Individual-Level Inverse Reinforcement Learning for Mean Field Games

Yang Chen
The University of Auckland
Auckland, New Zealand
yang.chen@auckland.ac.nz

Libo Zhang
The University of Auckland
Auckland, New Zealand
lzha797@aucklanduni.ac.nz

Jiamou Liu
The University of Auckland
Auckland, New Zealand
jiamou.liu@auckland.ac.nz

Shuyue Hu
National University of Singapore
Singapore
dcshus@nus.edu.sg

ABSTRACT

The recent mean field game (MFG) formalism has enabled the application of inverse reinforcement learning (IRL) methods in large-scale multi-agent systems, with the goal of inferring reward signals that can explain demonstrated behaviours of large populations. The existing IRL methods for MFGs are built upon reducing an MFG to a Markov decision process (MDP) defined on the collective behaviours and average rewards of the population. However, this paper reveals that the reduction from MFG to MDP holds only for the fully cooperative setting. This limitation invalidates existing IRL methods on MFGs with non-cooperative environments. To measure more general behaviours in large populations, we study the use of individual behaviours to infer ground-truth reward functions for MFGs. We propose *Mean Field IRL* (MFIRL), the first dedicated IRL framework for MFGs that can handle both cooperative and non-cooperative environments. Based on this theoretically justified framework, we develop a practical algorithm effective for MFGs with unknown dynamics. We evaluate MFIRL on both cooperative and mixed cooperative-competitive scenarios with many agents. Results demonstrate that MFIRL excels in reward recovery, sample efficiency and robustness in the face of changing dynamics.

KEYWORDS

Inverse Reinforcement Learning; Mean Field Games; Individual Level; Mean Field Nash Equilibrium

1 INTRODUCTION

Inverse reinforcement learning (IRL) is concerned with one or multiple agents operating in an environment that is agnostic towards reward signals. It provides a powerful solution to learn behavioural models by inferring reward functions from demonstrations. The majority of its successful applications, however, deal with systems with a handful of agents [7, 27]. Yet, many scenarios involve a much

larger population, such as traffic networks with millions of vehicles [6], online games with massive players [26], and online businesses with a large customer body [2]. Applying IRL methods to such large-scale systems is intractable due to the exponential growth of joint state-action spaces and agent interactions.

A promising concept to achieve tractability for modelling large-scale multi-agent systems (MAS) is the *mean field game* (MFG) [25, 30] that uses mean field theory to simplify interactions among a large number of agents. An MFG views agents as a homogeneous population, i.e., they are identical, indistinguishable, and interchangeable [22]. It thus uses a single entity, termed *mean field*, to denote the statistical information of the overall population, rather than modelling each agent individually. The interactions among agents are therefore reduced to those between a single representative agent and the overall population. This reduction to a dual-view interplay enables characterising the optimal behaviours in large-scale MAS using *mean field Nash equilibrium* (MFNE), where each agent's policy is a best response to the mean field and the mean field is in turn consistent with the policy. MFGs have enabled applications in many fields such as economics [19, 28], finance [9, 12] and crowd motion [8, 29]. To measure behaviours in large populations, it is thus promising to study IRL for MFGs, which aims to uncover reward signals behind demonstrated MFNE behaviours.

A recent research [55] studies IRL in MFGs, which shows that an MFG can be reduced to a Markov decision process (MDP). It thus extends IRL to MFGs via applying existing single-agent IRL methods to this MDP. Since this MDP describes the population's *collective behaviours* driven by the *societal* reward (i.e., the average reward of the population), we henceforth call this method *population-level IRL*. However, we reveal in this paper that the reduction from MFG to MDP holds only for the *fully cooperative* setting, i.e., all agents share the same societal reward. Consequently, population-level IRL is prone to biased reward inferences in non-cooperative (competitive or mixed cooperative-competitive) environments. In contrast to the societal reward, the reward of each *individual* can capture each agent's real intention, regardless of whether the environment is cooperative or not. To model and predict both cooperative and non-cooperative behaviours in large-scale MAS, it is thus important to consider inferring rewards of individual agents in MFGs.

In this paper, considering the preliminary investigation of IRL in MFGs and limitations of population-level IRL, we study IRL for MFGs at the individual level. We make the following contributions

to the field: (1) We begin with a theoretical justification for the fact that the reduction from MFG to MDP holds only for the fully cooperative setting. The exposure of this limitation reveals the restricted suitability of the existing MFG-MDP reduction-based IRL and reinforcement learning (RL) methods for general MFGs. (2) Towards measuring both cooperative and non-cooperative interactions in MFGs, we then propose and formalise the problem of *individual-level IRL* for MFGs. This new problem formulation targets recovering the individual reward function from demonstrated individual behaviours, which is not amenable to population-level IRL. (3) To solve this problem, we put forward a novel and dedicated IRL framework for MFGs, called *Mean Field IRL* (MFIRL). We show that MFIRL can recover a suitable reward function under standard assumptions, no matter whether the environment is cooperative or not. (4) Based on this new framework, we develop a practical algorithm effective for MFGs with unknown dynamics. (5) We empirically evaluate MFIRL on five numerical models and simulated battle games, both with cooperative and non-cooperative environments. In non-cooperative scenarios, MFIRL outperforms population-level IRL in terms of reward recovery and robustness against changing dynamics. Moreover, in cooperative scenarios, a notable advantage of MFIRL is that it requires fewer samples to achieve comparable performance.

2 RELATED WORK

RL for MFGs. MFGs were independently proposed by Lasry and Lions [30] and Huang et al. [25] in the continuous-time setting. Mathematically, the system dynamics is governed by two stochastic differential equations: one models the backward dynamics of a representative agent's value functions; the other models the forward dynamics of the mean field. Discrete-time MFG models, adopted in this paper, were then proposed in [18]. RL in MFGs has become a burgeoning research field recently. Yang et al. [56] and Subramanian et al. [51] used mean field theory to approximate joint actions in large-population stochastic games to approximate Nash equilibria. Mguni et al. [37] proposed a decentralised RL method for MFGs. Guo et al. [20] presented a Q-learning-based algorithm for computing *stationary MFNE*. Subramanian and Mahajan [50] used RL to compute *local MFNE* (a relaxed version). While, all these works presuppose the presence of reward functions. Our work takes a complementary view that the reward function is difficult to specify, and hence the necessity for IRL for MFGs.

IRL for MDPs. The problem of IRL was first studied by Ng and Russell [38] on MDPs. Existing IRL methods typically fall into the following categories: (1) *margin optimisation* based methods [1, 40, 42, 52] that find a reward by creating a margin between the expert policy and any other policy in terms of rewards; (2) *Bayesian IRL* methods [13, 32, 35, 41] that use demonstrations to facilitate a Bayesian update of a prior distribution over candidate reward functions; (3) *maximum entropy IRL* methods [16, 17, 59, 60] that use a probabilistic framework to find the policy maximising the entropy of expert demonstrations. As explained above, although the reduction from MFG to MDP enables applying these MDP-based IRL methods to MFGs, they can only deal with fully cooperative environments. In this paper, we formalise the problem of individual-level IRL for MFGs using the idea of margin optimisation, where

the expert policy and any other policy is separated by a margin in terms of rewards, based on the expert demonstrated mean field.

IRL for MAS. Recently, some research has explored IRL in the multi-agent setting. Most of these works assume specific reward structures, including fully cooperative games [5, 7], fully competitive games [34], or either of the two [43, 53]. For general stochastic games, Yu et al. [57] presented MA-AIRL, a multi-agent IRL method using adversarial learning. Another line of works studied inverse dynamic games [31, 39, 44], which aims to infer the cost functions for dynamic games from a control perspective. However, all these prior methods scale poorly to a large agent number. Šošić et al. [49] proposed SwarmIRL that views a large-scale MAS as a swarm system consisting of homogeneous agents. However, it cannot handle non-stationary policies and non-linear reward functions. Our work makes no modelling assumptions on policies and reward functions.

3 PRELIMINARIES

The formalism of *mean field game* (MFG) [25, 30] offers a mathmetically tractable model for analysing large-scale multi-agent systems. It approximates the interactions among homogeneous agents by those between a representative agent and the population. In this section, we introduce the formulation of MFGs and some standard equilibrium concepts which we will build upon in our method.

