnell, and Peters 2013), and autonomous driving (Kiran et al. 2022; Lin et al. 2023). A critical feature of most real-world systems is their uncertain nature, and consequently RL has emerged as a powerful tool for learning optimal policies for multi-agent systems to operate in unknown environments (Kim and Giannakis 2017; Zhang, Yang, and Başar 2021; Lin et al. 2024; Anand and Qu 2024). While the early literature on RL primarily focused on the single-agent setting, multi-agent reinforcement learning (MARL) has recently achieved impressive successes in a broad range of areas, such as coordination of robotic swarms (Preiss et al. 2017), self-driving vehicles (DeWeese and Qu 2024), real-time bidding (Jin et al. 2018), ride-sharing (Li et al. 2019), and stochastic games (Jin et al. 2020).

Despite growing interest in multi-agent RL (MARL), extending RL to multi-agent settings poses significant computational challenges due to the *curse of dimensionality* (Sayin et al. 2021). Even if the individual agents' state or action spaces are small, the global state space or action space can take values from a set with size that is exponentially large as a function of the number of agents. For example, even *model-free* RL algorithms such as temporal difference (TD) learning (Sutton et al. 1999) or tabular Q-learning require computing and storing a Q-function (Bertsekas and Tsitsiklis 1996) that is as large as the state-action space. Unfortunately, in MARL, the joint state-action space is exponentially large in the number of agents. In the case where the system's rewards are not discounted, reinforcement learning on multi-agent systems is provably NP-hard (Blondel and Tsitsiklis 2000), and such scalability issues have been observed in the literature in a variety of settings (Guestrin et al. 2003; Papadimitriou and Tsitsiklis 1999; Littman 1994). Independent Q-learning (Tan 1997) seeks to overcome these scalability challenges by considering other agents as a part of the environment; however, this often fails to capture a key feature of MARL: *inter-agent interactions*.

Even in the fully cooperative regime, MARL is fundamentally difficult, since agents in the real-world not only interact with the environment but also with each other (Shapley 1953). An exciting line of work that addresses this intractability is mean-field MARL (Lasry and Lions 2007; Yang et al. 2018; Gu et al. 2021, 2022b,a; Hu et al. 2023). The mean-field approach assumes that all the agents are homogeneous in their state and action spaces, enabling their interactions to be approximated by a two-agent setting: here, each agent interacts with a representative "mean agent" which is the empirical distribution of states of all other agents. Under these homogeneity assumptions, mean-field MARL allows learning of optimal policies with a sample complexity that is polynomial in the number of agents. However, when the number of homogeneous agents is large, storing a polynomially-large Q table (where the polynomial's degree depends on the size of the state space for a single agent) can still be infeasible. Therefore, we ask:

Can we design an efficient and approximately optimal MARL algorithm for policy-learning in a cooperative multi-agent system with many agents?

Contributions. We answer this question affirmatively. Our key contributions are outlined below.

- **Subsampling Algorithm.** We propose a novel algorithm SUBSAMPLE-MFQ to address the challenge of MARL with a large number of local agents. We model the problem as a Markov Decision Process (MDP) with a global agent and n local agents. SUBSAMPLE-MFQ selects $k \leq n$ local agents to learn a deterministic policy $\widehat{\pi}_k^{\text{est}}$, by applying mean-field value iteration on a subsystem with k local agents to learn $\widehat{Q}_k^{\text{est}}$, which can be viewed as a smaller Q function. It then deploys a stochastic policy $\widehat{\pi}_k$ which works as follows: the global agent uniformly samples k local agents at each step and uses $\widehat{\pi}_k$ to determine its action, while each local agent uniformly samples $k-1$ other local agents and uses $\widehat{\pi}_k$ to determine its action.
- **Sample Complexity and Theoretical Guarantee.** As the number of local agents increases, the size of $\widehat{Q}_k$ scales polynomially with k, rather than polynomially with n as in mean-field MARL. Analogously, when the size of the local agent's state space grows, the size of $\widehat{Q}_k$ scales exponentially with k, rather than exponentially with n, as in traditional Q-learning). The key analytic technique underlying our results is a novel MDP sampling result. Through it, we show that the performance gap between π_k^{est} and the optimal policy π^* is $\widetilde{O}(1/\sqrt{k})$. The choice of k reveals a fundamental trade-off between the size of the Q-table and the optimality of π_k^{est}. When $k = O(\log n)$, SUBSAMPLE-MFQ is the first centralized MARL algorithm to achieve a polylogarithmic run-time in n, representing an exponential speedup over the previously best-known polytime mean-field MARL methods, while maintaining a decaying optimality gap.
- **Numerical Simulations.** We evaluate the effectiveness of SUBSAMPLE-MFQ in two scenarios: a bounding box problem, and a Gaussian squeeze problem. Our experiments reveal a monotonic improvement in the learned policies as $k \to n$, providing a substantial speedup over mean-field Q-learning.

While our results are theoretical in nature, it is our hope that SUBSAMPLE-MFQ will lead to further exploration into the potential of subsampling in general stochastic/Markovian games and networked multi-agent RL. For further details, we refer the reader to the complete version of this manuscript on arXiV.

Related Literature

MARL has a rich history, starting with early works on Markov games used to characterize the decision-making process (Littman 1994; Sutton et al. 1999), which can be regarded as a multi-agent extension of the Markov Decision Process (MDP). MARL has since been actively studied (Zhang, Yang, and Başar 2021) in a broad range of settings. MARL is most similar to the category of "succinctly described" MDPs (Blondel and Tsitsiklis 2000), where the state/action space is a product space formed by the individual state/action spaces of multiple agents, and where the agents interact to maximize an objective. A promising line of research that has emerged over recent years constrains the problem to sparse networked instances to enforce local interactions between agents (Qu et al. 2020; Lin et al. 2020; Mondal et al. 2022). In this formulation, the agents correspond to vertices on a graph who only interact with nearby agents. By exploiting Gamarnik's correlation decay property from combinatorial optimization (Gamarnik, Goldberg, and Weber 2009), they overcome the curse of dimensionality by simplifying the problem to only search over the policy space derived from the truncated graph to learn approximately optimal solutions. However, as the underlying network structure becomes dense with many local interactions, the neighborhood of each agent gets large, causing these algorithms to become intractable.

Mean-Field RL. Under assumptions of homogeneity in the state/action spaces of the agents, the problem of densely networked multi-agent RL was answered in (Yang et al. 2018; Gu et al. 2021) by approximating the solution with a mean-field approach where the approximation error scales in $O(1/\sqrt{n})$. To avoid designing algorithms on probability spaces, they study MARL under Pareto optimality and use the law of large numbers to consider a lifted space with a *mean* agent that aggregates the system's rewards and dynamics. It then applies kernel regression on ϵ-nets of the lifted space to design policies in time polynomial in n. In contrast, our work achieves *subpolynomial* runtimes by directly sampling from this mean-field distribution. (Cui and Koeppl 2022) introduce heterogeneity to mean-field MARL by modeling non-uniform interactions through graphons; however, these methods make critical assumptions on the existence of a sequence of graphons converging in cut-norm to the finite instance. In the cooperative setting, (Cui, Fabian, and Koeppl 2023) considers a mean-field setting with q types of homogeneous agents; however, their approach does not converge to the optimum policy.

Structured RL. Our work is related to factored MDPs and exogenous MDPs. In factored MDPs, there is a global action affecting every agent whereas in our case, each agent has its own action (Min et al. 2023; Lauer and Riedmiller 2000). Our result has a similar flavor to MDPs with exogenous inputs from learning theory, (Dietterich, Trimponias, and Chen 2018; Foster et al. 2022; Anand and Qu 2024), where our subsampling algorithm treats each sampled state as an endogenous state, but where the exogenous dependencies can be dynamic.

Miscellaneous. Our work adds to the growing literature on the Centralized Training with Decentralized Execution regime (Zhou et al. 2023), as our algorithm learns a provably approximately optimal policy using centralized information, but makes decisions using only local information during execution. In the distributed setting, V-learning (Jin et al. 2020) reduces the dependence of the product action space to an additive dependence. In contrast, our work *further* accomplishes a reduction on the complexity of the joint state space, which has not been previously accomplished. Finally, one can approximate the Q-table through linear function approximation (Jin et al. 2020) which significantly reduces the computational complexity. However, achieving theoretical bounds on the performance loss caused by function approx-

imation is intractable without making strong assumptions such as Linear Bellman completeness (Golowich and Moitra 2024) or low Bellman-Eluder dimension (Jin, Liu, and Miryoosefi 2021). While our work primarily considers the finite tabular setting, we also extend it to the non-tabular setting, under Linear Bellman completeness assumptions.

