Adversarial Inverse Reinforcement Learning for Mean Field Games

Yang Chen[†], Libo Zhang[*], Zhenyun Deng[†], Neset Özkan Tan[†], Jiamou Liu[*] and Michael Witbrock[†]

[†] Strong AI Lab, [†,*]School of Computer Science, University of Auckland

Auckland, New Zealand

{yang.chen,jiamou.liu,m.witbrock}@auckland.ac.nz,lzha797@aucklanduni.ac.nz

ABSTRACT

Goal-based agents respond to environments and adjust behaviour accordingly to reach objectives. Understanding incentives of interacting agents from observed behaviour is a core problem in multi-agent systems. Inverse reinforcement learning (IRL) provides a solution to this problem, which infers underlying reward functions by observing the behaviour of rational agents. Despite IRL being principled, it becomes intractable when the number of agents grows because of the curse of dimensionality and the explosion of interactions among agents. The formalism of Mean field games (MFGs) has gained momentum as a mathematically tractable paradigm for studying large-scale multi-agent systems. By grounding IRL in MFGs, recent research attempts to push the limits of the agent number in IRL. However, the study of IRL for MFGs is far from being mature as existing methods rely on the assumption of strong rationality, while real-world agents often exhibit bounded rationality due to the limited cognitive or computational capacity. Towards a more general and practical IRL framework for MFGs, this paper proposes Mean-Field Adversarial IRL, a novel framework that is capable of tolerating bounded rationality. We build it upon the maximum entropy principle, adversarial learning, and a new equilibrium concept for MFGs. We evaluate our machinery on simulated tasks with imperfect demonstrations resulting from bounded rationality. Experimental results demonstrate the superiority of MF-AIRL over existing methods in reward recovery.

KEYWORDS

Inverse Reinforcement Learning; Mean Field Games; Maximum Entropy Principle; Adversarial Learning

1 INTRODUCTION

Game theory provides a general framework for predicting the strategic behaviours of interacting agents [11]. It concerns itself with a set of reward (or utility) functions and understands the behaviours of rational agents, i.e., the equilibrium, to be the outcome of utility-maximising strategies. An inverse problem naturally arises from this setting: how to infer the reward functions from observed behaviours? This problem is known as *multi-agent inverse reinforcement learning* (MAIRL) [36]. More precisely, it seeks to find domain parameters for reward functions that induce observed behaviours of rational decision-makers.

The recent surge in the scale of real-world multi-agent systems (MAS) [13, 27, 45] has raised the need for solving MAIRL in the presence of a large number of agents. In fact, we can identify at least two motivations for MAIRL in large-scale MAS. A straightforward one is as its name implies – we wish to detect and understand the behaviour of a population of agents by modelling them using reward functions. Examples include modelling infection spread [24], discovering pricing strategies in large-scale markets [37] and understanding the mechanism of social-norm emergence in a large population [26]. A second motivation is for the sake of designing environments for large-scale MAS so that the expected behaviour emerges if agents are rational. The behaviour of a MAS is uncertain, and unexpected behaviour is likely to arise; the growing size of the system will further exacerbate this. If one is able to pinpoint the causal relations between rewards and rational behaviours, he can manipulate the system's behaviour by tuning the reward functions. Contrary to the examples above, applications here can be controlling and restraining infection spread, making pricing strategies in large-scale markets, and even guiding and shaping the emergence of social norms in communities, all in a data-driven manner.

However, MAIRL is notoriously intractable in the face of a large number of agents. This is because MAIRL typically takes *stochastic games* [25] as the mathematical model, where a (Nash) equilibrium is computationally intractable when the number of agents scales to tens or hundreds [6]. To accommodate the need for MAIRL in large-scale MAS, we thus ask for a mathematically tractable substitute model. The recent paradigm of *mean field games* (MFGs) [22, 23] achieves tractability by borrowing the idea of mean field approximation from statistical physics to simplify agent interactions. It takes the limit as the number of agents approaches infinity and reduces the whole-system interactions to those between a single individual agent and the *mean field*, a virtual agent that represents aggregated behaviour of the population at large. This dual-view interplay gives rise to *mean field Nash equilibrium* (MFNE), which stipulates bidirectional constraints between the two sides: every agent's policy maximises its rewards given the mean field and, in turn, the mean field is uniquely resulted by all agents' policies. Importantly, MFNE is shown to be an approximate Nash equilibrium in the corresponding finite-agent stochastic games [34]. To break through the limitation of agent number in MAIRL, it is thereby promising to transfer the concept of IRL to MFGs.

Unfortunately, IRL remains largely unexplored in MFGs, albeit there are two recent attempts. Yang et al. [41] first proposed a *centralised* IRL method for MFGs by showing that an MFG can be reduced to a *Markov decision process* (MDP) that describes the population's collective behaviour and average rewards; they thus applied single-agent IRL methods on top of this MDP. Subsequently, Chen et al. [4] revealed that this reduction holds only if the MFNE is *socially optimal*, i.e., it maximises the population's average rewards; they thus framed the problem at the *decentralised* setting, i.e., inferring the reward function for an individual agent from the observed individual behaviour rather than the population behaviour. They put forward *Mean Field IRL* (MFIRL), a more general approach effective for both socially optimal and ordinary MFNE.

However, both methods above are still limited in terms of practical use. First, due to limited cognitive or computational capability, an agent often has *bounded rationality* in real life, i.e., choosing satisfactory rather than optimal actions [19]. Consequently, the resulting behaviour possesses uncertainties in general. For example, a customer in a restaurant orders an acceptable dish which is not necessarily his favourite as he is rush in time. These two methods cannot reason about such uncertainties as agents in MFNE are assumed never to take suboptimal behaviour. This makes both methods unsuitable for situations where the agents are bounded rational. Then, since an MFNE is not unique in general [5], the observed behaviour may be insufficient for us to learn a reward function that determines a unique policy. In this sense, from an application standpoint, the existing methods are also not useful for environment design for large-scale MAS.

Towards a more general and practical IRL method for MFGs, we invoke the idea of Maximum Entropy IRL (MaxEnt IRL) [46, 47], which provides a general probabilistic framework to tackle bounded-rational behaviour. It is state-action trajectory-centric and assumes the observed trajectories follow a distribution (in terms of rewards) with the maximum entropy. It thus allows us to find a reward function that rationalises observed behaviour with the *least commitment*. Moreover, since a policy leading to the maximum entropy trajectory distribution is unique given a reward function, MaxEnt IRL is more useful for environment design. However, extending MaxEnt IRL to MFGs is challenging. First, since MFNE assumes agents never take suboptimal behaviour, it is incompatible with MaxEnt IRL in the sense that it cannot provide a trajectory distribution that can be used for the probabilistic reward inference. Second, since in MFGs, the individual and population dynamics are intertwined (the policy and mean field are interdependent), the trajectory distribution is intractable to express in terms of rewards, which would prevent us from performing probabilistic inference for reward functions.

The primary contribution of this paper lies in the proposal of a new probabilistic IRL framework, *Mean-Field Adversarial IRL* (MF-AIRL), for MFGs. MF-AIRL integrates ideas from decentralised IRL for MFGs, MaxEnt IRL, and adversarial learning [15] into a unified probabilistic model for reward inferences in large-scale MAS. We summarise specific contributions as follows: **(1)** We build MF-AIRL upon a new equilibrium concept termed *entropy-regularised MFNE* (ERMFNE) (see Sec. 3). We show that ERMFNE can characterise an individual's trajectory distribution induced by a reward function in a principled way (see Theorem 1). **(2)** Taking ERMFNE as the solution concept, we extend MaxEnt IRL to MFGs (see Sec. 4). We

decouple individual and population dynamics by deriving the empirical value of the mean field from the observed behaviour (see Theorem 2). **(3)** By using adversarial learning to solve MaxEnt IRL in MFGs efficiently, we develop the practical MF-AIRL framework (see Sec. 5). **(4)** We evaluate MF-AIRL on tasks that simulate scenarios of marketing strategy making, virus propagation modelling and social norm emergence, all on a large scale (see Sec. 7). Results demonstrate the outperformance of MF-AIRL over existing methods in reward recovery.

2 PRELIMINARIES

This section introduces *mean field games* (MFGs) and the *maximum entropy inverse reinforcement learning* (MaxEnt IRL) framework. The marriage of the two gives rise to our proposed multi-agent IRL approach dedicated to large-scale multi-agent systems.

2.1 Mean Field Games

Following the conventional MFG model in the learning setting, we focus on MFGs with finite state and action spaces and, more generally, a finite time horizon [8]. Consider an N-player symmetric game, i.e., all agents share the same *local state* space $\mathcal{S}$, *action* space $\mathcal{A}$, and a reward function that is invariant under the permutation of identities of agents without changing their states and actions. Let $(s^1, \ldots, s^N) \in \mathcal{S}^N$ denote a *joint state*, where $s^i \in \mathcal{S}$ is the state of the ith agent. As N grows large, the game becomes intractable to analyse due to the curse of dimensionality. MFGs achieve tractability by considering the asymptotic limit when N approaches infinity. Formally, take the limit as $N \to \infty$. Due to the homogeneity of agents, MFGs use an empirical distribution $\mu \in \mathcal{P}(\mathcal{S})$, called a *mean field*, to represent the statistical information of the joint state:

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

Here, $\mathcal{P}(\mathcal{S})$ represents the set all probability measures over $\mathcal{S}$ and $\mathbb{1}$ denotes the indicator function, i.e., $\mathbb{1}_x = 1$ if x is true and 0 otherwise. The *transition function* $P \colon \mathcal{S} \times \mathcal{A} \times \mathcal{P}(\mathcal{S}) \times \mathcal{P}(\mathcal{S}) \to [0, 1]$ specifies how states evolve, i.e., an agent transits to the next state s_{t+1} with the probability $P(s_{t+1}|s_t, a_t, \mu_t)$. Let $T \in \mathbb{N}^+$ denote a finite time horizon. A *mean field flow* (MF flow for short) thus consists of a sequence of $T + 1$ mean fields $\boldsymbol{\mu} \triangleq \{\mu_t\}_{t=0}^{T}$. The initial mean field μ_0 is given. The *running reward* at each step is determined by a bounded *reward function* $r \colon \mathcal{S} \times \mathcal{A} \times \mathcal{P}(\mathcal{S}) \to \mathbb{R}$. Let $\tau \triangleq \{(s_t, a_t)\}_{t=0}^{T}$ denote a state-action *trajectory* of an agent. We write an agent's long-term reward under a given MF flow $\boldsymbol{\mu}$ as $\mathcal{R}(\tau) \triangleq \sum_{t=0}^{T-1} r(s_t, a_t, \mu_t)$.[1] In summary, an MFG is defined as a tuple $(\mathcal{S}, \mathcal{A}, P, \mu_0, r)$.