3.1 Mean Field Games

Throughout the paper, we focus on MFGs with finite state-action spaces and finite time horizons [15]. First, consider an N-player game where all agents share the same local state space $\mathcal{S}$ and action space $\mathcal{A}$. A *joint state* is a tuple $(s^1, \ldots, s^N) \in \mathcal{S}^N$ where $s^i \in \mathcal{S}$ is the state of the ith agent. Taking the limit as $N \to \infty$, instead of modelling each agent individually, MFGs model a single representative agent and collapse the joint state into an empirical distribution $\mu \in \mathcal{P}(\mathcal{S})$, called a *mean field*, given by

$$\mu(s) \triangleq \lim_{N \to \infty} \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}_{\{s^i = s\}}, \tag{1}$$

where $\mathbb{1}$ denotes the indicator function (i.e., $\mathbb{1}_{\{x\}} = 1$ if x is true and 0 otherwise) and $\mathcal{P}(\mathcal{S})$ denotes the set of probability distributions over $\mathcal{S}$. The *transition function* $P \colon \mathcal{S} \times \mathcal{A} \times \mathcal{P}(\mathcal{S}) \times \mathcal{S} \to [0, 1]$ specifies how an agent's states evolve, i.e., an agent transits to the next state s_{t+1} with probability $P(s_{t+1}|s_t, a_t, \mu_t)$, depending on its current state, action, and mean field. Let $T \in \mathbb{N}^+$ denote a finite time horizon. A *mean field flow* (MF flow for short) consists of a sequence of $T+1$ mean fields $\boldsymbol{\mu} \triangleq \{\mu_t\}_{t=0}^{T}$, where the initial value μ_0 is given. The *running reward* of an agent is specified by the *reward function* $r \colon \mathcal{S} \times \mathcal{A} \times \mathcal{P}(\mathcal{S}) \to \mathbb{R}$ with the exception that the reward at the last step ($t = T$) is defined separately. Following the convention, we set it as zero [15, 55]. The agent's long-term reward is thus the sum $\sum_{t=0}^{T-1} \gamma^t r(s_t, a_t, \mu_t)$, where $\gamma \in (0, 1]$ is the *discounted factor*. To summarise, an MFG is defined as a tuple $(\mathcal{S}, \mathcal{A}, P, \mu_0, r, \gamma)$.

MFGs adopt a *time-varying stochastic policy* $\boldsymbol{\pi} \triangleq \{\pi_t\}_{t=0}^{T}$ to characterise a strategic agent, where $\pi_t \colon \mathcal{S} \to \mathcal{P}(\mathcal{A})$ is the *per-step policy* at step t, i.e., π_t directs the agent to choose action $a_t \sim \pi_t(\cdot|s_t)$. Given a policy $\boldsymbol{\pi}$, the *expected return* (cumulative rewards) of an agent during the whole course of a game while interacting

with an MF flow $\boldsymbol{\mu}$ is given by

$$J(\boldsymbol{\mu}, \boldsymbol{\pi}) \triangleq \mathbb{E} \left[\sum_{t=0}^{T-1} \gamma^t r(s_t, a_t, \mu_t) \,\Big|\, s_0 \sim \mu_0, \boldsymbol{\mu}, \boldsymbol{\pi}, P \right]. \qquad (2)$$

At the individual level, an agent seeks an optimal control in the form of a policy to maximise the expected return. Fixing an MF flow $\boldsymbol{\mu}$, a policy $\boldsymbol{\pi}$ is called a *best response* to $\boldsymbol{\mu}$ if it maximises $J(\boldsymbol{\mu}, \boldsymbol{\pi})$. We denote the set of all best-response policies to a given $\boldsymbol{\mu}$ by

$$\Psi(\boldsymbol{\mu}) \triangleq \arg\max_{\boldsymbol{\pi}} J(\boldsymbol{\mu}, \boldsymbol{\pi}). \qquad (3)$$

Since all agents are homogeneous, MFG prescribes that every agent uses the same policy. The dynamics of MF flow is thereby governed by the (discrete-time) *McKean-Vlasov* (MKV) equation [10]:

$$\mu_{t+1}(s') = \sum_{s \in \mathcal{S}} \mu_t(s) \sum_{a \in \mathcal{A}} \pi_t(a|s) \, P(s'|s, a, \mu_t). \qquad (4)$$

Denote $\Phi(\boldsymbol{\pi})$ as the MF flow fulfilling the MKV equation above given a policy $\boldsymbol{\pi}$. We say $\boldsymbol{\mu}$ is *consistent* with $\boldsymbol{\pi}$ if $\boldsymbol{\mu} = \Phi(\boldsymbol{\pi})$.

At the population level, if all agents use the same policy $\boldsymbol{\pi}$, the population state distribution (i.e., the mean field μ_t) matches each individual's state visitation distribution (i.e., $P(\cdot|s_{t-1}, a_{t-1}, \mu_{t-1})$) [20]. As a result, the cumulative *societal rewards* (population's average rewards) coincide with an individual's expected return $J(\boldsymbol{\mu}, \boldsymbol{\pi})$. Formally, let $\bar{r}(\mu, \pi)$ denote the societal reward when all agents play the same per-step policy π under the mean field μ:

$$\bar{r}(\mu, \pi) \triangleq \sum_{s \in \mathcal{S}} \mu(s) \sum_{a \in \mathcal{A}} \pi(a|s) r(s, a, \mu). \qquad (5)$$

We can express $J(\boldsymbol{\mu}, \boldsymbol{\pi})$ in terms of the societal reward as follows:

$$J(\boldsymbol{\mu}, \boldsymbol{\pi}) = \sum_{t=0}^{T-1} \gamma^t \bar{r}(\mu_t, \pi_t) \quad \text{if} \quad \boldsymbol{\mu} = \Phi(\boldsymbol{\pi}). \qquad (6)$$

3.2 Mean Field Nash Equilibrium

When agents are strategic (non-cooperative), the *mean field Nash equilibrium* (MFNE) is adopted as the solution concept, where all agents use the same best-response policy to the MF flow. Meanwhile, the MF flow is consistent with the policy.

DEFINITION 1 (MEAN FIELD NASH EQUILIBRIUM). *A pair of MF flow and policy $(\boldsymbol{\mu}^\star, \boldsymbol{\pi}^\star)$ is called a* mean field Nash equilibrium *if*

- *Agent rationality: $\boldsymbol{\pi}^\star \in \Psi(\boldsymbol{\mu}^\star)$;*
- *Population consistency: $\boldsymbol{\mu}^\star = \Phi(\boldsymbol{\pi}^\star)$.*

An MFG admits an MFNE under the standard assumptions in game theory [14, 45]. The computation of MFNE typically involves a fixed-point iteration procedure for the MF flow. More formally, defining any mapping $\hat{\Psi} : \boldsymbol{\mu} \mapsto \boldsymbol{\pi}$ that identifies a best-response policy in $\Psi(\boldsymbol{\mu})$, we obtain the fixed-point iteration for the MF flow by alternating between $\boldsymbol{\pi} = \hat{\Psi}(\boldsymbol{\mu})$ and $\boldsymbol{\mu} = \Phi(\boldsymbol{\pi})$. The standard assumption for the uniqueness of MFNE is that the fixed-point iteration will converge to a unique MF flow. However, it does not hold in general [14], which implies the existence of multiple MFNE.

3.3 Mean Field Social Optimum

When agents are cooperative, the solution concept is the *mean field social optimum* (MFSO), which maximises cumulative societal rewards whilst satisfies the condition of population consistency.

DEFINITION 2 (MEAN FIELD SOCIAL OPTIMUM). *A pair of MF flow and policy $(\bar{\boldsymbol{\mu}}^\star, \bar{\boldsymbol{\pi}}^\star)$ is called a* mean field social optimum *if*

- *Maximum cumulative societal rewards: $J(\bar{\boldsymbol{\mu}}^\star, \bar{\boldsymbol{\pi}}^\star) \geq J(\boldsymbol{\mu}, \boldsymbol{\pi})$ for any $(\boldsymbol{\mu}, \boldsymbol{\pi})$ satisfying $\boldsymbol{\mu} = \Phi(\boldsymbol{\pi})$;*
- *Population consistency: $\bar{\boldsymbol{\mu}}^\star = \Phi(\bar{\boldsymbol{\pi}}^\star)$.*

MFSO is not equivalent to MFNE. In an MFNE, $\boldsymbol{\pi}^\star$ maximises the expected return $J(\boldsymbol{\mu}, \boldsymbol{\pi})$ given $\boldsymbol{\mu}^\star$; while an MFSO $(\bar{\boldsymbol{\mu}}^\star, \bar{\boldsymbol{\pi}}^\star)$ maximises $J(\boldsymbol{\mu}, \boldsymbol{\pi})$ among all $(\boldsymbol{\mu}, \boldsymbol{\pi})$ satisfying $\boldsymbol{\mu} = \Phi(\boldsymbol{\pi})$. In other words, an MFSO is a particular MFNE that maximises the expected return. Note that if an MFNE exists uniquely, it is also an MFSO. Unless specified otherwise, we will use the term MFNE to denote an equilibria that is not an MFSO. The process of finding an MFSO can be defined as a constrained optimisation problem:

$$\max_{\boldsymbol{\mu}, \boldsymbol{\pi}} J(\boldsymbol{\mu}, \boldsymbol{\pi}) \quad \text{subject to} \quad \boldsymbol{\mu} = \Phi(\boldsymbol{\pi}). \qquad (7)$$

In Sec. 4, we draw a connection between this optimisation problem and a particular Markov decision process (MDP).