Preliminaries

In this section, we formally introduce the problem, state some examples for our setting, and provide technical details of the mean-field and Q-learning techniques that will be used throughout the paper.

Notation. For $k, n \in \mathbb{N}$ where $k \leq n$, let $\binom{[n]}{k}$ denote the set of k-sized subsets of $[n] = \{1, \ldots, n\}$. For any vector $z \in \mathbb{R}^d$, let $\|z\|_1$ and $\|z\|_\infty$ denote the standard ℓ_1 and ℓ_∞ norms of z respectively. Let $\|\mathbf{A}\|_1$ denote the matrix ℓ_1-norm of $\mathbf{A} \in \mathbb{R}^{n \times m}$. Given a collection of variables $s_1, \ldots, s_n$ the shorthand s_Δ denotes the set $\{s_i : i \in \Delta\}$ for $\Delta \subseteq [n]$. We use $\widetilde{O}(\cdot)$ to suppress polylogarithmic factors in all problem parameters except n. For a discrete measurable space $(\mathcal{X}, \mathcal{F})$, the total variation distance between probability measures ρ_1, ρ_2 is given by $\mathrm{TV}(\rho_1, \rho_2) = \frac{1}{2}\sum_{x \in \mathcal{X}} |\rho_1(x) - \rho_2(x)|$. Next, $x \sim \mathcal{D}(\cdot)$ denotes that x is a random element sampled from a probability distribution $\mathcal{D}$, and we denote that x is a random sample from the uniform distribution over a finite set Ω by $x \sim \mathcal{U}(\Omega)$.

Problem Formulation

We consider a system of $n+1$ agents, where agent g is a "global decision making agent" and the remaining n agents, denoted by $[n]$, are "local agents." At time step t, the agents are in state $s(t) = (s_g(t), s_1(t), ..., s_n(t)) \in \mathcal{S} := \mathcal{S}_g \times \mathcal{S}_l^n$, where $s_g(t) \in \mathcal{S}_g$ denotes the global agent's state, and for each $i \in [n]$, $s_i(t) \in \mathcal{S}_l$ denotes the state of the i'th local agent. The agents cooperatively select actions $a(t) = (a_g(t), a_1(t), ..., a_n(t)) \in \mathcal{A}$ where $a_g(t) \in \mathcal{A}_g$ denotes the global agent's action and $a_i(t) \in \mathcal{A}_l$ denotes the i'th local agent's action. At each time-step t, the next state for all the agents is independently generated by stochastic transition kernels $P_g : \mathcal{S}_g \times \mathcal{S}_g \times \mathcal{A}_g \to [0,1]$ and $P_l : \mathcal{S}_l \times \mathcal{S}_l \times \mathcal{S}_g \times \mathcal{A}_g \to [0,1]$ as follows:

$$s_g(t+1) \sim P_g(\cdot | s_g(t), a_g(t)), \tag{1}$$

$$s_i(t+1) \sim P_l(\cdot | s_i(t), s_g(t), a_i(t)), \forall i \in [n]. \tag{2}$$

The system then collects a structured stage reward $r(s(t), a(t))$ where the reward $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ depends on $s(t)$ and $a(t)$ through eq. (3), and where the choice of functions r_g and r_l is typically application specific.

$$r(s,a) = \underbrace{r_g(s_g, a_g)}_{\text{global component}} + \frac{1}{n}\sum_{i \in [n]} \underbrace{r_l(s_i, s_g, a_i)}_{\text{local component}} \tag{3}$$

We define a policy π as a mapping from $\mathcal{S}$ to $\mathcal{A}$ where we want the policy π to maximize the value function which is defined for each $s \in S$ as the expected discounted reward

$$V^\pi(s) = \mathbb{E}_{a(t) \sim \pi(\cdot|s)} \left[\sum_{t=0}^{\infty} \gamma^t r(s(t), a(t)) | s(0) = s \right], \tag{4}$$

where $\gamma \in (0,1)$ is a discounting factor.

Notably, the cardinality of the search space simplex for the optimal policy is $|\mathcal{S}_g||\mathcal{S}_l|^n|\mathcal{A}_g||\mathcal{A}_l|^n$, which is exponential in the number of agents. When noting that the local agents are all homogeneous, and therefore permutation-invariant with respect to the rewards of the system (the order of the other agents does not matter to any single decision-making agent), techniques from mean-field MARL restrict the cardinality of the search space simplex for the optimal policy to $|\mathcal{S}_g||\mathcal{A}_g||\mathcal{S}_l||\mathcal{A}_l|n^{|\mathcal{S}_l||\mathcal{A}_l|}$, reducing the exponential complexity on n to a polynomial complexity on n. In practical systems, when n is large, the $\mathrm{poly}(n)$ sample-complexity may still be computationally infeasible. Therefore, the goal of this problem is to learn an approximately optimal policy with *subpolynomial sample complexity*, further overcoming the curse of dimensionality.

Example 0.1 (Gaussian Squeeze) In this task, n homogeneous agents determine their individual action a_i to jointly maximize the objective $r(x) = xe^{-(x-\mu)^2/\sigma^2}$, where $x = \sum_{i=1}^n a_i$, $a_i = \{0, \ldots, 9\}$, and μ and σ are the pre-defined mean and variance of the system. In scenarios of traffic congestion, each agent $i \in [n]$ is a traffic controller trying to send a_i vehicles into the main road, where controllers coordinate with each other to avoid congestion, hence avoiding either over-use or under-use, thereby contributing to the entire system. This GS problem serves as an ablation study on the impact of subsampling for MARL.

Example 0.2 (Constrained Exploration) Consider an $M \times M$ grid. Each agent's state is a coordinate in $[M] \times [M]$. The state represents the *center of a* $d \times d$ *box* where the global agent wishes to constrain the local agents' movements. Initially, all agents are in the same location. At each time-step, the local agents take actions $a_i(t) \in \mathbb{R}^2$ (e.g., up, down, left, right) to transition between states and collect stage rewards. The transition kernel ensures that local agents remain within the $d \times d$ box dictated by the global agent, by only using knowledge of $a_i(t), s_g(t)$, and $s_i(t)$. In warehouse settings where some shelves have collapsed, creating hazardous or inaccessible areas, we want agents to clean these areas. However, exploration in these regions may be challenging due to physical constraints or safety concerns. Therefore, through an appropriate design of the reward and transition functions, the global agent could guide the local agents to focus on specific $d \times d$ grids, allowing efficient cleanup while avoiding unnecessary risk or inefficiency.

Due to space constraints, we leave the details of the experiments to section J of the supplementary material. To efficiently learn policies that maximize the objective, we make the following standard assumptions:

Assumption 0.3 (Finite state/action spaces) We assume that the state and action spaces of all the agents in the MARL game are finite: $|\mathcal{S}_l|, |\mathcal{S}_g|, |\mathcal{A}_g|, |\mathcal{A}_l| < \infty$. Appendix H of the supplementary material weakens this assumption to the non-tabular setting with infinite sets.

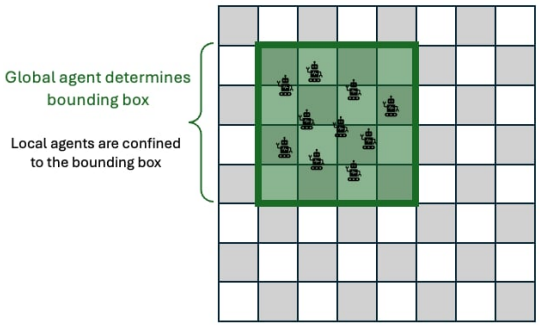

Figure 1: Constrained exploration for warehouse accidents.