A time-varying stochastic *policy* $\boldsymbol{\pi} \triangleq \{\pi_t\}_{t=0}^{T}$ is adopted to characterise a strategic agent, where each $\pi_t \colon \mathcal{S} \to \mathcal{P}(\mathcal{A})$ is the *per-step policy*, i.e., an agent chooses actions following $a_t \sim \pi_t(\cdot|s)$. Given an MF flow $\boldsymbol{\mu}$ and a policy $\boldsymbol{\pi}$, an agent's *expected return* (cumulative rewards) is written as

$$J(\boldsymbol{\mu}, \boldsymbol{\pi}) \triangleq \mathbb{E}_{\tau \sim (\boldsymbol{\mu}, \boldsymbol{\pi})} [\mathcal{R}(\tau)], \tag{1}$$

where $s_0 \sim \mu_0, a_t \sim \pi_t(\cdot|s_t), s_{t+1} \sim P(\cdot|s_t, a_t, \mu_t)$.

---

[1]Following the convention [4, 8, 41], we set the reward at the last step ($t = T$) as 0.

2.2 Mean Field Nash Equilibrium

Fixing an MF flow μ, a policy π is called a *best response* to μ if it maximises $J(\mu, \pi)$. We denote the set of all best-response policies to μ by $\Psi(\mu) \triangleq \arg\max_{\pi} J(\mu, \pi)$. However, since all agents optimise their policies simultaneously, the MF flow would shift. Thus, the solution must consider how a policy at the individual level affects the MF flow at the population level. Due to the homogeneity of agents, everyone would follow the same policy at optimality. The dynamics of the MF flow can thus be governed by the (discrete-time) *McKean-Vlasov* (MKV) equation [3]:

$$\mu_{t+1}(s') = \sum_{s \in \mathcal{S}} \mu_t(s) \sum_{a \in \mathcal{A}} \pi_t(a|s) \, P(s'|s, a, \mu_t). \qquad (2)$$

Given a policy π, denote $\mu = \Phi(\pi)$ as the MF flow that fulfils MKV equation. We say μ is *consistent* with π if $\mu = \Phi(\pi)$. This consistency guarantees that a single agent's state marginal distribution flow matches the MF flow at the population level. The conventional solution concept for MFGs is the *mean field Nash equilibrium*.

DEFINITION 1. *A pair* $(\mu^{\star}, \pi^{\star})$ *is called a* mean field Nash equilibrium *(MFNE) if it satisfies:*

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

Computing an MFNE typically requires a fixed-point iteration procedure for the MF flow [16]. Formally, through defining any mapping $\hat{\Psi} : \mu \mapsto \pi$ that identifies a best response in $\Psi(\mu)$, we obtain a fixed point iteration for μ by alternating between $\pi = \hat{\Psi}(\mu)$ and $\mu = \Phi(\pi)$. The assumption for the uniqueness of MFNE is that the fixed-point iteration will converge to a unique μ [16]. However, the fixed-point iteration is not guaranteed to converge to a unique μ, and multiple MFNE can coexist [5].

2.3 Maximum Entropy IRL

We next give an overview of MaxEnt IRL [46, 47] in the context of a Markov decision process (MDP) defined by a tuple $(\mathcal{S}, \mathcal{A}, P, \rho, r)$, where $r(s, a)$ is the reward function, and the environment dynamics is determined by the transition function $P(s'|s, a)$ and initial state distribution $\rho(s)$. In (forward) reinforcement learning (RL), an optimal policy may not exist uniquely. MaxEnt RL solves this ambiguity by augmenting the expected return with a *causal entropy* [2] [46] regularisation term $\mathcal{H}(\pi) \triangleq \mathbb{E}_{\pi}[-\log \pi(a|s)]$, i.e., the objective is to find a (stationary) policy $\pi^{\star}$ such that

$$\pi^{\star} = \arg\max_{\pi} \mathbb{E}_{\tau \sim \pi} \left[\sum_{t=0}^{T-1} r(s_t, a_t) + \beta \mathcal{H}(\pi(\cdot|s_t)) \right],$$

where τ is a state-action trajectory sampled via $s_0 \sim \rho_0, a_t \sim \pi(\cdot|s_t)$, $s_{t+1} \sim P(\cdot|s_t, a_t)$ and $\beta > 0$ controls the relative importance of reward and entropy.

Suppose we have no access to the reward function but have a set of observed trajectories sampled from an *unknown* expert policy π^E obtained by the above MaxEnt RL procedure. MaxEnt IRL aims to infer a reward function that rationalises the observed behaviour, which reduces to the following maximum likelihood estimation

(MLE) problem (assume $\beta = 1$ [44]):

$$p_{\omega}(\tau) \propto \rho(s_0) \cdot \prod_{t=0}^{T-1} P(s_{t+1}|s_t, a_t) \cdot e^{R_{\omega}(\tau)}, \qquad (3)$$

$$\max_{\omega} \mathbb{E}_{\tau \sim \pi^E} \left[\log p_{\omega}(\tau) \right] = \mathbb{E}_{\tau \sim \pi^E} \left[R(\tau) \right] - \log Z_{\omega}. \qquad (4)$$

Here, $R_{\omega}(\tau) \triangleq \sum_{t=0}^{T-1} r_{\omega}(s_t, a_t)$ where r_{ω} is an ω-parameterised reward function, and $Z_{\omega} \triangleq \sum_{\tau \sim \pi^E} e^{R_{\omega}(\tau)}$ is the *partition function* of the distribution defined in Eq. (3), i.e., a summation over all feasible trajectories. Exactly computing Z_{ω} is intractable if the state-action space is large.

Adversarial IRL (AIRL) was proposed by [9] as an efficient sampling-based approximation to MaxEnt IRL, which reframes Eq. (4) as optimising a *generative adversarial network* [15]. It uses a discriminator D_{ω} (a binary classifier) and a *adaptive sampler* π_{θ} (a policy). Particularly, the discriminator takes the following form:

$$D_{\omega}(s, a) = \frac{e^{f_{\omega}(s,a)}}{e^{f_{\omega}(s,a) + \pi_{\theta}(a|s)}},$$

where f_{ω} serves as the parameterised reward function. The update of D_{ω} is interleaved with the update of π_{θ}: D_{ω} is trained to update the reward function by distinguishing between the trajectories sampled from the expert and the adaptive sampler; while π_{θ} is trained to maximise

$$\mathbb{E}_{\tau \sim \pi_{\theta}} \left[\sum_{t=0}^{T-1} \log D_{\omega}(s_t, a_t) - \log(1 - D_{\omega}(s_t, a_t)) \right].$$

IRL faces the ambiguity of *reward shaping* [29], i.e., multiple reward functions can induce the same optimal policy. To mitigate this ambiguity, Fu et al. [9] further restrict the parameterised reward in D_{ω} to a specific structure by supplying a state-only *potential-based reward shaping* function $h_{\phi} : \mathcal{S} \to \mathbb{R}$:

$$f_{\omega,\phi}(s_t, a_t, s_{t+1}) = r_{\omega}(s_t, a_t) + h_{\phi}(s_{t+1}) - h_{\phi}(s_t).$$

Shown in [9], under certain conditions, $r_{\omega}(s, a) + h_{\phi}(s)$ will recover the ground-truth reward function up to a constant.

2.4 IRL for MFGs

We adopt the general decentralised IRL setting for MFGs as in [4], which aims to infer the individual reward function from observed individual behaviour. More formally, let $(\mathcal{S}, \mathcal{A}, P, \mu_0, r)$ be an MFG. Suppose we do not know $r(s, a, \mu)$ but have a set of M observed expert behaviour $\mathcal{D}_E = \{\tau_j\}_{j=1}^{M}$ sampled from an *unknown* equilibrium (μ^E, π^E), where each $\tau = \{(s_t, a_t)\}_{t=0}^{T}$ is an individual agent's state-action trajectory sampled via $s_0 \sim \mu_0$, $a_t \sim \pi_t(\cdot|s_t)$, $s_{t+1} \sim P(\cdot|s_t, a_t, \mu_t)$. IRL for MFG asks for a reward function $r(s, a, \mu)$ under which (μ^E, π^E) constitutes an equilibrium.

3 ENTROPY-REGULARISED MFNE

This section introduces and justifies a new solution concept for MFGs, which allows us to characterise the bounded rationality of agents and thereby extend MaxEnt IRL to MFGs.

---

[2]Throughout the rest of the paper, we refer to the term entropy as the causal entropy.

3.1 A New Equilibrium Concept for MFGs

To extend MaxEnt IRL to MFGs, we need to characterise the trajectory induced by a reward function with a particular distribution as analogous to Eq. (3). However, MFNE cannot *explicitly* define a tractable trajectory distribution as it requires agents never to take suboptimal actions, whereas, in MaxEnt IRL, an agent can take sub-optimal actions with certain low probabilities. To bridge the gap between MFGs and MaxEnt IRL, we need a "soft" equilibrium concept that can characterise uncertainties in observed behaviour as against MFNE being a "hard" equilibrium. To this end, a natural way in game theory is to incorporate policy entropy into rewards [12, 31], which enables *bounded rationality*, i.e., agents can take satisfactory rather than optimal actions. This inspires a new solution concept – *entropy-regularised MFNE* (ERMFNE) – where an agent aims to maximise the entropy-regularised rewards:

$$\tilde{J}(\boldsymbol{\mu}, \boldsymbol{\pi}) \triangleq \mathbb{E}_{\tau \sim (\boldsymbol{\mu}, \boldsymbol{\pi})} \left[\sum_{t=0}^{T-1} r(s_t, a_t, \mu_t) + \beta \mathcal{H}(\pi_t(\cdot|s_t)) \right].$$

DEFINITION 2. *A pair of MF flow and policy $(\tilde{\boldsymbol{\mu}}^\star, \tilde{\boldsymbol{\pi}}^\star)$ is called an entropy-regularised MFNE (ERMFNE) if it satisfies:*

(1) *Agent bounded rationality:* $\tilde{J}(\tilde{\boldsymbol{\mu}}^\star, \tilde{\boldsymbol{\pi}}^\star) = \max_{\boldsymbol{\pi}} \tilde{J}(\tilde{\boldsymbol{\mu}}^\star, \boldsymbol{\pi})$;
(2) *Population consistency:* $\tilde{\boldsymbol{\mu}}^\star = \Phi(\tilde{\boldsymbol{\pi}}^\star)$.