4 REVISITING IRL FOR MFG VIA REDUCING MFG TO MDP

Suppose we have no access to the reward function $r(s, a, \mu)$ but have a set of expert demonstrations. *Inverse reinforcement learning* (IRL) aims to uncover the reward function behind these demonstrations. In this section, we revisit the IRL method in [55], which showed that an MFG can be reduced to an MDP. It thus extended IRL to MFGs by applying existing single-agent IRL methods to this MDP. This reduction is also adopted in the study of reinforcement learning (RL) for MFGs [11]. However, we show that this reduction holds only for the *fully cooperative setting*, i.e., all agents share the same societal reward. As a result, IRL methods based on this reduction would lead to biased reward inferences if demonstrations are sampled from an MFNE rather than an MFSO. The exposure of this limitation motivates our restudy on IRL for MFGs. In Sec. 5, we propose a novel IRL method for MFGs with general non-cooperative environments.

4.1 The Reduction from MFG to MDP

Shown in [11, 55], an MFG can be reduced to an MDP that describes the population's collective behaviours. The state, action and reward in this MDP corresponds to the mean field, population's collective action and societal reward in the MFG, respectively. Its dynamics coincides with the MKV equation. Formally, the MDP associated with an MFG $(\mathcal{S}, \mathcal{A}, P, \mu_0, r, \gamma)$ is constructed as follows:

- State: μ_t, i.e., the state at step t is the mean field μ_t.
- Action: π_t, i.e., the action is a per-step policy in MFG.
- Reward: $\bar{r}(\mu_t, \pi_t)$, i.e., the societal reward.
- Deterministic Transition: $\mu_{t+1} = \Phi(\mu_t, \pi_t)$ [1] fulfilling the MKV equation defined in Eq. (4).
- Stationary Policy: $\varphi : \mu \mapsto \pi$.

---

[1] Here, we slightly abuse the notation Φ to denote the next mean field induced by the current mean field and current per-step policy, according to the MKV equation.

Intuitively, one can interpret the MDP policy φ as a "central controller" who makes decisions for the overall population based on the current population state distribution. Let $\varphi^\star$ be an optimal policy for the MDP, it was claimed in [11, 55] that the MDP state-action trajectory $(\mu_0, \pi_0, \ldots, \mu_T, \pi_T)$ generated by $\varphi^\star$ constitutes an MFNE of the MFG, where $\pi_t = \varphi^\star(\mu_t)$ and $\mu_{t+1} = \Phi(\mu_t, \pi_t)$.

However, we show that, more precisely, the MDP state-action trajectory generated by $\varphi^\star$ constitutes an MFSO rather than a general MFNE. The reason lies in the construction of the MDP: the optimal MDP policy $\varphi^\star$ maximises the cumulative societal rewards $J(\mu, \pi) = \sum_{t=0}^{T-1} \gamma^t \bar{r}(\mu_t, \pi_t)$ and meanwhile, the deterministic transition enforces the condition of population consistency, thereby exactly solving the constrained optimisation problem of computing an MFSO as defined in Eq. (7). Intuitively, at the macroscopic level, the dynamics of population's collective behaviours is governed by an MDP only if agents are fully cooperative. Therefore, if an MFG is not fully cooperative or we do not assume an MFNE exists uniquely, the reduction from MFG to MDP would no longer hold.

4.2 Population-Level IRL for MFGs

The work [55] proposed to infer the societal reward function $\bar{r}$ by applying single-agent IRL methods to the MDP defined above. Since this method infers the population's societal reward, we henceforth call it *Population-Level IRL* (PLIRL). In PLIRL, we assume that the demonstrations $\mathcal{D} = \{(\mu_0^j, \pi_0^j, \ldots, \mu_T^j, \pi_T^j)\}_{j=1}^M$ are a set of M MF flow-policy trajectories whose expectation is (μ^E, π^E). The goal of PLIRL is to find a suitable $\bar{r}$ that can rationalise the expert behaviour (μ^E, π^E). We can succinctly represent PLIRL as the following constrained optimisation problem:

$$\text{PLIRL}\left(\mu^E, \pi^E\right) = \underset{\bar{r}(\mu,\pi)}{\arg\max} \left[J\left(\mu^E, \pi^E\right) - \max_{\mu,\pi} J(\mu, \pi) \right], \quad (8)$$
$$\text{subject to} \quad \mu = \Phi(\pi)$$

where $J(\mu, \pi)$ is in the form of Eq. (6). Intuitively, if (μ^E, π^E) indeed maximises the societal rewards under a feasible $\bar{r}$, then the objective would attain the maximum 0; otherwise, it is negative. In IRL literature, problems in the form of Eq. (8) are generally solved by a bilevel optimisation procedure [16, 17, 60]. In PLIRL, specifically, the upper-level task is to tune the societal reward function $\bar{r}(\mu, \pi)$ given the optimal expected return of the MDP defined above; the lower-level task is to solve an MDP based on the current reward function. In practice, $J(\mu^E, \pi^E)$ is estimated from $\mathcal{D}$.

Although PLIRL allows us to model and predict the population's collective behaviours, it can only handle demonstrations sampled from an MFSO because the reduction from the MFG to the MDP holds only for the fully cooperative setting. The problem setting of PLIRL does not necessarily align with the interest of each individual agent in MFGs because, in general, agents exhibit non-cooperative interactions. The equilibrium behind demonstrations is thereby more likely to be an MFNE rather than an MFSO. Applying PLIRL on MFNE demonstrations would thus lead to biased reward inferences, as is illustrated in Fig 1. It can also be observed from Eq. (8), where $J(\mu^E, \pi^E)$ is generally not the maximised societal rewards under an MFNE. As a result, the policy elicited from a biased reward function may not coincide with any MFNE induced by the ground-truth reward function, leading to an unsuitable behavioural model.
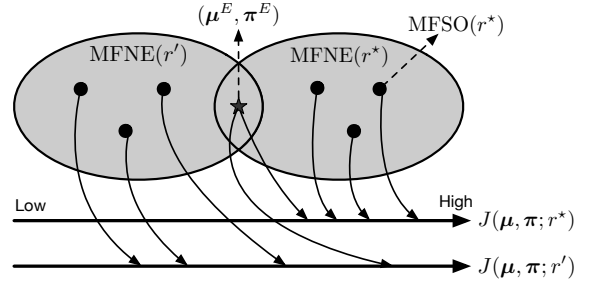

Figure 1: Illustration of the biased reward inference in population-level IRL. The shaded ellipse represents the set of all MFNE (including MFSO) induced by a specific reward function. The expert uses an MFNE (μ^E, π^E) under the ground-truth reward function $r^\star$, which is an MFSO under another reward function r'. Since PLIRL presupposes expert demonstrations are sampled from an MFSO, it tends to infer the societal reward induced by r' rather than that induced by $r^\star$.

5 INDIVIDUAL-LEVEL IRL FOR MFGs

The discussions above justify the necessity to recover the individual reward function $r(s, a, \mu)$ from demonstrations sampled from an MFNE because the individual reward is independent of the environment and thereby allows us to model and predict both cooperative and non-cooperative behaviours in large populations. In this section, we first formalise this problem as the *Individual-Level IRL* (ILIRL) for MFGs, as opposed to population-level IRL. We then propose our solution framework for ILIRL with theoretical guarantees.