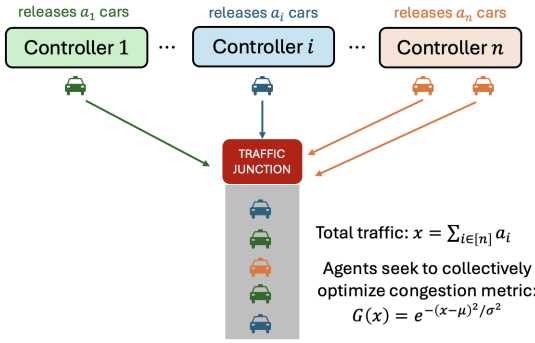

Figure 2: Traffic congestion settings with Gaussian squeeze.

Assumption 0.4 (Bounded rewards) The global and local components of the reward function are bounded. Specifically, $\|r_g(\cdot,\cdot)\|_\infty \leq \widetilde{r}_g$, and $\|r_l(\cdot,\cdot,\cdot)\|_\infty \leq \widetilde{r}_l$. This implies that $\|r(\cdot,\cdot)\|_\infty \leq \widetilde{r}_g + \widetilde{r}_l := \widetilde{r}$.

Definition 0.5 (ϵ-optimal policy) Given an objective function V and policy simplex Π, a policy $\pi \in \Pi$ is ϵ-optimal if $V(\pi) \geq \sup_{\pi^* \in \Pi} V(\pi^*) - \epsilon$.

Remark 0.6 Heterogeneity among the local agents can be captured by modeling agent types as part of the agent state: assign a type $\varepsilon_i \in \mathcal{E}$ to each local agent $i \in [n]$ by letting $\mathcal{S}_l = \mathcal{E} \times \mathcal{S}_l'$, where $\mathcal{E}$ is a set of possible types that are treated as a fixed part of the agent's state. The transition and reward functions can vary depending on the agent's type. The global agent can provide unique signals to local agents of each type by letting $s_g \in \mathcal{S}_g$ and $a_g \in \mathcal{A}_g$ denote a state/action vector where each element matches to a type $\varepsilon \in \mathcal{E}$.

Technical Background

Q-learning. To provide background for the analysis in this paper, we review a few key technical concepts in RL. At the core of the standard Q-learning framework (Watkins and Dayan 1992) for offline-RL is the Q-function $Q : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. Q-learning seeks to produce a policy $\pi^*(\cdot|s)$ that maximizes the expected infinite horizon discounted reward. For any policy π, $Q^\pi(s,a) = \mathbb{E}^\pi[\sum_{t=0}^\infty \gamma^t r(s(t),a(t))|s(0) = s, a(0) = a]$. One approach to learning the optimal policy $\pi^*(\cdot|s)$ is dynamic programming, where the Q-function is iteratively updated using value-iteration: $Q^0(s,a) = 0$, for all $(s,a) \in \mathcal{S} \times \mathcal{A}$. Then, for all $t \in [T]$, $Q^{t+1}(s,a) = \mathcal{T}Q^t(s,a)$, where $\mathcal{T}$ is the Bellman operator defined as

$$\mathcal{T}Q^t(s,a) = r(s,a) + \gamma \mathbb{E}_{\substack{s_g' \sim P_g(\cdot|s_g,a), \\ s_i' \sim P_l(\cdot|s_i,s_g), \forall i \in [n]}} \max_{a' \in \mathcal{A}_g \times \mathcal{A}_l^n} Q^t(s',a').$$

The Bellman operator $\mathcal{T}$ is γ-contractive, which ensures the existence of a unique fixed-point Q^* such that $\mathcal{T}Q^* = Q^*$, by the Banach fixed-point theorem (Banach 1922). Here, the optimal policy is the deterministic greedy policy $\pi^* : \mathcal{S}_g \times \mathcal{S}_l^n \to \mathcal{A}_g \times \mathcal{A}_l^n$, where $\pi^*(s) = \arg\max_{a \in \mathcal{A}_g \times \mathcal{A}_l^n} Q^*(s,a)$. However, the complexity of a single update to the Q-function is $O(|\mathcal{S}_g||\mathcal{S}_l|^n|\mathcal{A}_g||\mathcal{A}_l|^n)$, which grows exponentially with n. As the number of local agents increases ($n \gg |\mathcal{S}_l|$), this exponential update complexity renders Q-learning impractical.

Mean-field Transformation. To address this, mean-field MARL (under homogeneity assumptions) studies the distribution function $F_{z_{[n]}} : \mathcal{Z}_l \to \mathbb{R}$, where $\mathcal{Z}_l := \mathcal{S}_l \times \mathcal{A}_l$, defined for all $z := (z_s, z_a) \in \mathcal{S}_l \times \mathcal{A}_l$ by

$$F_{z_{[n]}}(z) := \frac{1}{n} \sum_{i=1}^n \mathbf{1}\{s_i = z_s, a_i = z_a\}. \tag{5}$$

Let $\mu_n(\mathcal{Z}_l) = \{\frac{b}{n} | b \in \{0,\dots,n\}\}^{|\mathcal{S}_l| \times |\mathcal{A}_l|}$ be the space of $|\mathcal{S}_l| \times |\mathcal{A}_l|$-sized tables, where each entry is an element of $\{0, \frac{1}{n}, \frac{2}{n}, \dots, 1\}$. In this space, $F_{z_{[n]}} \in \mu_n(\mathcal{Z}_l)$ where $F_{z_{[n]}}$ represents the proportion of agents in each state/action pair. The Q-function is permutation-invariant in the local agents, since permuting the labels of homogeneous local agents with the same state will not change the action of the decision-making agent. Hence, $Q(s_g, s_{[n]}, a_g, a_{[n]}) = \widehat{Q}(s_g, s_1, a_g, a_1, F_{z_{[n]\setminus 1}})$. Here, $\widehat{Q} : \mathcal{S}_g \times \mathcal{S}_l \times \mathcal{A}_g \times \mathcal{A}_1 \times \mu_{n-1}(\mathcal{Z}_l) \to \mathbb{R}$ is a reparameterized Q-function learned by mean-field value iteration: one initializes $\widehat{Q}^0(s_g, s_1, a_g, a_1, F_{z_{[n]\setminus 1}}) = 0$. At each time-step t, we update $\widehat{Q}$ as $\widehat{Q}^{t+1}(s_g, s_1, a_g, a_1, F_{z_{[n]\setminus 1}}) = \widehat{\mathcal{T}}\widehat{Q}^t(s_g, s_1, a_g, a_1, F_{z_{[n]\setminus 1}})$, where $\widehat{\mathcal{T}}$ is the Bellman operator in distribution space:

$$\widehat{\mathcal{T}}\widehat{Q}^t(s_g, s_1, a_g, a_1, F_{z_{[n]\setminus 1}}) = r(s,a) + \gamma \mathbb{E}_{\substack{s_g' \sim P_g(\cdot|s_g,a_g) \\ s_i' \sim P_l(\cdot|s_i,s_g,a_i) \\ \forall i \in [n]}} \max_{\substack{(a_g', a_1', a_{[n]\setminus 1}') \\ \in \mathcal{A}_g \times \mathcal{A}_l \times \mathcal{A}_l^{n-1}}} \widehat{Q}^t(s_g', s_1', a_g', a_1', F_{z'_{[n]\setminus 1}})$$

Since $\mathcal{T}$ is a γ-contraction, so is $\widehat{\mathcal{T}}$. Hence, $\widehat{T}$ has a unique fixed-point $\widehat{Q}^*$ such that $\widehat{Q}^*(s_g, s_1, a_g, a_1, F_{z_{[n]\setminus 1}}) = Q^*(s_g, s_{[n]}, a_g, a_{[n]})$ and the optimal policy is in turn given by the deterministic greedy policy given by

$$\widehat{\pi}^*(s_g, s_1, F_{s_{[n]\setminus 1}}) = \arg\max_{\substack{(a_g, a_1, a_{[n]\setminus 1}) \\ \in \mathcal{A}_g \times \mathcal{A}_l \times \mathcal{A}_l^{n-1}}} \widehat{Q}^*(s_g, s_1, a_g, a_1, F_{s_{[n]\setminus 1}, a_{[n]\setminus 1}}, a_g).$$

$O(|\mathcal{S}_g||\mathcal{A}_g||\mathcal{Z}_l|n^{|\mathcal{Z}_l|})$ is the update complexity to the $\widehat{Q}$-function, which scales polynomially in n.