Despite the entropy-regularised MFGs have been studied in the literature [1, 5, 17], existing work is motivated from a computational perspective, that is, entropy regularisation can improve the stability and the convergence of algorithms for computing an equilibrium. Particularly, Cui and Koeppl [5] independently proposed a similar solution concept and showed that entropy regularisation relaxes the condition for uniqueness as opposed to the unregularised case: **(1)** With entropy regularisation, the best-response policy $\tilde{\boldsymbol{\pi}}$ to an MF flow $\boldsymbol{\mu}$ exists *uniquely*; [3] **(2)** ERMFNE exists for any $\beta > 0$ if $r(s, a, \mu)$ and $P(s'|s, a, \mu)$ are continuous. Using $\tilde{\boldsymbol{\pi}} = \tilde{\Psi}(\boldsymbol{\mu})$ to denote the unique best-response policy to $\boldsymbol{\mu}$, we obtain the the fixed point iteration for ERMFNE by alternating between $\boldsymbol{\pi} = \tilde{\Psi}(\boldsymbol{\mu})$ and $\boldsymbol{\mu} = \Phi(\boldsymbol{\pi})$. The fixed point iteration converges to a unique MF flow if β is large (according to the reward function scale), thereby implying a *unique* ERMFNE.

Note that in ERMFNE, we recover optimality (MFNE) if $\beta = 0$. Although the optimality and uniqueness are approached respectively at two extremes of β, in this paper, we prioritise the property of uniqueness to ensure the well-definedness of IRL for MFGs, i.e., we assume trajectories are observed from a unique ERMFNE (with a suitable β) so that the observed behaviour can be interpreted with a unique equilibrium. Since we can adjust the relative importance between rewards and entropy by scaling reward functions, following the convention in MaxEnt IRL [9, 43, 44] and without loss of generality, we assume $\beta = 1$ in the remainder of the analysis.

3.2 Trajectory Distributions under ERMFNE

Besides uniqueness, in this paper, we take a step further to show that the entropy regularisation endows ERMFNE with the capability of reasoning about uncertainties in observed behaviour in a principled way. Specifically, we show that trajectory distributions induced

---

[3]Hereafter, we will slightly abuse the term "best response" to denote the policy that maximises the entropy-regularised rewards.
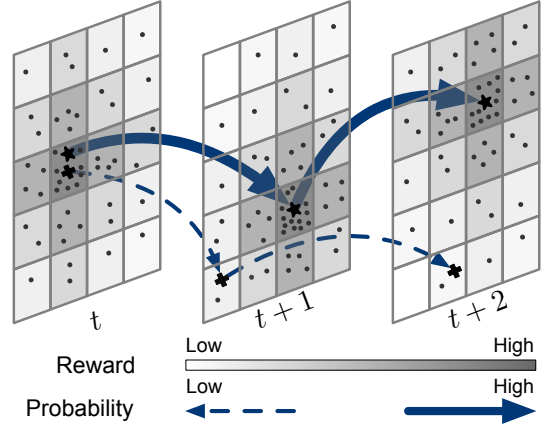

Figure 1: Illustration for the trajectory distribution induced by ERMFNE. A group of agents (dots) drift in a grid world over three time steps. Each grid represents a specific state. The distribution of agents represents the mean field. A darker grey background denotes a higher reward associated with a state. Two individual agents marked with "★" and "+" start at the same state. The solid arrows denote the trajectory of ★, which can be observed with an exponentially higher probability than the trajectory of + depicted using dashed arrows. Note that the rewards of states change over time because of the evolution of mean fields.

by ERMFNE can be characterised by an *energy-based model*, i.e., trajectories with high expected cumulative rewards are generated with exponentially high probabilities, as is illustrated in Fig. 1. It can thus be used for the probabilistic reward inference.

THEOREM 1. *Let $(\tilde{\boldsymbol{\mu}}^\star, \tilde{\boldsymbol{\pi}}^\star)$ be the ERMFNE for an MFG $(\mathcal{S}, \mathcal{A}, P, \mu_0, r)$, and D_{KL} denote the Kullback-Leibler (KL) divergence. Then, $(\tilde{\boldsymbol{\mu}}^\star, \tilde{\boldsymbol{\pi}}^\star)$ is the optimal solution to the following constrained optimisation problem:*

$$\min_{\boldsymbol{\mu}, \boldsymbol{\pi}} D_{\mathrm{KL}} \left(p_1(\tau) \parallel p_2(\tau) \right) \ \ s.t. \ \boldsymbol{\mu} = \Phi(\boldsymbol{\pi})$$

$$p_1(\tau) = \mu_0(s_0) \cdot \prod_{t=0}^{T-1} P(s_{t+1}|s_t, a_t, \mu_t) \cdot \prod_{t=0}^{T} \pi_t(a_t|s_t)$$

$$p_2(\tau) \propto \mu_0(s_0) \cdot \prod_{t=0}^{T-1} P(s_{t+1}|s_t, a_t, \mu_t) \cdot e^{\mathcal{R}(\tau)} \quad (5)$$

PROOF. See Appendix A. □

4 EXTENDING MAXENT IRL TO MFGS

From now on, we assume that observed trajectories are sampled from a unique ERMFNE $(\boldsymbol{\mu}^E, \boldsymbol{\pi}^E)$. Let $r_\omega(s, a, \mu)$ be an ω- parameterised reward function and $(\boldsymbol{\mu}^\omega, \boldsymbol{\pi}^\omega)$ denote the ERMFNE induced by ω. Then, recovering the underlying reward function reduces to tuning ω. The probability of a trajectory $\tau = \{(s_t, a_t)\}_{t=1}^{T}$ induced by ERMFNE with r_ω is defined by the following generative process:

$$p_\omega(\tau) = \mu_0(s_0) \cdot P(s_{t+1}|s_t, a_t, \mu_t^\omega) \cdot \prod_{t=0}^{T} \pi_t^\omega(a_t|s_t). \quad (6)$$

In the spirit of MaxEnt IRL, we should tune ω by maximising the likelihood of the observed trajectories concerning the distribution defined in Eq. (6). By Theorem 1, we can instead optimise the likelihood with respect to the distribution defined in Eq. (5) as a variational approximation:

$$\max_{\omega} \mathcal{L}(\omega) \triangleq \mathbb{E}_{\tau \sim (\boldsymbol{\mu}^E, \boldsymbol{\pi}^E)} \left[\mathcal{R}_{\omega}(\tau) + \sum_{t=0}^{T-1} \log P(s_{t+1}|s_t, a_t, \mu_t^{\omega}) \right] - \log \mathcal{Z}_{\omega},$$
(7)

where $\mathcal{Z}_{\omega}$ is the partition function of the distribution defined in Eq. (5), i.e., a summation over all feasible trajectories:

$$\mathcal{Z}_{\omega} \triangleq \sum_{\tau} e^{R_{\omega}(\tau)} \prod_{t=0}^{T-1} \log P(s_{t+1}|s_t, a_t, \mu_t^{\omega}).$$
(8)

However, directly optimising the likelihood objective in Eq. (7) is intractable because we cannot analytically derive the MF flow $\boldsymbol{\mu}^{\omega}$. This problem arises from the nature of MFGs that the policy and MF flow in ERMFNE are interdependent because $\tilde{\boldsymbol{\pi}}^{\star} = \tilde{\Psi}(\tilde{\boldsymbol{\mu}}^{\star})$ and in turn $\tilde{\boldsymbol{\mu}}^{\star} = \Phi(\tilde{\boldsymbol{\pi}}^{\star})$. This issue poses the main challenge for extending MaxEnt IRL to MFGs. Worse yet, the transition function P also depends on $\boldsymbol{\mu}^{\omega}$, posing an extra layer of complexity as the environment dynamics is generally unknown in practice.

While, notice that if we have access to an "oracle" (known a priori) MF flow that determines the shape of the observed MF flow $\boldsymbol{\mu}^E$, an individual would be decoupled from the population. Inspired by this fact, we sidestep this problem by substituting $\boldsymbol{\mu}^{\omega}$ with the empirical value of $\boldsymbol{\mu}^E$, denoted by $\hat{\boldsymbol{\mu}}^E \triangleq \{\mu_t^E\}_{t=0}^T$, estimated from observations $\mathcal{D}_E = \{\tau_j\}_{j=1}^M$ by calculating the occurrence frequencies of states:

$$\hat{\mu}_t^E(s) \triangleq \frac{1}{M} \sum_{j=1}^M \mathbb{1}_{\{s_{j,t}^i = s\}}.$$

Since the population consistency condition in ERMFNE guarantees that the state marginal distribution of a single agent matches the mean field at each time step, $\hat{\boldsymbol{\mu}}^E$ achieves an unbiased estimator of $\boldsymbol{\mu}^E$. Meanwhile, by substituting $\hat{\boldsymbol{\mu}}^E$ for $\boldsymbol{\mu}^{\omega}$, the transition function $P(s_t, a_t, \hat{\mu}_t^E)$ is decoupled the from the reward parameter ω as $\hat{\boldsymbol{\mu}}^E$ does not depend on ω, and henceforth being omitted in the likelihood function. Finally, with this substitution, we obtain a tractable version of the original MLE objective in Eq. (7):

$$\max_{\omega} \hat{\mathcal{L}}\left(\omega; \hat{\boldsymbol{\mu}}^E\right) \triangleq \mathbb{E}_{\tau \sim \mathcal{D}_E} \left[\hat{\mathcal{R}}_{\omega}(\tau)\right] - \log \hat{\mathcal{Z}}_{\omega},$$
(9)

which resembles the formulation of the MLE objective of MaxEnt IRL as given in Eq. (4). Here, $\hat{\mathcal{R}}_{\omega}(\tau) \triangleq \sum_{t=0}^{T-1} r_{\omega}(s_t, a_t, \hat{\mu}_t^E)$ and $\hat{\mathcal{Z}}_{\omega} \triangleq \sum_{\tau \in \mathcal{D}_E} e^{\hat{\mathcal{R}}_{\omega}(\tau)}$ denotes the simplified partition function in Eq. (8).

Statistically, Eq. (9) can be interpreted as that we use a likelihood function of a "mis-specified" model that treats the policy and MF flow as being independent and replaces the MF flow with its empirical value. In this manner, we estimate the optimal solution to the original MLE problem by maximising a simplified form of the actual likelihood function defined in Eq. (7). Although we sacrifice the accuracy for achieving tractability due to the estimation error of $\hat{\boldsymbol{\mu}}^E$, we show that Eq. (9) preserves the property of the asymptotic consistency, as $\hat{\boldsymbol{\mu}}^E$ converges almost surely to $\boldsymbol{\mu}^E$ as the number of samples tends to infinity due to the law of large numbers.