5.1 Problem Formulation

Suppose we do not know the ground-truth reward function $r(s, a, \mu)$ but have a set of expert demonstrations $\mathcal{D} = \{\tau^j\}_{j=1}^M$ sampled from an *unknown* MFNE (μ^E, π^E). Each $\tau^j = s_0^j, a_0^j, \ldots, s_T^j, a_T^j$ is a state-action trajectory of an *individual* agent, which is sampled via $s_0 \sim \mu_0^E$, $s_t \sim P(\cdot|s_{t-1}, a_{t-1}, \mu_{t-1}^E)$ [2] and $a_t \sim \pi_t^E(\cdot|s_t)$. Following the convention in IRL literature [21, 48, 57], we assume that $\mathcal{D}$ provides the entire supervision signals, i.e., we cannot further communicate with the expert for additional information. ILIRL for MFG asks for a reward function $r(s, a, \mu)$, under which (μ^E, π^E) forms an MFNE.

The individual-level inference characteristics of ILIRL are embodied in two key aspects, which distinguish it from PLIRL. First, ILIRL uses demonstrated state-action trajectories sampled from individuals. In contrast, PLIRL uses demonstrated MF flows and policies sampled from the population. Second, ILIRL aims to infer the individual reward $r(s, a, \mu)$ and we can calculate societal rewards accordingly; while PLIRL aims to infer the societal reward $\bar{r}(\mu, \pi)$ that is uninformative for acquiring individual rewards.

To frame ILIRL as an optimisation problem, we desire an inverse operator ILIRL(μ^E, π^E) in analogy to PLIRL as defined in Eq. (8). The key idea of Eq. (8) is to choose a societal reward $\bar{r}(\mu, \pi)$ that creates a *margin* between the expert and every other MF flow-policy pair. Since in an MFNE the policy maximises the expected

---

[2] Since under an MFNE, the mean field matches each individual's state visitation distribution, sampling the state from a single individual via $s_t \sim P(\cdot|s_{t-1}, a_{t-1}, \mu_{t-1}^E)$ is equivalent to sampling that from multiple individuals, i.e., $s_t \sim \mu_t^E$.
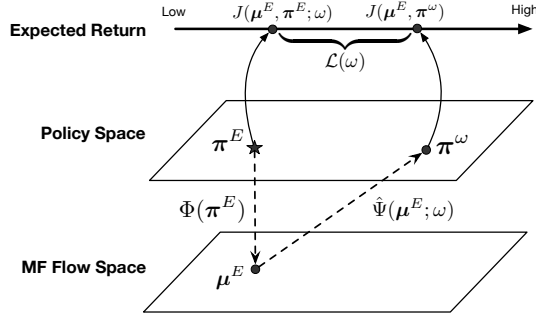
Figure 2: Illustration of the MFIRL framework. The function $\hat{\Psi}(\mu^E; \omega)$ identifies a best-response policy π^ω to μ^E under a reward function r_ω. The function $\mathcal{L}(\omega)$ measures the difference between expected returns induced by π^E and π^ω given μ^E. If we can find a parameter $\omega^\star$ such that $\mathcal{L}(\omega^\star) = 0$, then r_ω is a valid solution to ILIRL. In practice, μ^E is replaced by the empirical value $\hat{\mu}^E$ estimated from demonstrations $\mathcal{D}$.

return given the MF flow, we can interpret ILIRL as finding a reward $r(s, a, \mu)$ that creates a margin between the expert policy π^E and every other policy given the expert MF flow μ^E:

$$\text{ILIRL}\left(\mu^E, \pi^E\right) = \underset{r(s,a,\mu)}{\arg\max}\left[J\left(\mu^E, \pi^E\right) - \max_\pi J\left(\mu^E, \pi\right)\right], \quad (9)$$

where $J(\mu, \pi)$ is in the form of Eq. (2). If (μ^E, π^E) is an MFNE under a valid $r(s, a, u)$, then the objective attains the maximum 0, otherwise it is negative. Note that there may exist multiple feasible solutions to the problem. We do not intend to find all of them, as any a feasible reward function can explain expert demonstrations.

5.2 The Mean Field IRL Framework

We next present our proposed framework to solve the optimisation problem of ILIRL in Eq.(9), which we name as *Mean Field IRL* (MFIRL). The framework solves the ILIRL problem in a manner of bilevel optimisation, where the upper-level task is to tune the reward function $r(s, a, \mu)$ given the solution of the lower-level task that computes a best-response policy to the expert MF flow μ^E. Recall the fixed point iteration for computing MFNE from Sec. 3.2, where the mapping $\hat{\Psi}: \mu \mapsto \pi$ is used to identify a best-response policy to μ under a reward function r. Using this notation, we are left with finding a suitable $\hat{\Psi}$. But first, let us see how to solve the ILIRL problem in a more feasible way, where we estimate expected returns from $\mathcal{D}$ and use a parameterised reward function. Immediately after that, the instantiation of $\hat{\Psi}$ will be given.

5.2.1 Estimated Expected Returns and Parameterised Reward Functions. Since the true MF flow μ^E is unknown, we use an empirical value $\hat{\mu}^E \triangleq \{\hat{\mu}_t^E\}_{t=0}^T$ estimated from $\mathcal{D}$ by averaging the frequencies of state occurrences:

$$\hat{\mu}_t^E(s) = \frac{1}{M}\sum_{j=1}^M \mathbb{1}_{\{s_t^j = s\}}. \quad (10)$$

Since under an MFNE, the mean field matches each individual's state visitation distribution, $\hat{\mu}^E$ is an unbiased estimator of μ^E.

Following standard practice [16, 17, 57], we assume the reward function r is parameterised by $\omega \in \mathbb{R}^d$ and thus write it as r_ω. Also assume (μ^E, π^E) is induced by some *unknown* true parameter $\omega^\star$, i.e., $(\mu^E, \pi^E) = (\mu^{\omega^\star}, \pi^{\omega^\star})$. Let $\pi^\omega = \hat{\Psi}(\hat{\mu}^E; \omega)$ denote a best-response policy to $\hat{\mu}^E$ under r_ω. As illustrated in Fig. 2, optimising the problem in Eq. (9) on demonstration data $\mathcal{D}$ reduces to a search process for $\omega^\star$, where $J(\mu^E, \pi^E; \omega)$ is estimated from $\mathcal{D}$:

$$\max_\omega \mathcal{L}(\omega) \triangleq \mathbb{E}_{\tau \sim \mathcal{D}}\left[\sum_{t=0}^{T-1} \gamma^t r_\omega(s_t, a_t, \hat{\mu}_t^E)\right] - J(\hat{\mu}^E, \pi^\omega). \quad (11)$$

5.2.2 Characterising Best-Response Policies. The desired mapping $\hat{\Psi}$ relies on the *action value function* for MFGs, which represents the expected cumulative future rewards guided by an MF flow μ and a policy π. Formally, it is defined by

$$Q(t, s, a, \mu) \triangleq r(s, a, \mu_t) + \mathbb{E}\left[\sum_{\ell=t+1}^T \gamma^{\ell-t} r(s_\ell, a_\ell, \mu_\ell)\Big|\mu, \pi, P\right]. \quad (12)$$

The expected return can be expressed in terms of Q as follows: $J(\mu, \pi) = \mathbb{E}_{s \sim \mu_0}\left[\sum_{a \in \mathcal{A}} \pi_0(a|s) \, Q(0, s, a, \mu)\right]$. We can use *backward induction* to recursively compute the action value function: starting from the terminal step $t = T$: $Q(T, s, a, \mu) = r(s, a, \mu_T)$; then for $0 \le t < T$ we recursively compute:

$$Q(t, s, a, \mu) = r(s, a, \mu_t) + \gamma\mathbb{E}\left[Q(t+1, s', a', \mu)|\mu, \pi, P\right]. \quad (13)$$

For a fixed MF flow μ, we say a policy π^* is *greedy* with respect to μ if $\pi_t^*(\cdot|s)$ picks an action

$$a_t \in \underset{a \in \mathcal{A}}{\arg\max} Q(t, s, a, \mu) \quad (14)$$

uniformly at random. Since π^* maximises the action value function for each step, it is a best response to the corresponding MF flow μ. This provides an intuition to define the mapping $\pi^\omega = \hat{\Psi}(\hat{\mu}^E; \omega)$ by letting π^ω be greedy with respect to μ^E. More formally, we write $Q_\omega^*(t, s, a, \hat{\mu}^E)$ for the optimal action value function induced by a greedy policy given r_ω. By Eq. (13) and Eq. (14), π^ω and Q_ω^* can be recursively computed by backward induction. The expected return induced by π^ω in Eq. (11) can thereby be written as:

$$J(\hat{\mu}^E, \pi^\omega) = \mathbb{E}_{s \sim \hat{\mu}_0^E}\left[\sum_{a \in \mathcal{A}} \pi_0^\omega(a|s) \, Q_\omega^*(0, s, a, \hat{\mu}^E)\right]. \quad (15)$$

5.3 Theoretical Result

Now, we are ready to present our main result, which shows that the optimal solution to the optimisation problem in Eq. (11) is an asymptotically consistent estimator of the true reward parameter.