Remark 0.7 The solution offered by mean-field value iteration and standard Q-learning requires a sample complexity of $\min\{\widetilde{O}(|\mathcal{S}_g||\mathcal{A}_g||\mathcal{Z}_l|^n), \widetilde{O}(|\mathcal{S}_g||\mathcal{A}_g||\mathcal{Z}_l|n^{|\mathcal{Z}_l|})\}$, where one uses standard Q-learning when $|\mathcal{Z}_l|^{n-1} < n^{|\mathcal{Z}_l|}$, and mean-field value iteration otherwise. In each of these regimes, as n scales, the update complexity can become incredibly computationally intensive. Therefore, we introduce the SUBSAMPLE-MFQ algorithm to mitigate the cost of scaling the number of local agents.

Method and Theoretical Results

In this section, we propose the SUBSAMPLE-MFQ algorithm to overcome the polynomial (in n) sample complexity of mean-field value iteration and the exponential (in n) sample complexity of traditional Q-learning. In our algorithm, the global agent randomly samples a subset of local agents $\Delta \subseteq [n]$ such that $|\Delta| = k$, for $k \leq n$. It ignores all other local agents $[n] \setminus \Delta$, and performs value iteration to learn the Q-function $\widehat{Q}^*_{k,m}$ and policy $\widehat{\pi}^*_{k,m}$ for this surrogate subsystem of k local agents, where m is the number of samples used to update the Q-functions' estimates of the unknown system. Here, When $|\mathcal{Z}_l|^{k-1} < k^{|\mathcal{Z}_l|}$, the algorithm uses traditional value-iteration (algorithm 1), and when $|\mathcal{Z}_l|^{k-1} > k^{|\mathcal{Z}_l|}$, it uses mean-field value iteration (algorithm 2). The surrogate reward gained by this subsystem at each time step is $r_\Delta : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$:

$$r_\Delta(s, a) = r_g(s_g, a_g) + \frac{1}{|\Delta|} \sum_{i \in \Delta} r_l(s_g, s_i, a_i). \quad (6)$$

To convert the optimality of each agent's action within the k local-agent subsystem to an approximate optimality on the full n-agent system, we use a randomized policy $\pi^{\text{est}}_{k,m}$ (algorithm 3), where the global agent samples $\Delta \in \mathcal{U}\binom{[n]}{k}$ at each time-step to derive the action $a_g \leftarrow \widehat{\pi}^*_{k,m}(s_g, s_\Delta)$, and where each local i agent samples $k-1$ other local agents Δ_i to derive the action $\widehat{\pi}^*_{k,m}(s_g, s_i, s_{\Delta_i})$. Finally, theorem 0.11 shows that the policy $\pi^{\text{est}}_{k,m}$ converges to the optimal policy π^* as $k \to n$. We first present algorithms 1 and 2 (SUBSAMPLE-MFQ: Learning) and algorithm 3 (SUBSAMPLE-MFQ: Execution), which we describe below. For this, a crucial characterization is the notion of the empirical distribution function:

Definition 0.8 (Empirical Distribution Function) For any population $(z_1, \ldots, z_n) \in \mathcal{Z}_l^n$, where $\mathcal{Z}_l := \mathcal{S}_l \times \mathcal{A}_l$, define the empirical distribution function $F_{z_\Delta} : \mathcal{Z}_l \to \mathbb{R}_+$ for all $z := (z_s, z_a) \in \mathcal{S}_l \times \mathcal{A}_l$ and for all $\Delta \subseteq [n]$ such that $|\Delta| = k$ by:

$$F_{z_\Delta}(x) := F_{s_\Delta, a_\Delta}(x) := \frac{1}{k} \sum_{i \in \Delta} \mathbf{1}\{s_i = z_s, a_i = z_a\}. \quad (7)$$

Let $\mu_k(\mathcal{Z}_l) := \left\{\frac{b}{k} | b \in \{0, \ldots, k\}\right\}^{|\mathcal{S}_l| \times |\mathcal{A}_l|}$ be the space of $|\mathcal{S}_l| \times |\mathcal{A}_l|$-length vectors where each entry in a vector is an element of $\{0, \frac{1}{k}, \frac{2}{k}, \ldots, 1\}$ such that $F_{z_\Delta} \in \mu_k(\mathcal{Z}_l)$. Here, F_{z_Δ} is the proportion of agents in the k-local-agent subsystem at each state.

Algorithms 1 and 2 (Offline learning). Let $m \in \mathbb{N}$ denote the sample size for the learning algorithm with sampling parameter $k \leq n$. When $|\mathcal{Z}_l|^{k-1} \leq k^{|\mathcal{Z}_l|}$, we empirically learn the optimal Q-function for a subsystem with k-local agents denoted by $\widehat{Q}^{\text{est}}_{k,m} : \mathcal{S}_g \times \mathcal{S}_l^k \times \mathcal{A}_g \times \mathcal{A}_l^k \to \mathbb{R}$: set $\widehat{Q}^0_{k,m}(s_g, s_\Delta, a_g, a_\Delta) = 0$ for all $(s_g, s_\Delta, a_g, a_\Delta) \in \mathcal{S}_g \times \mathcal{S}_l^k \times \mathcal{A}_g \times \mathcal{A}_l^k$. At time step t, set $\widehat{Q}^{t+1}_{k,m}(s_g, s_\Delta, a_g, a_\Delta) = \widetilde{\mathcal{T}}_{k,m} \widehat{Q}^t_{k,m}(s_g, s_\Delta, a_g, a_\Delta)$, where $\widetilde{\mathcal{T}}_{k,m}$ is the *empirically adapted Bellman operator* in eq. (8).

When $|\mathcal{Z}_l|^{k-1} > k^{|\mathcal{Z}_l|}$, we empirically learn the optimal mean-field Q-function for a k local agent system, denoted (with abuse of notation) by $\widehat{Q}^{\text{est}}_{k,m} : \mathcal{S}_g \times \mathcal{S}_1 \times \mu_{k-1}(\mathcal{Z}_l) \times \mathcal{A}_g \times \mathcal{A}_l \to \mathbb{R}$. For all $(s_g, s_1, F_{z_{\tilde{\Delta}}}, a_1, a_g) \in \mathcal{S}_g \times \mathcal{S}_l \times \mu_{k-1}(\mathcal{Z}_l) \times \mathcal{A}_l \times \mathcal{A}_g$, set $\widehat{Q}^0_{k,m}(s_g, s_1, F_{z_{\tilde{\Delta}}}, a_1, a_g) = 0$. Then, at each time-step t, we set $\widehat{Q}^{t+1}_{k,m}(s_g, s_1, F_{z_{\tilde{\Delta}}}, a_1, a_g) = \widehat{\mathcal{T}}_{k,m} \widehat{Q}^t_{k,m}(s_g, s_1, F_{z_{\tilde{\Delta}}}, a_1, a_g)$, where $\widehat{\mathcal{T}}_{k,m}$ is the *empirically adapted mean-field Bellman operator* in eq. (9).

Since the system is unknown, $\mathcal{T}_{k,m}$ and $\widehat{\mathcal{T}}_{k,m}$ cannot compute the direct expectation from the Bellman operator and instead draw m random samples $s_g^j \sim P_g(\cdot|s_g, a_g)$ and $s_i^j \sim P_l(\cdot|s_i, s_g, a_i)$ for each $j \in [m]$, $i \in \Delta$:

$$\begin{aligned} &\widetilde{\mathcal{T}}_{k,m} \widehat{Q}^t_{k,m}(s_g, s_\Delta, a_g, a_\Delta) \\ &= r_\Delta(s, a) \\ &+ \frac{\gamma}{m} \sum_{j \in [m]} \max_{a'_g \in \mathcal{A}_g, a'_\Delta \in \mathcal{A}_l^k} \widehat{Q}^t_{k,m}(s_g^j, s_\Delta^j, a'_g, a'_\Delta). \end{aligned} \quad (8)$$

$$\begin{aligned} &\widehat{\mathcal{T}}_{k,m} \widehat{Q}^t_{k,m}(s_g, s_1, F_{z_{\tilde{\Delta}}}, a_1, a_g) = r_\Delta(s, a) \\ &+ \frac{\gamma}{m} \sum_{j \in [m]} \max_{\substack{a'_g \in \mathcal{A}_g, \\ a'_1 \in \mathcal{A}_l, \\ F_{a'_{\tilde{\Delta}}} \in \mu_{k-1}(\mathcal{A}_l)}} \widehat{Q}^t_{k,m}(s_g^j, s_1^j, F_{s^j_{\tilde{\Delta}}, a_{\tilde{\Delta}'}}, a'_1, a'_g). \end{aligned} \quad (9)$$