THEOREM 2. *Let the observed trajectories in $\mathcal{D}_E = \{\tau_j\}_{j=1}^M$ be independent and identically distributed and sampled from a unique ERMFNE induced by an unknown parameterised reward function. Suppose for all $s \in \mathcal{S}$, $a \in \mathcal{A}$ and $\mu \in \mathcal{P}(\mathcal{S})$, $r_{\omega}(s, a, \mu)$ is differentiable w.r.t. ω. Then, with probability 1 as the number of samples $M \to \infty$, the equation $\nabla_{\omega} \hat{\mathcal{L}}\left(\omega; \hat{\boldsymbol{\mu}}^E\right) = 0$ has a root $\hat{\omega}$ such that $\hat{\omega}$ is a maximiser of the likelihood objective $\mathcal{L}(\omega)$ in Eq. (7).*

PROOF. The gradient of $\hat{\mathcal{L}}$ concerning ω is given by:

$$\begin{aligned} \nabla_{\omega} \hat{\mathcal{L}}\left(\omega; \hat{\boldsymbol{\mu}}^E\right) &= \frac{1}{M} \sum_{j=1}^M \nabla_{\omega} \hat{\mathcal{R}}_{\omega}(\tau_j) - \nabla_{\omega} \log \hat{\mathcal{Z}}_{\omega} \\ &= \frac{1}{M} \sum_{j=1}^M \nabla_{\omega} \hat{\mathcal{R}}_{\omega}(\tau_j) - \frac{1}{\hat{\mathcal{Z}}_{\omega}} \nabla_{\omega} \hat{\mathcal{Z}}_{\omega} \\ &= \frac{1}{M} \sum_{j=1}^M \nabla_{\omega} \hat{\mathcal{R}}_{\omega}(\tau_j) - \sum_{j=1}^M \frac{e^{\hat{\mathcal{R}}_{\omega}(\tau_j)}}{\hat{\mathcal{Z}}_{\omega}} \nabla_{\omega} \hat{\mathcal{R}}_{\omega}(\tau_j). \end{aligned}$$
(10)

Let $\Pr_{\mathcal{D}_E}(\tau) \triangleq \frac{1}{M} \sum_{j=1}^M \mathbb{1}_{\{\tau_j = \tau\}}$ denote the empirical trajectory distribution, then Eq. (10) is equivalent to

$$\begin{aligned} \nabla_{\omega} \hat{\mathcal{L}}\left(\omega; \hat{\boldsymbol{\mu}}^E\right) &= \sum_{j=1}^M \Pr_{\mathcal{D}_E}(\tau_j) \nabla_{\omega} \hat{\mathcal{R}}_{\omega}(\tau_j) - \sum_{j=1}^M \frac{e^{\hat{\mathcal{R}}_{\omega}(\tau_j)}}{\hat{\mathcal{Z}}_{\omega}} \nabla_{\omega} \hat{\mathcal{R}}_{\omega}(\tau_j) \\ &= \sum_{j=1}^M \left(\Pr_{\mathcal{D}_E}(\tau_j) - \frac{e^{\hat{\mathcal{R}}_{\omega}(\tau_j)}}{\hat{\mathcal{Z}}_{\omega}} \right) \nabla_{\omega} \hat{\mathcal{R}}_{\omega}(\tau_j). \end{aligned}$$
(11)

When the number of samples $M \to \infty$, $\Pr_{\mathcal{D}_E}(\tau)$ tends to the true trajectory distribution $p(\tau)$ (see Eq. (5)) induced by ERMFNE. Meanwhile, according to the law of large numbers, $\hat{\boldsymbol{\mu}} \to \boldsymbol{\mu}^E$ with probability one as the number of samples $M \to \infty$. Let $\omega^{\star}$ be a maximiser of the likelihood objective in Eq. (7). Taking the limit as $M \to \infty$ and the optimality $\omega = \omega^{\star}$, we have:

$$\frac{e^{\mathcal{R}_{\omega^{\star}}(\tau_j)}}{\hat{\mathcal{Z}}_{\omega^{\star}}} = \frac{e^{\mathcal{R}_{\omega^{\star}}(\tau_j)}}{\sum_{j=1}^M e^{\mathcal{R}_{\omega^{\star}}(\tau_j)}} = \Pr(\tau_j) = \Pr_{\mathcal{D}_E}(\tau_j).$$

Therefore, the gradient in Eq. (11) will be zero. □

5 MEAN FIELD ADVERSARIAL IRL

Theorem 2 bridges the gap between optimising the original intractable MLE objective in Eq. (7) and the tractable empirical MLE objective in Eq. (9). However, as mentioned in Sec. 2.3, exactly computing the partition function $\hat{Z}_{\omega}$ is generally difficult. Similar to AIRL [9], we adopt *importance sampling* to estimate $\hat{Z}_{\omega}$ with *adaptive samplers*. Since policies are time-varying in MFGs, we use a set of T adaptive samplers $\boldsymbol{\pi}^{\theta} \triangleq (\pi^{\theta_0}, \pi^{\theta_1}, \ldots, \pi^{\theta_{T-1}})$, where each π^{θ_t} serves as the parameterised per-step policy.

Now, we are ready to present to our *Mean-Field Adversarial IRL* (MF-AIRL) framework, which trains a discriminator

$$\hat{D}_{\omega}(s_t, a_t) \triangleq \frac{e^{f_{\omega}(s_t, a_t, \hat{\mu}_t^E)}}{e^{f_{\omega}(s_t, a_t, \hat{\mu}_t^E)} + \pi^{\theta_t}(a_t|s_t)}$$

as

$$\max_{\omega} \mathbb{E}_{\tau \sim \mathcal{D}_E} \left[\sum_{t=0}^{T-1} \log \hat{D}_\omega(s_t, a_t) \right] + \mathbb{E}_{\tau \sim \boldsymbol{\pi}^{\theta}} \left[\sum_{t=0}^{T-1} \log(1 - \hat{D}_\omega(s_t, a_t)) \right],$$

(12)

and trains adaptive importance samplers $\boldsymbol{\pi}^{\theta}$ as

$$\max_{\boldsymbol{\theta}} \mathbb{E}_{\tau \sim \boldsymbol{\pi}^{\theta}} \left[\sum_{t=0}^{T-1} \log \hat{D}_\omega(s_t, a_t) - \log(1 - \hat{D}_\omega(s_t, a_t)) \right]$$

$$= \mathbb{E}_{\tau \sim \boldsymbol{\pi}^{\theta}} \left[\sum_{t=0}^{T-1} f_\omega(s_t, a_t, \hat{\mu}_t^E) - \log \pi^{\theta_t}(a_t | s_t) \right].$$

(13)

The update of policy parameters $\boldsymbol{\theta}$ is interleaved with the update of the reward parameter ω. Intuitively, tuning $\boldsymbol{\pi}^{\theta}$ can be viewed as a *policy optimisation* procedure, which is to find the ERMFNE policy induced by the current reward parameter in order to minimise the variance of importance sampling; f_ω is trained to estimate the reward function by distinguishing between the observed trajectories and those generated by the current adaptive samplers $\boldsymbol{\pi}^{\theta}$. We can use *backward induction* to train $\boldsymbol{\pi}^{\theta}$, i.e., tuning π^{θ_t} based on $\pi^{\theta_{t+1}}, \ldots, \pi^{\theta_{T-1}}$ that are already tuned. [4] At optimality, f_ω will approximate the underlying reward function for the observed ERMFNE and $\boldsymbol{\pi}^{\theta}$ will approximate the observed policy.

5.1 Reward Shaping in MFGs

As mentioned in Sec. 2.3, IRL faces reward ambiguity. This issue is called the effect of *reward shaping* [29], i.e., there is a class of reward transformations that induce the same set of optimal policies, where IRL cannot identify the ground-truth one without prior knowledge of environments. It is shown that for any state-only *potential function $h : \mathcal{S} \to \mathbb{R}$*, the reward transformation

$$r'(s_t, a_t, s_{t+1}) = r(s_t, a_t) + h(s_{t+1}) - h(s_t)$$

is the sufficient and necessary condition to ensure policy invariance for both MDPs and stochastic games [7]. We show that a similar idea can be extended to MFGs: for any potential function $g : \mathcal{S} \times \mathcal{P}(\mathcal{S}) \to \mathbb{R}$, the potential-based reward shaping can ensure the invariance of both ERMFNE and MFNE. The detailed justification and proofs are deferred until Appendix B.

To mitigate the effect of reward shaping, similar to AIRL [9], we assume that the parameterised reward function f_ω is in the following structure:

$$f_{\omega,\phi}(s_t, a_t, \mu_t, s_{t+1}, \mu_{t+1}) = r_\omega(s_t, a_t, \mu_t) + g_\phi(s_{t+1}, \mu_{t+1}) - g_\phi(s_t, \mu_t).$$

Here, g_ϕ is the ϕ-parameterised potential function for MFGs. To summarise, we present the pseudocode in Alg. 1.

6 RELATED WORK

We continue from the introduction to relate our work to the existing literature. MFGs were pioneered by [22, 23] in the continuous setting of stochastic differential games. The discrete MFG model was then proposed in [14], which was adopted in the learning setting. Recently, learning MFG has attracted significant attention [2], and

---

Algorithm 1 Mean-Field Adversarial IRL

1: **Input:** MFG with parameters $(\mathcal{S}, \mathcal{A}, P, \mu_0)$ and observed trajectories $\mathcal{D}_E = \{\tau_j\}_{j=1}^M$.
2: **Initialisation:** reward parameter ω, adaptive samplers $\boldsymbol{\theta}$ and potential function parameter ϕ.
3: Estimate the empirical expert MF flow $\hat{\boldsymbol{\mu}}^E$ from $\mathcal{D}_E$.
4: **for** each iteration **do**
5:     Sample a set of trajectories $\mathcal{D}_{\boldsymbol{\pi}} = \{\tau_j\}$ from $\boldsymbol{\pi}^{\theta}$ via $s_0 \sim \mu_0$, $a_t \sim \pi^{\theta_t}(\cdot|s_t)$, $s_{t+1} \sim P(\cdot|s_t, a_t, \mu_t)$.
6:     Sample subsets $\mathcal{X}_E, \mathcal{X}_{\boldsymbol{\pi}}$ from $\mathcal{D}_E, \mathcal{D}_{\boldsymbol{\pi}}$.
7:     Update ω, ϕ using $\mathcal{X}_E, \mathcal{X}_{\boldsymbol{\pi}}$ according to Eq. (12).
8:     **for** $t = T - 1, \ldots, 0$ **do**
9:        Update θ_t with respect to the reward estimate $r_\omega(s, a, \mu) + g_\phi(s, \mu)$ according to Eq. (13).
10:     **end for**
11: **end for**
12: **Output:** Learned reward function r_ω.