THEOREM 1. *Let the demonstrated trajectories in $\mathcal{D} = \{\tau^j\}_{j=1}^M$ be independent and identically distributed (i.i.d.) and sampled from an MFNE induced by an unknown parameterised reward function $r_{\omega^\star}(s, a, \mu)$. With probability 1 as the number of samples $M \to \infty$, the equation $\mathcal{L}(\omega) = 0$ has a root $\hat{\omega}$ such that $\hat{\omega} = \omega^\star$.*

PROOF. Under the i.i.d. assumption, $\hat{\mu}^E = \mu^E$ with probability 1 as $M \to \infty$, due to the law of large numbers. Having this, we further know that $\mathcal{L}(\hat{\omega}) = 0$ if and only if (μ^E, π^E) is an MFNE under reward $r_{\hat{\omega}}$. Finally, due to the fact (μ^E, π^E) is an MFNE induced by $r_{\omega^\star}$, there must exist one $\hat{\omega}$ such that $\hat{\omega} = \omega^\star$. □

6 PRACTICAL MFIRL ALGORITHM

This section develops a practical implementation of the MFIRL framework. For the sake of practical use, we consider gradient methods to optimise the objective $\mathcal{L}(\omega)$ in Eq. (11), where the gradient $\nabla\mathcal{L}$ is computed by

$$\nabla\mathcal{L} = \mathbb{E}_{\tau \sim \mathcal{D}}\left[\sum_{t=0}^{T-1} \gamma^t \nabla r_\omega(s_t, a_t, \hat{\mu}_t^E)\right] - \nabla J(\hat{\boldsymbol{\mu}}^E, \boldsymbol{\pi}^\omega). \tag{16}$$

By Eq. (15), the gradient $\nabla J(\hat{\boldsymbol{\mu}}^E, \boldsymbol{\pi}^\omega)$ is computed by:

$$\nabla J(\hat{\boldsymbol{\mu}}^E, \boldsymbol{\pi}^\omega) = \mathbb{E}_{s \sim \hat{\mu}_0^E}\left[\sum_{a \in \mathcal{A}} Q_\omega^*(0, s, a, \hat{\boldsymbol{\mu}}^E)\nabla\pi_0^\omega(a|s)\right.$$
$$\left. + \pi_0^\omega(a|s)\nabla Q_\omega^*(0, s, a, \hat{\boldsymbol{\mu}}^E)\right]. \tag{17}$$

While, one difficulty with such an approach is that the greedy policy $\boldsymbol{\pi}^\omega$ is non-differentiable [20]. We thus need to find alternative smooth $\hat{\Psi}$ mapping. To this end, we adopt Boltzmann policy [3] to approximate the non-differentiable greedy policy. Formally, we use $\tilde{\boldsymbol{\pi}}^\omega$ to denote a Boltzmann policy and use $\tilde{Q}_\omega^*$ to represent the corresponding action value function, which are defined by:

$$\tilde{\pi}_t^\omega(a|s) \triangleq \frac{\exp\left(\beta\tilde{Q}_\omega^*(t, s, a, \hat{\boldsymbol{\mu}}^E)\right)}{\sum_{a' \in \mathcal{A}} \exp\left(\beta\tilde{Q}_\omega^*(t, s, a', \hat{\boldsymbol{\mu}}^E)\right)}. \tag{18}$$

Here, $\beta > 0$ is the *inverse Boltzmann temperature* controlling the degree of approximation. Note that we recover the optimality (i.e., the greedy policy) if $\beta \to \infty$. By Eq. (18) and Eq. (13), gradients $\nabla\tilde{\pi}_t^\omega(a|s)$ and $\nabla\tilde{Q}_\omega^*(t, s, a, \hat{\boldsymbol{\mu}}^E)$ can be recursively calculated:

$$\nabla\tilde{\pi}_t^\omega(a|s) = \tilde{\pi}_t^\omega(a|s) \cdot \beta\left[\nabla\tilde{Q}_\omega^*(t, s, a, \hat{\boldsymbol{\mu}}^E)\right.$$
$$\left. - \mathbb{E}_{a' \sim \tilde{\pi}_t^\omega(\cdot|s)}\left[\nabla\tilde{Q}_\omega^*(t, s, a', \hat{\boldsymbol{\mu}}^E)\right]\right], \tag{19}$$

$$\nabla\tilde{Q}_\omega^*(t, s, a, \hat{\boldsymbol{\mu}}^E) = \nabla r_\omega(s, a, \hat{\mu}_t^E)$$
$$+ \gamma\mathbb{E}_{s' \sim P}\left[\sum_{a' \in \mathcal{A}} \tilde{Q}_\omega^*(t+1, s', a', \hat{\boldsymbol{\mu}}^E)\nabla\tilde{\pi}_{t+1}^\omega(a'|s')\right.$$
$$\left. + \mathbb{E}_{a' \sim \tilde{\pi}_{t+1}^\omega(\cdot|s')}\left[\nabla\tilde{Q}_\omega^*(t+1, s', a', \hat{\boldsymbol{\mu}}^E)\right]\right]. \tag{20}$$

Detailed derivation of $\nabla\tilde{\pi}_t^\omega(a|s)$ in Eq. (19) is given in Appendix A. Substituting $\nabla\tilde{Q}_\omega^*$ for ∇Q_ω^* in Eq. (17) yields an approximate $\nabla\mathcal{L}$, which we denote by $\nabla\tilde{\mathcal{L}}$.

Monte-Carlo Simulation under Unknown Dynamics. The transition function P is generally unknown in practice. We can instead use Monte-Carlo simulation to estimate empirical gradients $\hat{\nabla}\tilde{Q}_\omega^*$ and $\hat{\nabla}\tilde{\pi}_t^\omega$, that is, suppose we have the access to a simulator of the environment and calculate $\hat{\nabla}\tilde{Q}_\omega^*$ by estimating the expectation with respect to s' in Eq. (20) using sampled states s' from the simulator. We denote the resulting empirical gradient of $\mathcal{L}$ by $\hat{\nabla}\tilde{\mathcal{L}}$.

---

[3]Other smooth operators (e.g., Mellowmax [3]) may also be used to approximate $\hat{\Psi}$.

Algorithm 1 Practical MFIRL Algorithm

1: **Input:** MFG with parameters $(\mathcal{S}, \mathcal{A}, P, \mu_0, \gamma)$ and expert demonstrations $\mathcal{D} = \{\tau^j\}_{j=1}^M$.
2: **Initialisation:** Initialise reward parameter ω.
3: Estimate empirical expert MF flow $\hat{\boldsymbol{\mu}}^E$ according to Eq. (10).
4: **for** each epoch **do**
5:    **for** $t = T, \ldots, 0$ **do**
6:       Calculate $\nabla\tilde{\pi}_t^\omega$ and $\nabla\tilde{Q}_\omega^*$ according to Eq. (19) and Eq. (20).
7:    **end for**
8:    Calculate the empirical gradient $\hat{\nabla}\tilde{\mathcal{L}}$ according to Eq. (16).
9:    Update ω to increase $\mathcal{L}$ according to $\hat{\nabla}\tilde{\mathcal{L}}$.
10: **end for**
11: **Output:** Learned reward function r_ω.

Truncated Recursive Computation of Gradients. With the horizon of an MFG increasing, the recursive computation of $\nabla\tilde{Q}_\omega^*(t, s, a, \hat{\boldsymbol{\mu}}^E)$ tends to be intractable. To overcome this potential issue in practice, we can approximate $\nabla\tilde{Q}_\omega^*(t, s, a, \hat{\boldsymbol{\mu}}^E)$ by truncating the recursion. More formally, let $\omega^{(i)}$ denote the reward parameter in the ith round of update. In the $i + 1$th update, we look H-step ($H < T$) ahead according to Eq. (20) until we arrive at time step $t + H$, where we stop tracing deeper by replacing $\nabla\tilde{Q}_\omega^*(t+H+1, s, a, \hat{\boldsymbol{\mu}}^E)|_{\omega=\omega^{(i)}}$ and $\nabla\tilde{\pi}_{t+H+1}^\omega(a'|s')|_{\omega=\omega^{(i)}}$ with the corresponding variables induced by the old value $\omega^{(i-1)}$. Using old parameter values to approximate the corresponding variables induced by new values has also been adopted in the RL literature [46, 47]. The intuition is that the change of the parameter between two updates is not severe. Meanwhile, due to the presence of the discount factor, this approximation would not cause a significant error in the estimation of gradients.