$\widehat{Q}^t_{k,m}$ depends on s_Δ and a_Δ through F_{z_Δ}, and $\widetilde{\mathcal{T}}_{k,m}$/$\widehat{\mathcal{T}}_{k,m}$ are γ-contractive. So, algorithms 1 and 2 apply value iteration with their Bellman operator until $\widehat{Q}_{k,m}$ converges to a fixed point satisfying $\widetilde{\mathcal{T}}_{k,m} \widehat{Q}^{\text{est}}_{k,m} = \widehat{Q}^{\text{est}}_{k,m}$ and $\widehat{\mathcal{T}}_{k,m} \widehat{Q}^{\text{est}}_{k,m} = \widehat{Q}^{\text{est}}_{k,m}$, yielding equivalent deterministic policies $\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_\Delta)$ and $\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_1, F_{s_{\tilde{\Delta}}})$:

$$\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_\Delta) = \operatorname*{arg\,max}_{a_g \in \mathcal{A}_g, a_\Delta \in \mathcal{A}_l^k} \widehat{Q}^{\text{est}}_{k,m}(s_g, s_\Delta, a_g, a_\Delta)$$

$$\begin{aligned} &\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_1, F_{s_{\tilde{\Delta}}}) \\ &= \operatorname*{arg\,max}_{a_g \in \mathcal{A}_g, a_1 \in \mathcal{A}_l, F_{a_{\tilde{\Delta}}} \in \mu_{k-1}(\mathcal{A}_l)} \widehat{Q}^{\text{est}}_{k,m}(s_g, s_1, F_{z_{\tilde{\Delta}}}, a_1, a_g) \end{aligned}$$

Algorithm 1: SUB-SAMPLE-MFQ: Learning (if $|\mathcal{Z}_l|^{k-1} \leq k^{|\mathcal{Z}_l|}$)

Require: A multi-agent system. Parameter T for the number of iterations in the initial value iteration step. Sampling parameters $k \in [n]$ and $m \in \mathbb{N}$. Discount parameter $\gamma \in (0,1)$. Oracle $\mathcal{O}$ to sample $s'_g \sim P_g(\cdot|s_g, a_g)$ and $s'_i \sim P_l(\cdot|s_i, s_g, a_i)$ for all $i \in [n]$.

Uniformly sample $\Delta \subseteq [n]$ such that $|\Delta| = k$.
For $(s_g, s_\Delta, a_g, a_\Delta) \in \mathcal{S}_g \times \mathcal{S}_l^k \times \mathcal{A}_g \times \mathcal{A}_l^k$, initialize $\widehat{Q}^0_{k,m}(s_g, s_\Delta, a_g, a_\Delta) = 0$.
for $t = 1$ to T **do**
  for $(s_g, s_\Delta, a_g, a_\Delta) \in \mathcal{S}_g \times \mathcal{S}_l^k \times \mathcal{A}_g \times \mathcal{A}_l^k$ **do**
    $\widehat{Q}^{t+1}_{k,m}(s_g, s_\Delta, a_g, a_\Delta)$
      $= \widehat{\mathcal{T}}_{k,m} \widehat{Q}^t_{k,m}(s_g, s_\Delta, a_g, a_\Delta)$
  end for
end for
Return $\widehat{Q}^T_{k,m}$.
For all $s_g \in \mathcal{S}_g$ and $s_\Delta \in \mathcal{S}_l^k$, define the greedy argmax policy by $\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_\Delta)$ such that $\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_\Delta) = \arg\max_{a_g \in \mathcal{A}_g, a_\Delta \in \mathcal{A}_l^k} \widehat{Q}^T_{k,m}(s_g, s_\Delta, a_g, a_\Delta)$.

algorithm 3 (Online implementation). Here, algorithm 3 (SUBSAMPLE-MFQ: Execution) randomly samples $\Delta \sim \mathcal{U}(\binom{[n]}{k})$ at each time step and uses action $a_g \sim \widehat{\pi}^{\text{est}}_{k,m}(s_g, F_{s_\Delta})$ to get reward $r(s, a_g)$. This procedure of first sampling Δ and then applying $\widehat{\pi}^{\text{est}}_{k,m}$ is denoted by a stochastic policy $\pi^{\text{est}}_{k,m}(a|s) = [\pi^{\text{est}}_{k,m}(a_g|s), \pi^{\text{est}}_{k,m}(a_l|s)]$, where $\pi^{\text{est}}_{k,m}(a_g|s)$ is the global agent's action distribution and $\pi^{\text{est}}_{k,m}(a_l|s)$ is the local agent's action distribution:

$$\pi^{\text{est}}_{k,m}(a_g|s) = \frac{1}{\binom{n}{k}} \sum_{\Delta \in \binom{[n]}{k}} \mathbf{1}(\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_\Delta) = a) \tag{10}$$

$$\pi^{\text{est}}_{k,m}(a_i|s) = \frac{1}{\binom{n-1}{k-1}} \sum_{\widetilde{\Delta} \in \binom{[n]\setminus i}{k-1}} \mathbf{1}(\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_i, F_{s_{\widetilde{\Delta}}}) = a_i). \tag{11}$$

Then, each agent transitions to their next state based on eq. (1).

We define the greedy deterministic policy for the k-local-agent subsystem by: $\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_\Delta)$ by:

$$\begin{aligned}\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_\Delta) &:= \arg\max_{a^*_g, a^*_\Delta} \widehat{Q}^T_{k,m}(s_g, s_\Delta, a^*_g, a^*_\Delta) \\ &:= ([\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_\Delta)]_g, [\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_i, s_{\Delta\setminus i})]_l),\end{aligned}$$

where $[\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_\Delta)]_g$ reads the maximizer a^*_g, and $[\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_i, s_{\Delta\setminus i})]_l$ reads the maximizer a^*_i. Then, for the n-agent system, the global agent samples local agents Δ uniformly from $\binom{[n]}{k}$ to derive action $a_g(t) = [\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_\Delta)]_g$, and allows agent i to choose agents Δ_i uniformly from $\binom{[n]\setminus i}{k-1}$ to derive action $a_i(t) = [\widehat{\pi}^{\text{est}}_{k,m}(s_g, (s_{\Delta_i}, s_i)]_l$.

Algorithm 2: SUBSAMPLE-MFQ: Learning (if $|\mathcal{Z}_l|^{k-1} > k^{|\mathcal{Z}_l|}$)

Require: A multi-agent system. Parameter T for the number of iterations in the initial value iteration step. Sampling parameters $k \in [n]$ and $m \in \mathbb{N}$. Discount parameter $\gamma \in (0,1)$. Oracle $\mathcal{O}$ to sample $s'_g \sim P_g(\cdot|s_g, a_g)$ and $s_i \sim P_l(\cdot|s_i, s_g, a_i)$ for all $i \in [n]$.