---

most methods are based on reinforcement learning [5, 16, 37, 42], *fictitious play* [2, 8, 40], or a combination of the two [32]. While these methods require a well-designed reward function that is challenging to hand-tune in practice. In contrast, our method recovers a reward function from the observed behaviour.

IRL was introduced by [30] in the single-agent setting. Early IRL methods were based on *margin optimisation* [33], which makes IRL ill-defined. *Maximum entropy* (MaxEnt) *IRL* [46, 47] addresses this issue by providing a probabilistic approach to find a most noncommittal reward function whose induced state-action trajectory distribution has the MaxEnt among those matching the reward expectation of the observed behaviour. However, it is only suitable for small and discrete problems since MaxEnt IRL requires iteratively solving an RL problem while tuning a reward function. *Adversarial IRL* (AIRL) [9] was later proposed, which scales MaxEnt IRL to high-dimensional or continuous domains. It implements a sampling-based approximation to MaxEnt IRL by drawing an analogy between *generative adversarial networks* [15] and MaxEnt IRL, thereby being able to partially solve each RL problem associated with reward tuning.

Another line of work extends ILR to the multi-agent setting, where the problem is cast to finding individual reward functions of stochastic games. They typically take a specific equilibrium concept, such as the conventional Nash equilibrium [10], logistic stochastic best response equilibrium [43] and equilibrium for the cooperative setting [28], and assume the equilibrium exists uniquely in order to guarantee the well-definedness (as we assumed in this paper). However, these methods scale poorly to large-scale scenarios due to the exponential growth of state-action spaces and agent interactions. By extending MaxEnt IRL to MFGs, our method realises an effective IRL framework for large-scale scenarios.

7 EXPERIMENTS

We seek to answer the following fundamental question via experiments: *Can MF-AIRL effectively and efficiently recover a suitable reward function of an MFG by observing bounded rational behaviour?*

---

[4] Since the reward at $t = T$ is 0, π_T always selects actions with ties broken arbitrarily to maximise the policy entropy.
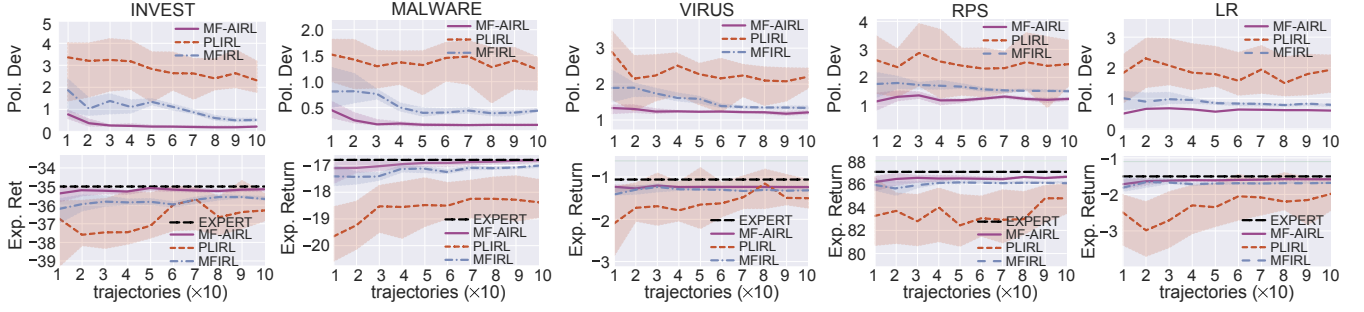
Figure 2: Results for numerical tasks. The line and shade are the median and variance over 10 independent runs.

To that end, we evaluate MF-AIRL on a series of simulated tasks motivated by real-world applications, where the observed behaviour is sampled from an ERMFNE.

7.1 Experimental Setup

7.1.1 Tasks. We adopt five MFG tasks: *investment in product quality* (INVEST for short), *malware spread* (MALWARE), *virus infection* (VIRUS), *Rock-Paper-Scissors* (RPS) and *Left-Right* (LR), which simulate a series of large-scale multi-agent scenarios in the contexts of marketing, virus propagation modelling and norm emergence. These tasks were originally studied in [5, 20, 21, 37, 39] and adapted by [4]. Detailed descriptions and settings are in Appendix C.

7.1.2 Baselines. We compare MF-AIRL against the two IRL methods above for MFGs: **(1)** The centralised method [41] relies on the reduction from MFG to MDP. Since it aims to recover the population's average rewards, we call it *population-level IRL* (**PLIRL**). Shown by [4], PLIRL is only compatible with socially optimal equilibria that maximise the population's average rewards; otherwise, it can result in biased reward inference. **(2)** The decentralised method, *Mean Field IRL* (**MFIRL**) [4], is based on margin optimisation, i.e., finding a reward function by minimising the margin (in terms of expected return) between the observed equilibrium and every other equilibrium; it is able to recover the ground-truth reward function with no bias, regardless of whether the observed equilibrium is socially optimal or not.

7.1.3 Performance Metrics. The quality of a learned reward function r_ω can be evaluated by the difference between its induced best-response policy to μ^E, denoted by π^ω, and π^E, because a best-response policy is unique under the entropy regularisation. We adopt the following two metrics that measure the difference between π^ω and π^E reflected in the statistical distance and the expected return, respectively:

(1) *Policy Deviation* (Pol. Dev). We use the cumulative KL- divergence,

$$\sum_{t=0}^{T-1} \sum_{s \in \mathcal{S}} D_{\mathrm{KL}}\big(\pi_t^E(\cdot|s) \parallel \pi_t^\omega(\cdot|s)\big),$$

to measure the statistical distance between two policies.

(2) *Expected return* (Exp. Ret). The difference between two expected returns $J(\mu^E, \pi^\omega)$ and $J(\mu^E, \pi^E)$ under the ground-truth reward function.

Table 1: Results for new environment dynamics. Mean and variance are taken across 10 independent runs.

Task	Metric	Algorithm		
		MF-AIRL	PLIRL	MFIRL
INVEST	Pol. Dev	**0.24** (0.02)	1.06 (0.21)	0.78 (0.18)
	Exp. Ret	**-35.19** (0.51)	-37.73 (2.76)	-35.92 (0.98)
MALWARE	Pol. Dev	**0.52** (0.01)	1.54 (1.20)	0.73 (0.14)
	Exp. Ret	**-18.49** (0.14)	-19.59 (0.29)	-18.82 (0.05)
VIRUS	Pol. Dev	**1.48** (0.01)	1.76 (0.18)	1.55 (0.03)
	Exp. Ret	**-1.71** (0.02)	-2.66 (0.14)	-2.16 (0.06)
RPS	Pol. Dev	**6.11** (0.46)	6.47 (0.98)	6.56 (0.82)
	Exp. Ret	**93.36** (2.51)	91.99 (0.44)	91.28 (2.15)
LR	Pol. Dev	**0.57** (0.04)	0.62 (0.22)	0.71 (0.07)
	Exp. Ret	**-1.70** (0.01)	-2.67 (1.01)	-1.93 (0.06)

Note: The Exp. Ret of the observed behaviour for five tasks are -35.87, -18.90, -1.24, 93.16 and -0.64, respectively.

7.1.4 Training Procedures. In all tasks, we have access to the ground-truth reward functions and environment dynamics, which allows us to numerically compute an ERMFNE through the fixed point iteration as introduced in Sec. 3. Unless specified otherwise, we set the entropy regularisation coefficient $\beta = 1$. After obtaining the ERMFNE, we sample trajectories from them, each with a length of 50 time steps, the same as the number used in [4, 36, 43]. We use one-hot encoding to represent states and actions. All three algorithms share the same neural network architecture as the reward model: two hidden layers of 64 leaky rectified linear units each. Implementation details are given in Appendix D.

7.2 Reward Recovery with Fixed Dynamics

The first experiment tests the capability of reward recovery with fixed environment dynamics. Results are depicted in Fig. 2. On all tasks, MF-AIRL achieves the closest performance to the observed behaviour with the same number of trajectories and the fastest convergence with the number of trajectories increases, suggesting that among all three algorithms, MF-AIRL is the most effective and efficient for bounded-rational agents. MFIRL shows larger deviations even if the number of trajectories is large. This may be because MFIRL takes MFNE as the solution concept, thereby lacking the

Table 2: Comparisons between MF-AIRL and MFIRL on varying entropy regularisation strength β.

Task	β	Metric	Algorithm		
			OBSERVED	MF-AIRL	MFIRL
INVEST	0	Pol. Dev	−	0.45 (0.03)	0.44 (**0.02**)
		Exp. Ret	-35.05	-35.92 (**0.68**)	-35.54 (2.55)
	0.1	Pol. Dev	−	**0.31** (0.02)	0.39 (0.04)
		Exp. Ret	-36.37	**-37.08** (0.71)	-37.40 (1.08)
MALWARE	0	Pol. Dev	−	0.39 (0.07)	0.38 (0.07)
		Exp. Ret	-18.06	-19.15 (**0.25**)	-18.52 (0.51)
	0.1	Pol. Dev	−	**0.41** (0.03)	0.50 (0.10)
		Exp. Ret	-19.36	**-19.84** (0.22)	-20.39 (0.69)
VIRUS	0	Pol. Dev	−	1.50 (**0.04**)	1.34 (0.09)
		Exp. Ret	-1.17	-2.55 (**0.04**)	-1.61 (0.17)
	0.1	Pol. Dev	−	**1.54** (0.01)	1.80 (0.07)
		Exp. Ret	-2.15	**-2.61** (0.06)	-2.98 (0.43)
RPS	0	Pol. Dev	−	9.71 (**0.24**)	9.36 (0.40)
		Exp. Ret	94.27	93.21 (**0.49**)	93.58 (2.51)
	0.1	Pol. Dev	−	**7.09** (0.54)	8.40 (0.40)
		Exp. Ret	91.43	**90.43** (3.09)	89.40 (0.96)
LR	0	Pol. Dev	−	0.45 (**0.07**)	0.37 (0.08)
		Exp. Ret	-0.52	-2.60 (**0.08**)	-1.70 (1.08)
	0.1	Pol. Dev	−	**0.68** (0.04)	0.70 (0.04)
		Exp. Ret	-0.64	**-0.81** (0.04)	-0.96 (0.71)

ability to tolerate suboptimal behaviours. PLIRL shows the largest deviation and variance. This is as expected because PLIRL is only suitable for socially optimal equilibria, while an ERMFNE is not socially optimal as it captures bounded rationality. Therefore, biased reward inferences occur when applying PLIRL to these tasks.