To summarise, we present the pseudocode in Alg. 1.

7 EXPERIMENTS

In experiments, we seek to answer the following key question: *Can MFIRL efficiently recover the underlying individual reward function, regardless of whether the environment is cooperative or not?* To this end, we evaluate the quality of a learned reward function r_ω by comparing its induced MFSO $(\bar{\mu}^\omega, \bar{\pi}^\omega)$ and the ground-truth MFSO $(\bar{\mu}^\star, \bar{\pi}^\star)$, because an MFSO is more likely to be unique than an MFNE in terms of policy and it is always unique in term of expect return. Specifically, we use the following metrics:

(1) *Policy Deviation* (Dev. Policy). We use the cumulative KL-divergence, $\sum_{t=0}^{T}\sum_{s \in \mathcal{S}} D_{\text{KL}}(\bar{\pi}_t^\star(\cdot|s) \parallel \bar{\pi}_t^\omega(\cdot|s))$, to measure the difference over two policies.
(2) *MF flow Deviation* (Dev. MF). Similarly, we use the cumulative KL-divergence, $\sum_{t=0}^{T} D_{\text{KL}}(\bar{\mu}_t^\star \parallel \bar{\mu}_t^\omega)$, to measure the difference over two MF flows.
(3) *Expected return.* (Exp. Return) The expected return of $(\bar{\mu}^\omega, \bar{\pi}^\omega)$ and $(\bar{\mu}^\star, \bar{\pi}^\star)$ under the ground-truth reward function.

We compare MFIRL against the PLIRL in [55] as it is the only IRL method for MFGs in the literature as of the present. Also, existing multi-agent IRL and imitation learning methods (e.g., MA-GAIL [48] and MA-AIRL [57]) scale poorly when the population size is in hundreds, due to the exponential growth of joint state-action spaces. We carry out two classes of tests, one on numerical MFG models and
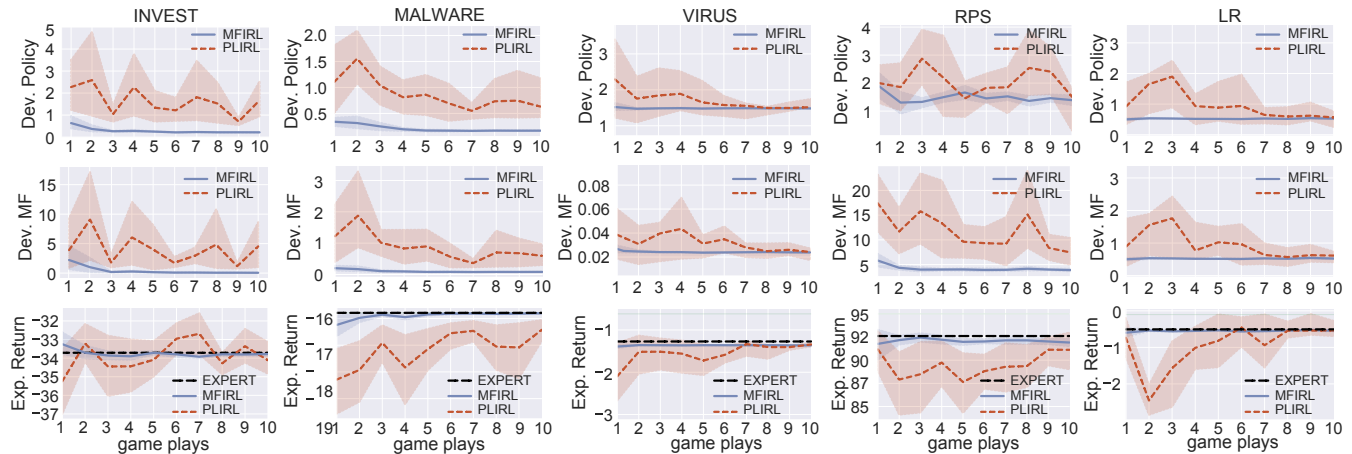
Figure 3: Results for numerical tasks under original dynamics. The solid line shows the median and the shaded area represents the standard deviation over 10 independent runs.

the other on simulated mixed cooperative-competitive battle games. For both classes of tests, we set the discounted factor $\gamma = 0.99$ and the time horizon as 50, which is the same as the number of time steps used in [48, 57]. For MFIRL, we set the inverse Boltzmann temperature $\beta = 1$. In each task, we set N agents where N is a large but finite number. We sample state-action trajectories from each of N individual agents by executing a pre-trained expert policy. We call an execution of the expert policy on all N agents a *game play*. MFIRL directly takes as input these individual trajectories. While, each demonstrated expert policy and mean field fed into PLIRL are estimated by averaging occurrence frequencies of states and actions in all N trajectories per game play. For both algorithms, we adopt the same neural network architecture as the reward model: two hidden layers of 64 leaky rectified linear units (ReLU) each. Implementation details are given in Appendix C.

7.1 Numerical Models

Settings. We evaluate MFIRL on five numerical discrete MFG models: *investment in product quality* [50, 54] (INVEST for short), *malware spread* [23, 24, 50] (MALWARE), *virus infection* [14] (VIRUS), *Rock-Paper-Scissors* [14] (RPS) and *Left-Right* [14] (LR), ordered in decreasing complexity. These models simulate a series of large-scale decision making scenarios in the real world. Among these five models, we use VIRUS and LR as cooperative scenarios, i.e., the demonstrations in these two models are sampled from MFSO. The remaining three models are used as non-cooperative scenarios. Statistical information of these models is summarised in Tab. 1. Detailed descriptions and settings can be found in Appendix B. We take 100 agents for each model. We train MFNE experts through the fixed-point iteration as described in Sec. 3.2, and train MFSO experts using DDPG [33] to solve the MDP reduced from MFG.

Reward Recovery under Original Dynamics. We first conduct tests under fixed environment dynamics. Fig. 3 depicts results. On all non-cooperative models, MFIRL achieves near-expert performance, while PLIRL shows larger deviations. The reason is that the demonstrations in these models are sampled from an MFNE rather than

Table 1: Statistics of Numerical MFG Models

Model	States	Actions	Cooperative
INVEST	10	2	✗
MALWARE	10	2	✗
VIRUS	2	2	✓
RPS	3	3	✗
LR	3	2	✓

MFSO. This verifies the biased reward inference in PLIRL. On cooperative models, both MFIRL and PLIRL show expert-like performance, but MFIRL is more sample efficient. This is because one game play provides N samples to our MFIRL but can only provide one sample of MF flow and policy to PLIRL.

Robustness to New Dynamics. To investigate the robustness of the learned reward function r_ω in the face of uncertainties in dynamics, we change the transition function (see Appendix B for details), re-compute MFSO induced by the ground-truth reward and r_ω (trained with 10 game plays), respectively. Results are summarised in Tab. 2. MFIRL has comparable performance with PLIRL on cooperative models but shows much smaller errors on non-cooperative models. We attribute the low robustness of PLIRL to the conjecture that the changing dynamics can exacerbate the biased reward inference.

7.2 Simulated Battle Games

Settings. The *Mixed Cooperative-Competitive Battle Game* [58] contains two groups of homogenous agents fighting against each other in a 2D grid. The goal of each group is to destroy all opponents in the other group. Each group has 64 agents. Each agent can move to the neighbourhood or take attack actions. The setting is illustrated in Fig. 4. For each agent, the default reward setting is: -0.005 for every move (M), 0.2 for attacking an opponent (AO), 5 for killing an opponent (K), -0.1 for attacking an empty grid (AE). If an agent is attacked or destroyed, it also receives a reward -1. To adapt the game for MFG, we supply an additional term to the default reward

Table 2: Results for numerical models under new dynamics. Mean and variance are taken across 10 independent runs.