Set $\widetilde{\Delta} = \{2, \ldots, k\}$.
Set $\mu_{k-1}(\mathcal{Z}_l) = \{\frac{b}{k-1} : b \in \{0, 1, \ldots, k-1\}\}^{|\mathcal{S}_l| \times |\mathcal{A}_l|}$.
Set $\widehat{Q}^0_{k,m}(s_g, s_1, F_{z_{\widetilde{\Delta}}}, a_1, a_g) = 0$, for $(s_g, s_1, F_{z_{\widetilde{\Delta}}}, a_1, a_g) \in \mathcal{S}_g \times \mathcal{S}_l \times \mu_{k-1}(\mathcal{Z}_l) \times \mathcal{A}_l \times \mathcal{A}_g$.
for $t = 1$ to T **do**
  for $(s_g, s_1, F_{z_{\widetilde{\Delta}}}, a_1, a_g) \in \mathcal{S}_g \times \mathcal{S}_l \times \mu_{k-1}(\mathcal{Z}_l) \times \mathcal{A}_l \times \mathcal{A}_g$ **do**
    $\widehat{Q}^{t+1}_{k,m}(s_g, s_1, F_{z_{\widetilde{\Delta}}}, a_1, a_g)$
      $= \widehat{\mathcal{T}}_{k,m} \widehat{Q}^t_{k,m}(s_g, s_1, F_{z_{\widetilde{\Delta}}}, a_1, a_g)$
  end for
end for
$\forall (s_g, s_i, F_{s_{\widetilde{\Delta}}}) \in \mathcal{S}_g \times \mathcal{S}_l \times \mu_{k-1}(\mathcal{S}_l)$, let

$$\widehat{\pi}^{\text{est}}_{k,m}(s_g, s_i, F_{s_{\widetilde{\Delta}}}) := \arg\max_{a_g \in \mathcal{A}_g, a_i \in \mathcal{A}_l, F_{a_{\widetilde{\Delta}}} \in \mu_{k-1}(\mathcal{A}_l)} \widehat{Q}^T_{k,m}(s_g, s_1, F_{z_{\widetilde{\Delta}}}, a_1, a_g)$$

Theoretical Guarantee

This subsection shows that the value of the expected discounted cumulative reward produced by $\pi^{\text{est}}_{k,m}$ is approximately optimal, where the optimality gap decays as $k \to n$ and m becomes large.

Bellman noise. We introduce the notion of Bellman noise, which is used in the main theorem. Consider $\widehat{\mathcal{T}}_{k,m}$. Clearly, it is an unbiased estimator of the generalized adapted Bellman operator $\widehat{\mathcal{T}}_k$,

$$\begin{aligned}\widehat{\mathcal{T}}_k \widehat{Q}_k(s_g, s_\Delta, a_g, a_\Delta) &= r_\Delta(s, a) \\ &+ \gamma \mathbb{E}_{\substack{s'_g \sim P_g(\cdot|s_g, a_g), \\ s'_i \sim P_l(\cdot|s_i, s_g, a_i), \\ \forall i \in \Delta}} \max_{\substack{a'_g \in \mathcal{A}_g, \\ a'_\Delta \in \mathcal{A}_l^k}} \widehat{Q}_k(s'_g, s'_\Delta, a'_g, a'_\Delta).\end{aligned} \tag{12}$$

For all $(s_g, s_\Delta, a_g, a_\Delta) \in \mathcal{S}_g \times \mathcal{S}_l^k \times \mathcal{A}_g \times \mathcal{A}_l^k$, set $\widehat{Q}^0_k(s_g, s_\Delta, a_g, a_\Delta) = 0$. For $t \in \mathbb{N}$, let $\widehat{Q}^{t+1}_k = \widehat{\mathcal{T}}_k \widehat{Q}^t_k$, where $\widehat{\mathcal{T}}_k$ is defined for $k \leq n$ in eq. (12). Then, $\widehat{\mathcal{T}}_k$ is also a γ-contraction with fixed-point $\widehat{Q}^*_k$. By the law of large numbers, $\lim_{m\to\infty} \widehat{\mathcal{T}}_{k,m} = \widehat{\mathcal{T}}_k$ and $\|\widehat{Q}^{\text{est}}_{k,m} - \widehat{Q}^*_k\|_\infty \to 0$ as $m \to \infty$. For finite m, $\epsilon_{k,m} := \|\widehat{Q}^{\text{est}}_{k,m} - \widehat{Q}^*_k\|_\infty$ is the well-studied Bellman noise.

Lemma 0.9 By the Chernoff bound, for $k \in [n]$ and $m \in \mathbb{N}$, where m is the number of samples in eq. (9), $\|\widehat{Q}^{\text{est}}_{k,m} - \widehat{Q}^*_k\|_\infty \leq \epsilon_{k,m} \leq \widetilde{O}(1/\sqrt{k})$, when $m = m^* = \frac{2|\mathcal{S}_g||\mathcal{A}_g||\mathcal{S}_l||\mathcal{A}_l|k^{2.5+|\mathcal{S}_l||\mathcal{A}_l|}}{(1-\gamma)^5} \log(|\mathcal{S}_g||\mathcal{A}_g||\mathcal{A}_l||\mathcal{S}_l|) \log\left(\frac{1}{(1-\gamma)^2}\right)$.

Algorithm 3: SUB−SAMPLE−MFQ: Execution

Require: A multi-agent system as described in **??** . Parameter T' for the number of iterations for the decision-making sequence. Hyperparameter $k \in [n]$. Discount parameter γ. Policy $\widehat{\pi}^{\text{est}}_{k,m}(s_g, F_{s_\Delta})$.

Sample $(s_g(0), s_{[n]}(0)) \sim s_0$, where s_0 is a distribution on the initial global state $(s_g, s_{[n]})$
Initialize the total reward $R_0 = 0$.
Policy $\pi^{\text{est}}_k(s)$ is defined as follows:
for $t = 0$ to T' **do**
  Choose Δ uniformly at random from $\binom{[n]}{k}$ and let $a_g(t) = [\widehat{\pi}^{\text{est}}_{k,m}(s_g(t), s_\Delta(t))]_g$.
  for $i = 1$ to n **do**
    Choose Δ_i uniformly at random from $\binom{[n]\setminus i}{k-1}$ and let $a_i(t) = [\widehat{\pi}^{\text{est}}_{k,m}(s_g(t), s_i(t), s_{\Delta_i}(t))]_l$.
  end for
  Let $s_g(t+1) \sim P_g(\cdot|s_g(t), a_g(t))$.
  Let $s_i(t+1) \sim P_l(\cdot|s_i(t), s_g(t), a_i(t))$, for all $i \in [n]$.
  $R_{t+1} = R_t + \gamma^t \cdot r(s, a)$
end for

We defer the proof of this lemma to Appendix F.1 in the supplementary material.

Let $\pi^{\text{est}}_k \coloneqq \widetilde{\pi}^{\text{est}}_{k,m^*}$. We next compare the difference in the performance of π^* and π^{est}_k, we define the value function of a policy π by V^π:

Definition 0.10 *The value function $V^\pi : \mathcal{S} \to \mathbb{R}$ of a given policy π, for $\mathcal{S} := \mathcal{S}_g \times \mathcal{S}_l^n$ is:*

$$V^\pi(s) = \mathbb{E}_{a(t)\sim\pi(\cdot|s(t))}\left[\sum_{t=0}^{\infty} \gamma^t r(s(t), a(t)) \middle| s(0) = s\right]. \tag{13}$$

Intuitively, the value function $V^\pi(s)$ is the expected discounted cumulative reward when starting from state s and applying actions from the policy π across an infinite horizon. With the above preparations, we are primed to present our main result: a decaying bound on the optimality gap for our learned policy π^{est}_k.

Theorem 0.11 Let $\widetilde{\pi}_k$ denote the learned policy from SUBSAMPLE−MFQ. Then, $\forall s \in \mathcal{S}$:

$$\begin{aligned} &V^{\pi^*}(s_0) - V^{\pi^{\text{est}}_k}(s_0) \\ &\leq \frac{\widetilde{r}}{(1-\gamma)^2}\sqrt{\frac{n-k+1}{2nk}}\sqrt{\ln\frac{40\widetilde{r}|\mathcal{S}_l||\mathcal{A}_l||\mathcal{A}_g|k^{|\mathcal{A}_l|+\frac{1}{2}}}{(1-\gamma)^2}} + \frac{3}{\sqrt{k}} \\ &\leq \widetilde{O}(1/\sqrt{k}) \end{aligned}$$

We provide a proof sketch for the theorem in Appendix C of the supplementary material, and defer its proof to Appendix F.

Discussion 0.12 Between algorithms 1 and 2, the asymptotic sample complexity to learn $\widehat{\pi}^{\text{est}}_k$ for a fixed k is $\min\{O(|\mathcal{Z}_l|^k), O(k^{|\mathcal{Z}_l|})\}$. By theorem 0.11, as $k \to n$, the optimality gap decays, revealing a fundamental trade-off in the choice of k: increasing k improves the performance of the policy, but increases the size of the Q-function. We explore this trade-off further in our experiments. For $k = O(\log n)$, the runtime is $\min\{O(n^{\log|\mathcal{Z}_l|}), O((\log n)^{|\mathcal{Z}_l|})\}$. This is an exponential speedup on the complexity from mean-field value iteration (from $\text{poly}(n)$ to $\text{poly}(\log n)$), as well as over traditional value-iteration (from $\exp(n)$ to $\text{poly}(n)$). Further, the optimality gap decays to 0 at the rate of $O(1/\sqrt{\log n})$.