7.3 Policy Transfer across Varying Dynamics

The second experiment investigates the robustness against changing environment dynamics. We change the transition function (see Appendix C for details), recompute an ERMFNE (μ^E_{new}, π^E_{new}) induced by the ground-truth reward function, compute the best-response policy π^ω_{new} to μ^E_{new} under the learned reward function (trained with 100 demonstrated trajectories), and calculate two metrics again. Results are summarised in Tab. 1. Consistently, MF-AIRL outperforms two baselines on all tasks. We attribute the high robustness of MF-AIRL to the following reasons: (1) MF-AIRL uses a potential function to mitigate the effect of reward shaping while two baselines do not; (2) The issue of biased inference in PLIRL can be exacerbated by the changing dynamics, as is argued in [4]. To summarise, MF-AIRL can recover ground-truth reward functions with high robustness to changing dynamics.

7.4 Weak Entropy Regularisation

Suppose the entropy regularisation in ERMFNE is too strong. In that case, it becomes easy and trivial to perform MaxEnt IRL as the policy tends to select actions uniformly due to the maximum entropy principle. Our third experiment thus investigates the performance

under weak entropy regularisation. Since PLIRL is known to lead to biased reward inference if the demonstrated equilibrium is not socially optimal, we eliminate it here and only compare two decentralised methods. To weaken the entropy regularisation, we set the coefficient β in the observed ERMFNE to 0 and 0.1, respectively and sample 100 demonstrated trajectories from each. The environment dynamic is fixed. Results are summarised in Tab. 2. Note that an ERMFNE is recovered to an MFNE if $\beta = 0$; this complements the above two experiments where all trajectories are sampled from an ERMFNE with $\beta = 1$. It also enables a fair comparison between MF-AIRL and MFIRL as, technically, they are designed under two prescribed equilibrium concepts.

With trajectories sampled from an MFNE ($\beta = 0$), MF-AIRL shows a more significant deviation from the observed behaviour than MFIRL, but its variance is lower on average. This fact can be attributed to the reason that a best-response policy in MFNE does not exist uniquely, though MFIRL is unbiased under MFNE. In contrast, MF-AIRL can always recover a unique best-response policy from a learned reward function, though with bias under MFNE. This result again validates our argument that by taking MFNE as the solution concept, MFIRL fails to elicit a unique policy from the learned reward function. While, even with a positive yet small strength of entropy regularisation ($\beta = 0.1$), our MF-AIRL quickly outperforms MFIRL in terms of both accuracy and variance. This suggests that our MF-AIRL can handle imperfect behaviours resulting from bounded rationality, even with minor uncertainties.

8 CONCLUDING REMARKS

In this paper, we propose MF-AIRL, the first probabilistic IRL framework effective for MFGs with bounded-rational agents. We first extend MaxEnt IRL to MFGs based on the solution concept termed ERMFNE, which allows us to characterise uncertainties in observed behaviour using the maximum entropy principle. We then develop the practical MF-AIRL framework using an adversarial learning approach to solve MaxEnt IRL for MFGs efficiently. Experimental results on simulated tasks demonstrate the effectiveness and efficiency of MF-AIRL against existing IRL methods for MFGs.

We argue that MF-AIRL is worth following generalisations: **(1)** *Continuous states and actions*. The arguments in this paper still hold for continuous state-action spaces, except that some techniques (e.g., ϵ-net [16]) are needed to discretise a mean field because it turns to a probability density function if states are continuous. **(2)** *Infinite time horizon*. When the time horizon tends to infinity, the mean field is shown to converge almost surely to a constant limit, resulting in the *stationary MFNE* [37]. MF-AIRL is compatible with infinite time horizons because non-stationary equilibria recover stationary ones as special cases. **(3)** *Generalised mean fields*. Some work [16] generalises the mean field $\mu \in \mathcal{P}(\mathcal{S})$ to $(\mu, \alpha) \in \mathcal{P}(\mathcal{S} \times \mathcal{A})$ by additionally considering population's average action $\alpha \in \mathcal{P}(\mathcal{A})$. MF-AIRL is adaptive to generalised mean fields by incorporating the marginal distribution α in all mean field arguments. **(4)** *Heterogeneous agents*. A large-scale heterogeneous multi-agent system can be converted to a homogeneous system by considering the type of the agent as a component of states [38]. Our MF-AIRL is, therefore, compatible with heterogeneous agents.

REFERENCES

[1] Berkay Anahtarci, Can Deha Kariksiz, and Naci Saldi. 2020. Q-learning in regularized mean-field games. *arXiv preprint arXiv:2003.12151* (2020).

[2] Pierre Cardaliaguet and Saeed Hadikhanloo. 2017. Learning in mean field games: the fictitious play. *ESAIM: Control, Optimisation and Calculus of Variations* 23, 2 (2017), 569–591.

[3] 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.

[4] Yang Chen, Libo Zhang, Jiamou Liu, and Shuyue Hu. 2022. Individual-level inverse reinforcement learning for mean field games. In *Proceedings of the 21st International Conference on Autonomous Agents and Multi-agent Systems*.

[5] 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.

[6] Constantinos Daskalakis, Paul W Goldberg, and Christos H Papadimitriou. 2009. The complexity of computing a Nash equilibrium. *SIAM J. Comput.* 39, 1 (2009), 195–259.

[7] Sam Devlin and Daniel Kudenko. 2011. Theoretical considerations of potential-based reward shaping for multi-agent systems. In *The 10th International Conference on Autonomous Agents and Multiagent Systems*. ACM, 225–232.

[8] 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.

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

[10] Justin Fu, Andrea Tacchetti, Julien Perolat, and Yoram Bachrach. 2021. Evaluating Strategic Structures in Multi-Agent Inverse Reinforcement Learning. *Journal of Artificial Intelligence Research* 71 (2021), 925–951.

[11] Drew Fudenberg and Jean Tirole. 1991. *Game theory*. MIT press.

[12] Xavier Gabaix. 2014. A sparsity-based model of bounded rationality. *The Quarterly Journal of Economics* 129, 4 (2014), 1661–1710.

[13] Alessandro Garcia, Carlos Lucena, Franco Zambonelli, Andrea Omicini, and Jaelson Castro. 2002. Software Engineering for Large-Scale Multi-Agent Systems Research Issues and Practical Applications. In *Conference proceedings SELMAS*. Springer, 154.

[14] 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.

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems* 27 (2014).

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

[17] Xin Guo, Renyuan Xu, and Thaleia Zariphopoulou. 2020. Entropy regularization for mean field games with learning. *arXiv preprint arXiv:2010.00145* (2020).

[18] Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. 2017. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning*. PMLR, 1352–1361.

[19] Ronald M Harstad and Reinhard Selten. 2013. Bounded-rationality models: tasks to become intellectually competitive. *Journal of Economic Literature* 51, 2 (2013), 496–511.

[20] 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.

[21] 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.

[22] 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.

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

[24] Kiyeob Lee, Desik Rengarajan, Dileep Kalathil, and Srinivas Shakkottai. 2021. Reinforcement learning for mean field games with strategic complementarities. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 2458–2466.

[25] Yehuda John Levy and Eilon Solan. 2020. Stochastic games. *Complex Social and Behavioral Systems: Game Theory and Agent-Based Models* (2020), 229–250.

[26] Andreasa Morris-Martin, Marina De Vos, and Julian Padget. 2019. Norm emergence in multiagent systems: a viewpoint paper. *Autonomous Agents and Multi-Agent Systems* 33, 6 (2019), 706–749.

[27] Ulrich Munz, Antonis Papachristodoulou, and Frank Allgower. 2008. Delay-dependent rendezvous and flocking of large scale multi-agent systems with communication delays. In *2008 47th IEEE Conference on Decision and Control*. IEEE, 2038–2043.

[28] Sriraam Natarajan, Gautam Kunapuli, Kshitij Judah, Prasad Tadepalli, Kristian Kersting, and Jude Shavlik. 2010. Multi-agent inverse reinforcement learning. In *2010 ninth international conference on machine learning and applications*. IEEE, 395–400.

[29] Andrew Y Ng, Daishi Harada, and Stuart Russell. 1999. Policy invariance under reward transformations: Theory and application to reward shaping. In *ICML*, Vol. 99. 278–287.

[30] 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.

[31] Daniel Alexander Ortega and Pedro Alejandro Braun. 2011. Information, utility and bounded rationality. In *International Conference on Artificial General Intelligence*. Springer, 269–274.

[32] Sarah Perrin, Mathieu Laurière, Julien Pérolat, Romuald Élie, Matthieu Geist, and Olivier Pietquin. 2022. Generalization in mean field games by learning master policies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 9413–9421.

[33] 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.

[34] 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.

[35] Lloyd Shapley. 1964. Some topics in two-person games. *Advances in game theory* 52 (1964), 1–29.

[36] 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.

[37] 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.

[38] 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.

[39] 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.

[40] Qiaomin Xie, Zhuoran Yang, Zhaoran Wang, and Andreea Minca. 2021. Learning while playing in mean-field games: Convergence and optimality. In *International Conference on Machine Learning*. PMLR, 11436–11447.

[41] 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*.

[42] 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.

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

[44] Lantao Yu, Tianhe Yu, Chelsea Finn, and Stefano Ermon. 2019. Meta-inverse reinforcement learning with probabilistic context variables. *Advances in Neural Information Processing Systems* 32 (2019).

[45] Ming Zhou, Yong Chen, Ying Wen, Yaodong Yang, Yufeng Su, Weinan Zhang, Dell Zhang, and Jun Wang. 2019. Factorized q-learning for large-scale multi-agent systems. In *Proceedings of the First International Conference on Distributed Artificial Intelligence*. 1–7.

[46] 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.

[47] 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.

Appendices

A PROOF OF THEOREM 1

Our proof of Theorem 1 relies on the following lemma from [5], which shows that an energy-based model can characterise the policy in ERMFNE in terms of action values.