Metric	Algorithm	Task				
		`INVEST`	`MALWARE`	`VIRUS`	`RPS`	`LR`
Dev. Policy	MFIRL	**0.305** ± 0.017	**0.411** ± 0.025	**1.544** ± 0.012	**7.089** ± 0.541	**0.683** ± 0.035
	PLIRL	1.130 ± 0.334	1.466 ± 1.322	1.892 ± 0.237	7.550 ± 0.841	0.734 ± 0.373
Dev. MF	MFIRL	**0.464** ± 0.029	**0.435** ± 0.007	**0.057** ± 0.0004	**2.932** ± 0.057	0.353 ± 0.029
	PLIRL	1.510 ± 0.697	1.731 ± 2.207	0.076 ± 0.0085	3.112 ± 0.569	**0.348** ± 0.310
Expected Return	EXPERT	-35.051	-18.055	-1.167	94.274	-0.518
	MFIRL	**-35.542** ± 0.677	**-18.519** ± 0.245	-1.614 ± 0.042	**93.578** ± 2.508	-0.563 ± 0.078
	PLIRL	-35.917 ± 2.548	-19.151 ± 0.507	**-1.553** ± 0.170	93.212 ± 0.493	**-0.547** ± 1.080
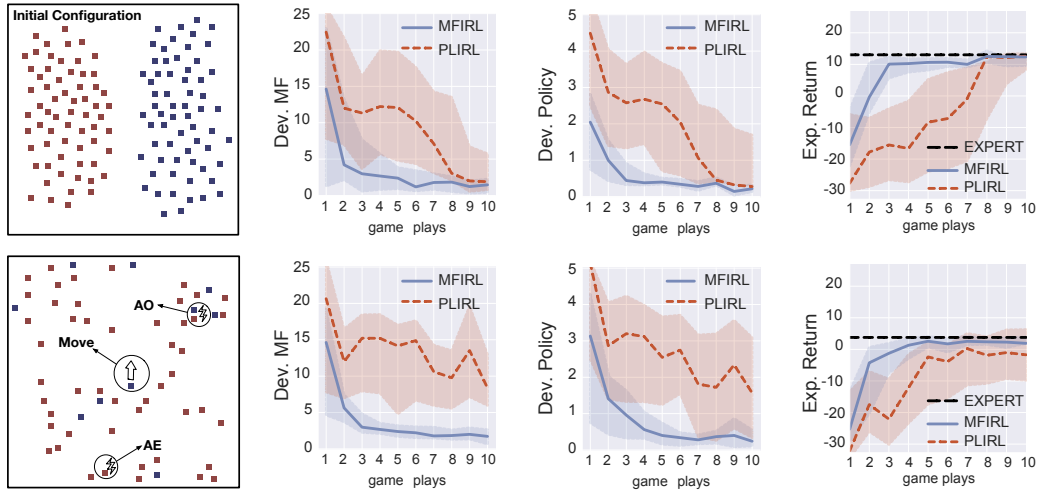
Figure 4: Illustration and results for simulated battle games. The first and second rows correspond to the cooperative and mixed cooperative-competitive setting, respectively. Results are averaged over 10 independent runs.

function according to the following intuition: it is less risky to attack a foe if more friends are nearby. The mean field can thus be involved in the reward to measure the average distance between an agent and all other friends. Formally, let $r(s, a)$ denote the default reward and write modified reward function by:

$$r(s, a, \mu) = r(s, a) - d \cdot 1_{\{a=\text{AO}\}} \cdot \mathbb{E}_{s' \sim \mu} \left[\texttt{dist}(s, s') \right],$$

where dist denotes the Manhattan distance and $d > 0$ controls its importance. In experiments, we set $d = 0.1$. We generate two types of environments: (1) Mixed cooperative-competitive: the reward of each agent at each step is defined above. (2) Fully cooperative: the reward of each agent is set as the societal reward of the group it belongs to. We train experts (of each group) using MADDPG [36], a multi-agent actor-critic algorithm. If an agent is destroyed halfway, we treat all variables of it as null in the subsequent training.

Results. Results are reported in Fig. 4. Consistently, MFIRL demonstrates higher accuracy in non-cooperative environments and higher sample efficiency in both cooperative and non-cooperative environments. PLIRL again shows large deviations from the expert performance. To summarise, MFIRL can accurately recover individual reward functions for MFGs in the non-cooperative setting with high sample efficiency, in line with our theoretical analysis.

8 CONCLUSIONS AND FUTURE WORK

This paper amounts to an effort towards individual-level IRL for MFGs. We reveal that the reduction from MFG to MDP holds only for the fully cooperative setting, which restricts the suitability of existing IRL methods in general MFGs. In order to handle MFGs with general non-cooperative environments, we propose and formalise the individual-level IRL problem that asks for recovering an individual reward function for MFGs. To address this problem, we propose MFIRL, the first dedicated IRL framework for MFGs that can deal with both cooperative and non-cooperative environments. Moreover, by making a series of approximations to the MFIRL framework, we develop a practical algorithm effective for MFGs with unknown dynamics. Experiments on both cooperative and non-cooperative scenarios verify the advantages of MFIRL on reward recovery, sample efficient and robustness to changing dynamics.

Alongside the direction opened up by this work, a straightforward future work is to scale MFIRL to MFGs with continuous or high-dimensional state-action spaces. Another promising work is to study IRL methods for general MFGs that can tolerate imperfect expert demonstrations. A third future work is to apply MFIRL to more real-world scenarios, e.g., dynamic demand management in power grids [4] and behaviour analysis in large social media [55].

REFERENCES

[1] Pieter Abbeel and Andrew Y Ng. 2004. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the 21st International Conference on Machine Learning*. 1.

[2] Yong-Yeol Ahn, Seungyeop Han, Haewoon Kwak, Sue Moon, and Hawoong Jeong. 2007. Analysis of topological characteristics of huge online social networking services. In *Proceedings of the 16th International Conference on World Wide Web*. 835–844.

[3] Kavosh Asadi and Michael L Littman. 2017. An alternative softmax operator for reinforcement learning. In *International Conference on Machine Learning*. 243–252.

[4] Fabio Bagagiolo and Dario Bauso. 2014. Mean-field games and dynamic demand management in power grids. *Dynamic Games and Applications* 4, 2 (2014), 155–176.

[5] Samuel Barrett, Avi Rosenfeld, Sarit Kraus, and Peter Stone. 2017. Making friends on the fly: Cooperating with new teammates. *Artificial Intelligence* 242 (2017), 132–171.

[6] Ana LC Bazzan. 2009. Opportunities for multiagent systems and multiagent reinforcement learning in traffic control. *Autonomous Agents and Multi-Agent Systems* 18, 3 (2009), 342.

[7] Kenneth Bogert and Prashant Doshi. 2014. Multi-robot inverse reinforcement learning under occlusion with interactions. In *Proceedings of the 13th International Conference on Autonomous Agents and Multi-agent Systems*. Citeseer, 173–180.

[8] Martin Burger, Marco Di Francesco, Peter A Markowich, and Marie-Therese Wolfram. 2014. Mean field games with nonlinear mobilities in pedestrian dynamics. *Discrete and Continuous Dynamical Systems-Series B* 19, 5 (2014), 1311–1333.

[9] Pierre Cardaliaguet and Charles-Albert Lehalle. 2018. Mean field game of controls and an application to trade crowding. *Mathematics and Financial Economics* 12, 3 (2018), 335–363.

[10] René Carmona, François Delarue, and Aimé Lachapelle. 2013. Control of McKean–Vlasov dynamics versus mean field games. *Mathematics and Financial Economics* 7, 2 (2013), 131–166.

[11] René Carmona, Mathieu Laurière, and Zongjun Tan. 2019. Model-free mean-field reinforcement learning: mean-field MDP and mean-field Q-learning. *arXiv preprint arXiv:1910.12802* (2019).

[12] Philippe Casgrain and Sebastian Jaimungal. 2019. Algorithmic trading in competitive markets with mean field games. *SIAM News* 52, 2 (2019), 1–2.

[13] Jaedeug Choi and Kee-Eung Kim. 2011. MAP Inference for Bayesian Inverse Reinforcement Learning. In *Advances in Neural Information Processing Systems*. 1989–1997.

[14] Kai Cui and Heinz Koeppl. 2021. Approximately Solving Mean Field Games via Entropy-Regularized Deep Reinforcement Learning. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 1909–1917.

[15] Romuald Elie, Julien Pérolat, Mathieu Laurière, Matthieu Geist, and Olivier Pietquin. 2020. On the Convergence of Model Free Learning in Mean Field Games.. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*. 7143–7150.