Appendix G in the supplementary material extends the theorem to stochastic reward distributions.

Theorem 0.13 Suppose we are given two families of distributions on the reward functions, $r_g(s_g, a_g) \sim \{\mathcal{G}_{s_g,a_g}\}_{s_g,a_g\in\mathcal{S}_g\times\mathcal{A}_g}$ and $r_l(s_i, s_g, a_i) \sim \{\mathcal{L}_{s_i,s_g,a_i}\}_{s_i,s_g,a_i\in\mathcal{S}_l\times\mathcal{S}_g\times\mathcal{A}_l}$. Then, under standard assumptions of boundedness of the support of $\mathcal{G}_{s_g,a_g}$ and $\mathcal{L}_{s_i,s_g,a_i}$, SUBSAMPLE−MFQ learns a stochastic policy π^{est}_k satisfying

$$\Pr\left[V^{\pi^*}(s_0) - V^{\pi^{\text{est}}_k}(s_0) \leq \widetilde{O}\left(\frac{1}{\sqrt{k}}\right)\right] \geq 1 - \frac{1}{100\sqrt{k}}.$$

In the non-tabular setting with infinite state/action spaces, one could replace the Q-learning algorithm with any arbitrary value-based RL method that learns $\widehat{Q}_k$ with function approximation (Sutton et al. 1999) such as deep Q-networks (Silver et al. 2016). Doing so raises an additional error that factors into theorem 0.11.

Definition 0.14 (Linear MDP) $\text{MDP}(S, A, \mathbb{P}, r)$ is a linear MDP with feature map $\phi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$ if there exist d unknown (signed) measures $\mu = (\mu^1, \ldots, \mu^d)$ over $\mathcal{S}$ and a vector $\theta \in \mathbb{R}^d$ such that for any $(s, a) \in \mathcal{S} \times \mathcal{A}$, we have $\mathbb{P}(\cdot|s, a) = \langle\phi(s, a), \mu(\cdot)\rangle$ and $r(s, a) = \langle\phi(s, a), \theta\rangle$.

Suppose the system is a linear MDP, where $\mathcal{S}_g$ and $\mathcal{S}_l$ are infinite compact sets. By a reduction from (Ren et al. 2024) and using function approximation to learn the spectral features ϕ_k for $\widehat{Q}_k$, we derive a performance guarantee for the learned policy π^{est}_k, where the optimality gap decays with k.

Theorem 0.15 When π^{est}_k is derived from the spectral features ϕ_k learned in $\widehat{Q}_k$, and M is the number of samples used in the linear function approximation, *let E be the event that*

$$\begin{aligned} V^{\pi^*}(s_0) - V^{\pi^{\text{est}}_k}(s_0) \leq \widetilde{O}\Bigg(&\frac{1}{\sqrt{k}} + \log\left(2k^2\right)\frac{\|\phi_k\|^5}{\sqrt{M}} \\ &+ \frac{2}{\sqrt{k}} \cdot \frac{\gamma\widetilde{r}}{1-\gamma}\|\bar{\phi}_k\|\Bigg). \end{aligned}$$

Then,

$$\Pr[E] \geq \left(1 - \frac{1}{100\sqrt{k}}\right) \cdot \left(1 - \frac{2}{\sqrt{k}}\right).$$

We defer the proof of the theorem to Appendix H in the supplementary material.

Remark 0.16 If $k = O(\log n)$, SUBSAMPLE-MFQ can handle $|\mathcal{E}| = O(\log n / \log\log n)$ different types of local agents, since the run-time of the learning algorithm becomes $\mathrm{poly}(n)$. This additionally supersedes the previous-best heterogeneity capacity from (Mondal et al. 2022), which only handles constant $|\mathcal{E}|$.

Remark 0.17 Our algorithm and policy also contributes to the growing literature on the centralized-training-decentralized-execution paradigm (Zhou et al. 2023; Wang, Ye, and Lu 2023), and the literature on exogenous MDPs, wherein our algorithm has an advantage in that the agents do not all have to "see" each other during the online (execution) stage, and hence contains a partially observable setting.

Conclusion and Future Works

This work develops subsampling for mean field MARL in a cooperative system with a global decision-making agent and n homogeneous local agents. We propose SUBSAMPLE-MFQ which learns each agent's best response to the mean effect from a sample of its neighbors, allowing an exponential reduction on the sample complexity of approximating a solution to the MDP. We provide a theoretical analysis on the optimality gap of the learned policy, showing that the learned policy converges to the optimal policy with the number of agents k sampled at the rate $\widetilde{O}(1/\sqrt{k})$ validate our theoretical results through numerical experiments. We further extend this result to the non-tabular setting with infinite state and action spaces.

We recognize several future directions. Firstly, this model studies a 'star-network' setting to model a single source of density. It would be fascinating to extend this subsampling framework to general networks. We believe expander-graph decompositions (Anand and Umans 2023; Reingold 2008) are amenable for this. A second direction would be to find connections between our sub-sampling method to algorithms in federated learning, where the rewards can be stochastic. A third direction of this work would be to consider the setting of truly heterogeneous local agents. Finally, it would be exciting to generalize this work to the online setting without a generative oracle: we conjecture that tools from recent works on stochastic approximation (Chen and Theja Maguluri 2022) and no-regret RL (Jin et al. 2021) might be valuable.

Impact Statement

This paper contributes to the theoretical foundations of multi-agent reinforcement learning, with the goal of developing mean-field tools that can apply to the control of networked systems. The work can potentially lead to RL-based algorithms for the adaptive control of cyber-physical systems, such as the power grid, smart traffic systems, and other smart infrastructure systems. While the subsampling approach we describe is promising, it is limited by its assumptions. Furthermore, any applications of the proposed algorithm in its current form should be considered cautiously since the analysis here focuses on efficiency and optimality, and does not consider the issue of fairness.

Acknowledgements

We gratefully acknowledge insightful discussions with Siva Theja Maguluri, Sarah Liaw, Yiheng Lin, and Yi Wu.

References

Anand, E.; and Qu, G. 2024. Efficient Reinforcement Learning for Global Decision Making in the Presence of Local Agents at Scale. *arXiV*.

Anand, E.; and Umans, C. 2023. Pseudorandomness of the Sticky Random Walk. *Caltech Undergraduate Thesis*.

Banach, S. 1922. Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales. *Fundamenta Mathematicae*, 3(1): 133–181.

Bertsekas, D. P.; and Tsitsiklis, J. N. 1996. *Neuro-Dynamic Programming*. Athena Scientific, 1st edition. ISBN 1886529108.

Blondel, V. D.; and Tsitsiklis, J. N. 2000. A Survey of Computational Complexity Results in Systems and Control. *Automatica*, 36(9): 1249–1274.

Chen, Z.; and Theja Maguluri, S. 2022. Sample Complexity of Policy-Based Methods under Off-Policy Sampling and Linear Function Approximation. In Camps-Valls, G.; Ruiz, F. J. R.; and Valera, I., eds., *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, 11195–11214. PMLR.

Cui, K.; Fabian, C.; and Koeppl, H. 2023. Multi-Agent Reinforcement Learning via Mean Field Control: Common Noise, Major Agents and Approximation Properties. arXiv:2303.10665.

Cui, K.; and Koeppl, H. 2022. Learning Graphon Mean Field Games and Approximate Nash Equilibria. In *International Conference on Learning Representations*.

DeWeese, A.; and Qu, G. 2024. Locally Interdependent Multi-Agent MDP: Theoretical Framework for Decentralized Agents with Dynamic Dependencies. In *Forty-first International Conference on Machine Learning*.

Dietterich, T. G.; Trimponias, G.; and Chen, Z. 2018. Discovering and Removing Exogenous State Variables and Rewards for Reinforcement Learning. arXiv:1806.01584.

Foster, D. J.; Rakhlin, A.; Sekhari, A.; and Sridharan, K. 2022. On the Complexity of Adversarial Decision Making. In Oh, A. H.; Agarwal, A.; Belgrave, D.; and Cho, K., eds., *Advances in Neural Information Processing Systems*.