LEMMA A.1 ([5]). *Let* $(\mathcal{S}, \mathcal{A}, P, \mu_0, r)$ *be an MFG with the entropy regularisation and* $(\tilde{\mu}^\star, \tilde{\pi}^\star)$ *be the ERMFNE. Denote the* action value function *(i.e., cumulative future rewards of selecting an action in a state) of* (μ, π) *by*

$$Q^{\mu, \pi_{t+1:T-1}}(s_t, a_t, \mu_t) \triangleq r(s_t, a_t, \mu_t) + \mathbb{E}_{\pi_{t+1:T-1}} \left[\sum_{\ell=t+1}^{T-1} r(s_\ell, a_\ell, \mu_\ell) + \mathcal{H}(\pi_\ell(\cdot|s_\ell)) \right].$$

Then, the policy $\tilde{\pi}^\star$ *is in the form of*

$$\tilde{\pi}_t^\star(a_t|s_t) = \frac{e^{Q^{\tilde{\mu}^\star, \tilde{\pi}_{t+1:T-1}}(s_t, a_t, \tilde{\mu}_t^\star)}}{\sum_{a' \in \mathcal{A}} e^{Q^{\tilde{\mu}^\star, \tilde{\pi}_{t+1:T-1}}(s_t, a', \tilde{\mu}_t^\star)}}.$$

A.1 Proof of Theorem 1

PROOF. Let $(\tilde{\mu}^\star, \tilde{\pi}^\star)$ be the ERMFNE for a MFG $(\mathcal{S}, \mathcal{A}, p, \mu_0, r)$. For an arbitrary policy π, the probability of a trajectory $\tau = \{(s_t, a_t)\}_{t=0}^{T-1}$ induced by (μ, π) satisfies the following distribution:

$$p_1(\tau) = \mu_0(s_0) \cdot \prod_{t=0}^{T-1} P(s_{t+1}|s_t, a_t, \mu_t) \cdot \prod_{t=0}^{T} \pi_t(a_t|s_t). \tag{A1}$$

Recall our desired energy-based trajectory distribution formula is

$$p_2(\tau) \propto \mu_0(s_0) \cdot \prod_{t=0}^{T-1} P(s_{t+1}|s_t, a_t, \mu_t) \cdot e^{R(\tau)}.$$

Let D_{KL} denote the Kullback-Leibler (KL) divergence. We now show that the ERMFNE $(\tilde{\mu}^\star, \tilde{\pi}^\star)$ is the optimal solution to the following optimisation problem:

$$\min_{\mu, \pi} D_{\text{KL}}(p_1(\tau) \| p_2(\tau)) \ s.t. \ \mu = \Phi(\pi). \tag{A2}$$

The constraint enforces the condition of population consistency. Thus, fixing μ as $\tilde{\mu}^\star$, to show the ERMFNE $(\tilde{\mu}^\star, \tilde{\pi}^\star)$ is the optimal solution to Eq. (A2) is equivalent to show that $\tilde{\pi}^\star$ is the optimal solution to the following optimisation problem:

$$\min_{\pi} D_{\text{KL}}(p_1(\tau) \| p_2(\tau)) \ where \ \mu = \tilde{\mu}^\star. \tag{A3}$$

Our proof is in a manner of dynamic programming. We construct the basic case for step $T-1$, where Eq. (A3) holds according to the definition of the policy in ERMFNE. Then, for each $t < T - 1$, we construct the optimal policy for steps from t to $T-1$ based on the optimal policy that is already constructed from $t+1$ to $T-1$. We show that the constructed optimal policy that minimises the above KL divergence is consistent with $\tilde{\pi}^\star$ in ERMFNE. We next elaborate on the procedure of dynamic programming.

For simplicity, we omit the partition function for p_2 as it is a constant. Substituting $p_1(\tau)$ and $p_2(\tau)$ in Eq. (A3) with their definitions and roll out the KL-divergence, we obtain that maximising the KL-divergence between $p_1(\tau)$ and $p_2(\tau)$ is equivalent to the following optimisation problem

$$\max_{\pi} \mathbb{E}_{\tau \sim p_1} \left[\log \frac{p_2(\tau)}{p_1(\tau)} \right] = \mathbb{E}_{\tau \sim p_1} \left[\log \mu_0(s_0) + \sum_{t=0}^{T-1} \left(r(s_t, a_t, \tilde{\mu}_t^\star) + \log P(s_{t+1}|s_t, a_t, \tilde{\mu}_t^\star) \right) - \right.$$

$$\left. \log \mu_0(s_0) - \sum_{t=0}^{T-1} \left(\log \pi_t(a_t|s_t) + \log P(s_{t+1}|s_t, a_t, \tilde{\mu}_t^\star) \right) \right] \tag{A4}$$

$$= \mathbb{E}_{\tau \sim p_1} \left[\sum_{t=0}^{T-1} r(s_t, a_t, \tilde{\mu}_t^\star) - \log \pi_t(a_t|s_t) \right]$$

We maximise the objective in Eq. (A4) using a dynamic programming method.

Consider the terminal step $t = T$, since the reward at the terminal step is always 0, to maximise the entropy, the policy π_T chooses actions evenly, i.e, $\tilde{\pi}_T^\star(a|s) = 1/|\mathcal{A}|$ for any $a \in \mathcal{A}$ and $s \in \mathcal{S}$.

We then construct the base case that maximises π_{T-1}:

$$\max_{\pi_{T-1}} \mathbb{E}_{(s_{T-1},a_{T-1})\sim p_1} \left[r(s_{T-1}, a_{T-1}, \tilde{\mu}_{T-1}^\star) - \log \pi_{T-1}(a_{T-1}|s_{T-1}) \right]$$

$$= \mathbb{E}_{(s_{T-1},a_{T-1})\sim p_1} \left[-D_{\mathrm{KL}}\left(\pi_{T-1}(a_{T-1}|s_{T-1}) \,\middle\|\, \frac{e^{r(s_{T-1},a_{T-1},\tilde{\mu}_{T-1}^\star)}}{e^{V(s_{T-1},\tilde{\mu}_{T-1}^\star)}} \right) + V(s_{T-1},\tilde{\mu}_{T-1}^\star) \right], \tag{A5}$$

where we define

$$V(s_{T-1}, \tilde{\mu}_{T-1}^\star) \triangleq \log \sum_{a'\in\mathcal{A}} e^{r(s_{T-1},a',\tilde{\mu}_{T-1}^\star)}.$$

The optimal π_{T-1} for Eq. (A5) is

$$\tilde{\pi}_{T-1}^\star(a_{T-1}|s_{T-1}) = \frac{e^{r(s_{T-1},a_{T-1},\tilde{\mu}_{T-1}^\star)}}{e^{V(s_{T-1},\tilde{\mu}_{T-1}^\star)}} = \frac{e^{r(s_{T-1},a_{T-1},\tilde{\mu}_{T-1}^\star)}}{\sum_{a'\in\mathcal{A}} e^{r(s_{T-1},a',\tilde{\mu}_{T-1}^\star)}}, \tag{A6}$$

which coincides with the form given in Lemma A.1.

With the optimal policy given in Eq. (A6), the KL-divergence in Eq. (A5) attains 0 and Eq. (A5) attains the minimum $V(s_{T-1},\tilde{\mu}_{T-1}^\star)$.

Then recursively, we can show that for any step $t < T - 1$, π_t is the maximiser of the following optimisation problem:

$$\max_{\pi_t} \mathbb{E}_{(s_t,a_t)\sim p_1} \left[-D_{\mathrm{KL}}\left(\pi_t(a_t|s_t) \,\middle\|\, \frac{e^{Q^{\tilde{\mu}^\star,\tilde{\pi}_{t+1:T-1}^\star}(s_t,a_t,\tilde{\mu}_t^\star)}}{e^{V^{\tilde{\mu}^\star,\tilde{\pi}_{t+1:T-1}^\star}(s_t,\tilde{\mu}_t^\star)}} \right) + V^{\tilde{\mu}^\star,\tilde{\pi}_{t+1:T-1}^\star}(s_t,\tilde{\mu}_t^\star) \right], \tag{A7}$$

where

$$V^{\tilde{\mu}^\star,\tilde{\pi}_{t+1:T-1}^\star}(s_t,\tilde{\mu}_t^\star) \triangleq \log \sum_{a'\in\mathcal{A}} e^{Q^{\tilde{\mu}^\star,\tilde{\pi}_{t+1:T-1}^\star}(s_t,a_t,\tilde{\mu}_t^\star)}.$$

In fact, $V^{\mu,\pi_{t+1:T-1}}$ resembles the *soft value function* in *soft Q-learning* [18].

The optimal policy for Eq. (A7) is given by

$$\pi_t(a_t|s_t) = \frac{e^{Q^{\tilde{\mu}^\star,\tilde{\pi}_{t+1:T-1}^\star}(s_t,a_t,\tilde{\mu}_t^\star)}}{e^{V^{\tilde{\mu}^\star,\tilde{\pi}_{t+1:T-1}^\star}(s_t,\tilde{\mu}_t^\star)}} = \frac{e^{Q^{\tilde{\mu}^\star,\tilde{\pi}_{t+1:T-1}^\star}(s_t,a_t,\tilde{\mu}_t^\star)}}{\sum_{a\in\mathcal{A}} e^{Q^{\tilde{\mu}^\star,\tilde{\pi}_{t+1:T-1}^\star}(s_t,a,\tilde{\mu}_t^\star)}},$$

which coincides with the form given in Lemma A.1. □

B JUSTIFICATIONS FOR THE REWARD SHAPING IN MFGS

We show in the following theorem that for any potential function $g : \mathcal{S} \times \mathcal{P}(\mathcal{S}) \to \mathbb{R}$, the potential-based reward shaping can ensure the invariance of both ERMFNE and MFNE.

THEOREM 3. *Let any $\mathcal{S}, \mathcal{A}$ be given. We say $F : \mathcal{S} \times \mathcal{A} \times \mathcal{P}(\mathcal{S}) \times \mathcal{S} \times \mathcal{P}(\mathcal{S}) \to \mathbb{R}$ is a* potential-based reward shaping *for MFG if there exists a real-valued function $g : \mathcal{S} \times \mathcal{P}(\mathcal{S}) \to \mathbb{R}$ such that $F(s_t, a_t, \mu_t, s_{t+1}, \mu_{t+1}) = g(s_{t+1}, \mu_{t+1}) - g(s_t, \mu_t)$. Then, F is sufficient and necessary to guarantee the invariance of the set of MFNE and ERMFNE in the sense that:*

- **Sufficiency:** *Every MFNE or ERMFNE in the MFG $\mathcal{M}' = (\mathcal{S}, \mathcal{A}, P, \mu_0, r + F)$ is also a MFNE or ERMFNE in $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, \mu_0, r)$;*
- **Necessity:** *If F is not a potential-based reward shaping, then there exist an initial mean field μ_0, transition function P, horizon T, and reward function r such that no MFNE or ERMFNE in $\mathcal{M}'$ is an equilibrium in $\mathcal{M}$.*

PROOF. We first prove the sufficiency. First, we show the set of MFNE remains invariant under the potential-based reward shaping F. Let μ be an arbitrary MF flow. The optimal action function for the MFG induced by μ, denoted by $Q^\star$, fulfil the Bellman equation:

$$Q^\star(s_t, a_t) = r(s_t, a_t, \mu_t) + \mathbb{E}_{s_{t+1}\sim p}\left[\max_{a_{t+1}\in\mathcal{A}} Q^\star(s_{t+1}, a_{t+1}) \right].$$

Applying some simple algebraic manipulation gives:

$$Q^\star(s_t, a_t) - g(s_t, \mu_t)$$

$$= r(s_t, a_t, \mu_t) - g(s_t, \mu_t) + g(s_{t+1}, \mu_{t+1}) + \mathbb{E}_{s_{t+1}\sim P}\left[\max_{a_{t+1}\in\mathcal{A}} (Q^\star(s_{t+1}, a_{t+1}) - g(s_{t+1}, \mu_{t+1})) \right]$$

$$= r(s_t, a_t, \mu_t) + F(s_t, a_t, \mu_t, s_{t+1}, \mu_{t+1}) + \mathbb{E}_{s_{t+1}\sim P}\left[\max_{a_{t+1}\in\mathcal{A}} (Q^\star(s_{t+1}, a_{t+1}) - g(s_{t+1}, \mu_{t+1})) \right].$$

From here, we know that the above equation is exactly the Bellman equation induced by μ with the reward function $r + F$, and $Q^\star(s_t, a_t) - g(s_t, \mu_t)$ is the unique set of optimal Q values. Since $\arg\max_{a\in\mathcal{A}} Q^\star(s_t, a_t) - g(s_t, \mu_t) = \arg\max_{a\in\mathcal{A}} Q^\star(s_t, a_t)$, we have that for any fix a MF flow, any optimal policy of $\mathcal{M}'$ is also an optimal policy of $\mathcal{M}$. Hence, we finish the proof of the sufficiency of MFNE.