[16] Chelsea Finn, Sergey Levine, and Pieter Abbeel. 2016. Guided cost learning: deep inverse optimal control via policy optimization. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning*. 49–58.

[17] Justin Fu, Katie Luo, and Sergey Levine. 2018. Learning Robust Rewards with Adverserial Inverse Reinforcement Learning. In *International Conference on Learning Representations*.

[18] Diogo A Gomes, Joana Mohr, and Rafael Rigao Souza. 2010. Discrete time, finite state space mean field games. *Journal de Mathématiques Pures et Appliquées* 93, 3 (2010), 308–328.

[19] Diogo A Gomes, Levon Nurbekyan, and E Pimentel. 2015. Economic models and mean-field games theory. *Publicaoes Matematicas, IMPA, Rio, Brazil* (2015).

[20] Xin Guo, Anran Hu, Renyuan Xu, and Junzi Zhang. 2019. Learning mean-field games. In *Advances in Neural Information Processing Systems*. 4967–4977.

[21] Jonathan Ho and Stefano Ermon. 2016. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*. 4565–4573.

[22] Minyi Huang, Peter E Caines, and Roland P Malhamé. 2003. Individual and mass behaviour in large population stochastic wireless power control problems: centralized and Nash equilibrium solutions. In *42nd IEEE Conference on Decision and Control*, Vol. 1. IEEE, 98–103.

[23] Minyi Huang and Yan Ma. 2016. Mean field stochastic games: Monotone costs and threshold policies. In *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE, 7105–7110.

[24] Minyi Huang and Yan Ma. 2017. Mean field stochastic games with binary actions: Stationary threshold policies. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 27–32.

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

[26] Seong Hoon Jeong, Ah Reum Kang, and Huy Kang Kim. 2015. Analysis of Game Bot's Behavioral Characteristics in Social Interaction Networks of MMORPG.

[27] Henrik Kretzschmar, Markus Spies, Christoph Sprunk, and Wolfram Burgard. 2016. Socially compliant mobile robot navigation via inverse reinforcement learning. *The International Journal of Robotics Research* 35, 11 (2016), 1289–1307.

[28] Aime Lachapelle, Julien Salomon, and Gabriel Turinici. 2010. Computation of mean field equilibria in economics. *Mathematical Models and Methods in Applied Sciences* 20, 04 (2010), 567–588.

[29] Aimé Lachapelle and Marie-Therese Wolfram. 2011. On a mean field game approach modeling congestion and aversion in pedestrian crowds. *Transportation research part B: methodological* 45, 10 (2011), 1572–1589.

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

[31] Simon Le Cleac'h, Mac Schwager, and Zachary Manchester. 2021. LUCIDGames: Online Unscented Inverse Dynamic Games for Adaptive Trajectory Prediction and Planning. *IEEE Robotics and Automation Letters* 6, 3 (2021), 5485–5492.

[32] Sergey Levine, Zoran Popovic, and Vladlen Koltun. 2011. Nonlinear inverse reinforcement learning with gaussian processes. In *Advances in Neural Information Processing Systems*. 19–27.

[33] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. 2016. Continuous control with deep reinforcement learning.. In *4th International Conference on Learning Representations*.

[34] Xiaomin Lin, Peter A Beling, and Randy Cogill. 2014. Multi-agent inverse reinforcement learning for zero-sum games. *arXiv preprint arXiv:1403.6508* (2014).

[35] Manuel Lopes, Francisco Melo, and Luis Montesano. 2009. Active learning for reward estimation in inverse reinforcement learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 31–46.

[36] Ryan Lowe, Yi I Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. In *Advances in Neural Information Processing Systems*. 6379–6390.

[37] David Mguni, Joel Jennings, and Enrique Munoz de Cote. 2018. Decentralised learning in systems with many, many strategic agents. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[38] Andrew Y Ng and Stuart J Russell. 2000. Algorithms for Inverse Reinforcement Learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*. 663–670.

[39] Lasse Peters, David Fridovich-Keil, Vicenç Rubies-Royo, Claire J Tomlin, and Cyrill Stachniss. 2021. Inferring Objectives in Continuous Dynamic Games from Noise-Corrupted Partial State Observations. *arXiv preprint arXiv:2106.03611* (2021).

[40] Matteo Pirotta and Marcello Restelli. 2016. Inverse Reinforcement Learning through policy gradient minimization. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*. 1993–1999.

[41] Deepak Ramachandran and Eyal Amir. 2007. Bayesian Inverse Reinforcement Learning. In *IJCAI*, Vol. 7. 2586–2591.

[42] Nathan D Ratliff, J Andrew Bagnell, and Martin A Zinkevich. 2006. Maximum margin planning. In *Proceedings of the 23rd International Conference on Machine Learning*. 729–736.

[43] Tummalapalli Sudhamsh Reddy, Vamsikrishna Gopikrishna, Gergely Zaruba, and Manfred Huber. 2012. Inverse reinforcement learning for decentralized non-cooperative multiagent systems. In *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 1930–1935.

[44] Simon Rothfuß, Jairo Inga, Florian Köpf, Michael Flad, and Sören Hohmann. 2017. Inverse optimal control for identification in non-cooperative differential games. *IFAC-PapersOnLine* 50, 1 (2017), 14909–14915.

[45] Naci Saldi, Tamer Basar, and Maxim Raginsky. 2018. Markov–Nash Equilibria in Mean-Field Games with Discounted Cost. *SIAM Journal on Control and Optimization* 56, 6 (2018), 4256–4287.

[46] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. 2015. Trust region policy optimization. In *International Conference on Machine Learning*. PMLR, 1889–1897.

[47] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[48] Jiaming Song, Hongyu Ren, Dorsa Sadigh, and Stefano Ermon. 2018. Multi-agent generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*. 7461–7472.

[49] Adrian Šošić, Wasiur R KhudaBukhsh, Abdelhak M Zoubir, and Heinz Koeppl. 2017. Inverse Reinforcement Learning in Swarm Systems. In *Proceedings of the 16th International Conference on Autonomous Agents and Multi-agent Systems*. 1413–1421.

[50] Jayakumar Subramanian and Aditya Mahajan. 2019. Reinforcement learning in stationary mean-field games. In *Proceedings of the 18th International Conference on Autonomous Agents and Multi-agent Systems*. 251–259.

[51] Sriram Subramanian, Pascal Poupart, Matthew E Taylor, and Nidhi Hegde. 2020. Multi Type Mean Field Reinforcement Learning. In *Proceedings of the 19th International Conference on Autonomous Agents and Multi-agent Systems*. 411–419.

ACM SIGCOMM Computer Communication Review 45, 4 (2015), 99–100.

[52] Umar Syed and Robert E Schapire. 2007. A game-theoretic approach to apprenticeship learning. In *Proceedings of the 20th International Conference on Neural Information Processing Systems*. 1449–1456.

[53] Kevin Waugh, Brian D Ziebart, and J Andrew Bagnell. 2011. Computational rationalization: the inverse equilibrium problem. In *Proceedings of the 28th International Conference on Machine Learning*. 1169–1176.

[54] Gabriel Y Weintraub, C Lanier Benkard, and Benjamin Van Roy. 2010. Computational methods for oblivious equilibrium. *Operations research* 58, 4-part-2 (2010), 1247–1265.

[55] Jiachen Yang, Xiaojing Ye, Rakshit Trivedi, Huan Xu, and Hongyuan Zha. 2018. Learning Deep Mean Field Games for Modeling Large Population Behavior. In *International Conference on Learning Representations*.

[56] Y Yang, R Luo, M Li, M Zhou, W Zhang, and J Wang. 2018. Mean Field Multi-Agent Reinforcement Learning. In *35th International Conference on Machine Learning*,

Vol. 80. PMLR, 5571–5580.

[57] Lantao Yu, Jiaming Song, and Stefano Ermon. 2019. Multi-Agent Adversarial Inverse Reinforcement Learning. In *International Conference on Machine Learning*. 7194–7201.

[58] Lianmin Zheng, Jiacheng Yang, Han Cai, Ming Zhou, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Magent: A many-agent reinforcement learning platform for artificial collective intelligence. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[59] Brian D Ziebart, J Andrew Bagnell, and Anind K Dey. 2010. Modeling interaction via the principle of maximum causal entropy. In *Proceedings of the 27th International Conference on Machine Learning*. 1255–1262.

[60] Brian D Ziebart, Andrew Maas, J Andrew Bagnell, and Anind K Dey. 2008. Maximum entropy inverse reinforcement learning. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*. 1433–1438.