Gamarnik, D.; Goldberg, D.; and Weber, T. 2009. Correlation Decay in Random Decision Networks. arXiv:0912.0338.

Golowich, N.; and Moitra, A. 2024. The Role of Inherent Bellman Error in Offline Reinforcement Learning with Linear Function Approximation. arXiv:2406.11686.

Gu, H.; Guo, X.; Wei, X.; and Xu, R. 2021. Mean-Field Controls with Q-Learning for Cooperative MARL: Convergence and Complexity Analysis. *SIAM Journal on Mathematics of Data Science*, 3(4): 1168–1196.

Gu, H.; Guo, X.; Wei, X.; and Xu, R. 2022a. Dynamic Programming Principles for Mean-Field Controls with Learning. arXiv:1911.07314.

Gu, H.; Guo, X.; Wei, X.; and Xu, R. 2022b. Mean-Field Multi-Agent Reinforcement Learning: A Decentralized Network Approach. arXiv:2108.02731.

Guestrin, C.; Koller, D.; Parr, R.; and Venkataraman, S. 2003. Efficient Solution Algorithms for Factored MDPs. *J. Artif. Int. Res.*, 19(1): 399–468.

Hu, Y.; Wei, X.; Yan, J.; and Zhang, H. 2023. Graphon Mean-Field Control for Cooperative Multi-Agent Reinforcement Learning. *Journal of the Franklin Institute*, 360(18): 14783–14805.

Jin, C.; Liu, Q.; and Miryoosefi, S. 2021. Bellman Eluder Dimension: New Rich Classes of RL Problems, and Sample-Efficient Algorithms. arXiv:2102.00815.

Jin, C.; Liu, Q.; Wang, Y.; and Yu, T. 2021. V-Learning – A Simple, Efficient, Decentralized Algorithm for Multiagent RL. arXiv:2110.14555.

Jin, C.; Yang, Z.; Wang, Z.; and Jordan, M. I. 2020. Provably efficient reinforcement learning with linear function approximation. In Abernethy, J.; and Agarwal, S., eds., *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, 2137–2143. PMLR.

Jin, J.; Song, C.; Li, H.; Gai, K.; Wang, J.; and Zhang, W. 2018. Real-Time Bidding with Multi-Agent Reinforcement Learning in Display Advertising. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, 2193–2201. New York, NY, USA: Association for Computing Machinery. ISBN 9781450360142.

Kim, S.-J.; and Giannakis, G. B. 2017. An Online Convex Optimization Approach to Real-Time Energy Pricing for Demand Response. *IEEE Transactions on Smart Grid*, 8(6): 2784–2793.

Kiran, B. R.; Sobh, I.; Talpaert, V.; Mannion, P.; Sallab, A. A. A.; Yogamani, S.; and Pérez, P. 2022. Deep Reinforcement Learning for Autonomous Driving: A Survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(6): 4909–4926.

Kober, J.; Bagnell, J. A.; and Peters, J. 2013. Reinforcement Learning in Robotics: A Survey. *The International Journal of Robotics Research*, 32(11): 1238–1274.

Lasry, J.-M.; and Lions, P.-L. 2007. Mean Field Games. *Japanese Journal of Mathematics*, 2(1): 229–260.

Lauer, M.; and Riedmiller, M. A. 2000. An Algorithm for Distributed Reinforcement Learning in Cooperative Multi-Agent Systems. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, 535–542. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. ISBN 1558607072.

Li, M.; Qin, Z.; Jiao, Y.; Yang, Y.; Wang, J.; Wang, C.; Wu, G.; and Ye, J. 2019. Efficient Ridesharing Order Dispatching with Mean Field Multi-Agent Reinforcement Learning. In *The World Wide Web Conference*, WWW '19, 983–994. New York, NY, USA: Association for Computing Machinery. ISBN 9781450366748.

Lin, Y.; Preiss, J. A.; Anand, E. T.; Li, Y.; Yue, Y.; and Wierman, A. 2023. Online Adaptive Policy Selection in Time-Varying Systems: No-Regret via Contractive Perturbations. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Lin, Y.; Preiss, J. A.; Xie, F.; Anand, E.; Chung, S.-J.; Yue, Y.; and Wierman, A. 2024. Online Policy Optimization in Unknown Nonlinear Systems. In Agrawal, S.; and Roth, A., eds., *Proceedings of Thirty Seventh Conference on Learning Theory*, volume 247 of *Proceedings of Machine Learning Research*, 3475–3522. PMLR.

Lin, Y.; Qu, G.; Huang, L.; and Wierman, A. 2020. Distributed Reinforcement Learning in Multi-Agent Networked Systems. *CoRR*, abs/2006.06555.

Littman, M. L. 1994. Markov Games as a Framework for Multi-Agent Reinforcement Learning. In *Machine learning proceedings*, Elsevier, 157–163.

Min, Y.; He, J.; Wang, T.; and Gu, Q. 2023. Cooperative Multi-Agent Reinforcement Learning: Asynchronous Communication and Linear Function Approximation. In Krause, A.; Brunskill, E.; Cho, K.; Engelhardt, B.; Sabato, S.; and Scarlett, J., eds., *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 24785–24811. PMLR.

Mondal, W. U.; Agarwal, M.; Aggarwal, V.; and Ukkusuri, S. V. 2022. On the Approximation of Cooperative Heterogeneous Multi-Agent Reinforcement Learning (MARL) Using Mean Field Control (MFC). *Journal of Machine Learning Research*, 23(1).

Papadimitriou, C. H.; and Tsitsiklis, J. N. 1999. The Complexity of Optimal Queuing Network Control. *Mathematics of Operations Research*, 24(2): 293–305.

Preiss, J. A.; Honig, W.; Sukhatme, G. S.; and Ayanian, N. 2017. Crazyswarm: A large nano-quadcopter swarm. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 3299–3304.

Qu, G.; Lin, Y.; Wierman, A.; and Li, N. 2020. Scalable Multi-Agent Reinforcement Learning for Networked Systems with Average Reward. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781713829546.

Reingold, O. 2008. Undirected Connectivity in Log-Space. *J. ACM*, 55(4).

Ren, Z.; Runyu; Zhang; Dai, B.; and Li, N. 2024. Scalable spectral representations for network multiagent control. arXiv:2410.17221.

Sayin, M. O.; Zhang, K.; Leslie, D. S.; Basar, T.; and Ozdaglar, A. E. 2021. Decentralized Q-learning in Zero-sum Markov Games. In Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*.

Shapley, L. S. 1953. A Value for n-Person Games. In *Contributions to the Theory of Games*.

Silver, D.; Huang, A.; Maddison, C. J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; Dieleman, S.; Grewe, D.; Nham, J.; Kalchbrenner, N.; Sutskever, I.; Lillicrap, T.; Leach, M.; Kavukcuoglu, K.; Graepel, T.; and Hassabis, D. 2016. Mastering the Game of Go with Deep Neural Networks and Tree Search. *Nature*, 529(7587): 484–489.

Sutton, R. S.; McAllester, D.; Singh, S.; and Mansour, Y. 1999. Policy Gradient Methods for Reinforcement Learning with Function Approximation. In Solla, S.; Leen, T.; and Müller, K., eds., *Advances in Neural Information Processing Systems*, volume 12. MIT Press.

Tan, M. 1997. *Multi-agent reinforcement learning: independent vs. cooperative agents*, 487–494. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. ISBN 1558604952.

Wang, J.; Ye, D.; and Lu, Z. 2023. More Centralized Training, Still Decentralized Execution: Multi-Agent Conditional Policy Factorization. arXiv:2209.12681.

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

Yang, Y.; Luo, R.; Li, M.; Zhou, M.; Zhang, W.; and Wang, J. 2018. Mean Field Multi-Agent Reinforcement Learning. In Dy, J.; and Krause, A., eds., *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 5571–5580. PMLR.

Zhang, K.; Yang, Z.; and Başar, T. 2021. Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms. arXiv:1911.10635.

Zhou, Y.; Liu, S.; Qing, Y.; Chen, K.; Zheng, T.; Huang, Y.; Song, J.; and Song, M. 2023. Is Centralized Training with Decentralized Execution Framework Centralized Enough for MARL? arXiv:2305.17352.