Next, we will show the sufficiency of ERMFNE. We write $\tilde{Q}^\star$ for the optimal action-value functions for the MFG with the entropy regularisation such that

$$\tilde{Q}^\star(s_t, a_t) = r(s_t, a_t, \mu_t) + \mathbb{E}_{s_{t+1} \sim p} \left[\sum_{a_{t+1} \in \mathcal{A}} \frac{e^{\tilde{Q}^\star(s_{t+1}, a_{t+1})}}{\sum_{a' \in \mathcal{A}} e^{\tilde{Q}^\star(s_t, a')}} \tilde{Q}^\star(s_{t+1}, a_{t+1}) \right].$$

Using the same manner as we show the invariance of MFNE, we have:

$$\tilde{Q}^\star(s_t, a_t) - g(s_t, \mu_t)$$
$$= r(s_t, a_t, \mu_t) - g(s_t, \mu_t) + g(s_{t+1}, \mu_{t+1})$$
$$+ \mathbb{E}_{s_{t+1} \sim P} \left[\frac{e^{\tilde{Q}^\star(s_{t+1}, a_{t+1}) - g(s_{t+1}, \mu_{t+1})}}{\sum_{a' \in \mathcal{A}} e^{\tilde{Q}^\star(s_t, a') - g(s_{t+1}, \mu_{t+1})}} (\tilde{Q}^\star(s_{t+1}, a_{t+1}) - g(s_{t+1}, \mu_{t+1})) \right]$$
$$= r(s_t, a_t, \mu_t) - g(s_t, \mu_t) + g(s_{t+1}, \mu_{t+1})$$
$$+ \mathbb{E}_{s_{t+1} \sim P} \left[\frac{e^{\tilde{Q}^\star(s_{t+1}, a_{t+1})}}{\sum_{a' \in \mathcal{A}} e^{\tilde{Q}^\star(s_t, a')}} (\tilde{Q}^\star(s_{t+1}, a_{t+1}) - g(s_{t+1}, \mu_{t+1})) \right].$$

Hence, we know that $\tilde{Q}^\star(s_t, a_t) - g(s_t, \mu_t)$ is the set of optimal action values induced by $\boldsymbol{\mu}$ under the reward function $r + F$. Thus, any optimal policy of $\mathcal{M}'$ is also an optimal policy of $\mathcal{M}$. Hence, we finish the proof of sufficiency for MFNE.

We next show the necessity by constructing a counter-example where a non-potential-based reward shaping can change the set of MFNE and ERMFNE. Consider the *Left-Right* problem [5], which is also used as a task in experiments. At each step, each agent is at a position (state) of either "left", "right" or "center", and can choose to move either "left" or "right", receives a reward according the current population density (mean field) at each position, and with probability one (dynamics) reaches "left" or "right". Once an agent leaves the "centre", she can never head back and can only be in either left or right thereafter. Formally, we configure the MFG as follows: $\mathcal{S} = \{C, L, R\}$, $\mathcal{A} = \mathcal{S} \setminus \{C\}$, initial mean field $\mu_0(C) = 1$ (i.e., all agents are at "center" initially) and the reward function

$$r(s, a, \mu_t) = -\mathbb{1}_{\{s=L\}} \cdot \mu_t(L) - \mathbb{1}_{\{s=R\}} \cdot \mu_t(R).$$

This reward setting means that each agent will incur a negative reward determined by the population density at her current position. The transition function is deterministic that directs an agent to the next state with probability one:

$$P(s_{t+1}|s_t, a_t, \mu_t) = \mathbb{1}_{\{s_{t+1}=a_t\}}.$$

This configuration is illustrated below.
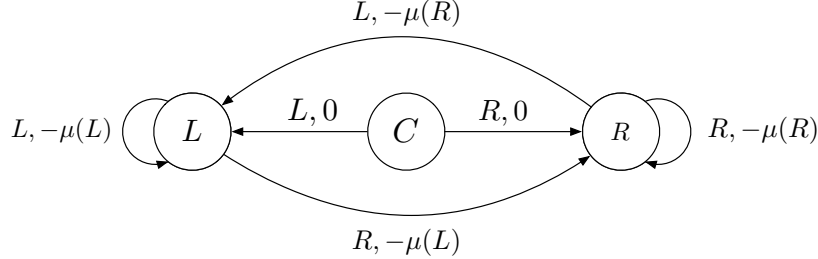
Now, we consider the case that the time horizon = 1.

Since all agents are at "center" initially, we have that $\mu_1^\star(L) = \pi_0^\star(L|C)$ and $\mu_1^\star(R) = \pi_0^\star(R|C)$. Therefore, we have that the expected return of each agent under MFNE is

$$-1 \cdot \pi_0^\star(L|C) \cdot \mu_1^\star(L) - 1 \cdot \pi_0^\star(R|C) \cdot \mu_1^\star(R)$$
$$= -(\pi_0^\star(L|C))^2 - (1 - \pi_0^\star(L|C))^2$$
$$= -\left(2(\pi_0^\star(L|C))^2 - 2\pi_0^\star(L|C) + 1\right).$$

Clearly, the expected return attains the maximum when $\pi_0^\star(L|C) = 1/2$. Therefore, any MFNE $(\boldsymbol{\mu}^\star, \boldsymbol{\pi}^\star)$ under the configuration above must fulfil $\pi_0^\star(R|C) = \pi_0^\star(L|C) = 1/2$ and $\pi_1^\star$ can be arbitrary. And clearly, there exists a unique ERMFNE $(\tilde{\boldsymbol{\mu}}^\star, \tilde{\boldsymbol{\pi}}^\star)$ such that any action at any state and any time step is chosen with probability 1/2. This result is also shown as a case study in [5].

Next, we change the reward function by adding an additional reward based on *actions* to the original reward function. We penalise the action "left" by a negative value -1, i.e.,

$$r'(s, a, \mu_t) = r(s, a, \mu_t) - \mathbb{1}_{\{a=L\}} = -\mathbb{1}_{\{s=L\}} \cdot \mu_t(L) - \mathbb{1}_{\{s=R\}} \cdot \mu_t(R) - \mathbb{1}_{\{a=L\}}.$$

This equivalent to that an action-based reward shaping function $F(s_t, a_t, \mu_t, s_{t+1}, \mu_{t+1}) = g(s_{t+1}, a_{t+1}, \mu_{t+1}) - g(s_t, a_t, \mu_t)$ is added to the original reward function where

$$g(s_t, a_t, \mu_t) = -\mathbb{1}_{\{a_t = L\}}.$$

The following diagram shows this new reward configuration.
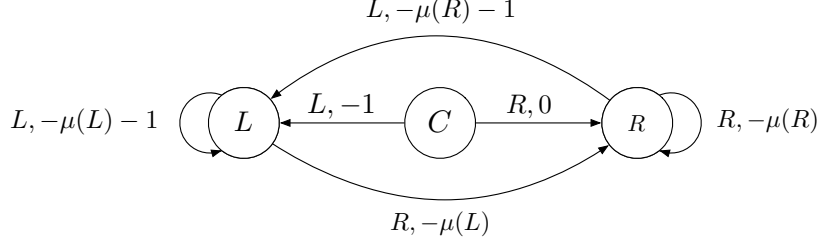
Now, let us investigate the form of MFNE and ERMFNE under this new reward configuration. We first show the set of MFNE induced by the new reward function is no longer the same as that induced by the original reward function. Consider the step $t = 1$ (the last step), since the reward of moving right is always higher than moving left by 1 and MFNE aims to maximise cumulative rewards, all agent will move right, i.e., $\pi_1^\star(R|L) = \pi_1^\star(R|R) = 1$. Hence, each agent earns a reward $-\mu_1^\star(L)$ if she is at "left" and $-\mu_1^\star(R)$ otherwise. Using the fact that $\mu_1^\star(L) = \pi_0^\star(L|C)$ and $\mu_1^\star(R) = \pi_0^\star(R|C)$, we have that the expected return of each agent under MFNE is

$$
\begin{aligned}
&- 1 \cdot \pi_0^\star(L|C) + 0 \cdot \pi_0^\star(R|C) - \mu_1^\star(L) \cdot \pi_0^\star(L|C) - \mu_1^\star(R) \cdot \pi_0^\star(R|C) \\
= &- \pi_0^\star(L|C) - (\pi_0^\star(L|C))^2 - (\pi_0^\star(R|C))^2 \\
= &- \pi_0^\star(L|C) - (\pi_0^\star(L|C))^2 - (1 - \pi_0^\star(L|C))^2 \\
= &- \left(2(\pi_0^\star(L|C))^2 - \pi_0^\star(L|C) + 1 \right).
\end{aligned}
$$

From here, we have that the expected return attains the maximum when $\pi_0^\star(L|C) = 1/4$, contradicting the MFNE induced by the original reward function.

Next, we show that the ERMFNE induced by the new reward function also changes. Again, consider the last step. According to the definition of ERMFNE, we have

$$\tilde{\pi}_1^\star(L|L) = \frac{e^{-\tilde{\mu}_1^\star(L)}}{e^{-\tilde{\mu}_1^\star(L)} + e^{-\tilde{\mu}_1^\star(L) - 1}}.$$

Therefore, $\tilde{\pi}_1^\star(L|L) = 1/2$ if and only if $\tilde{\mu}_1^\star(L) = \tilde{\mu}_1^\star(L) + 1$. A contradiction occurs. □

C DESCRIPTIONS FOR NUMERICAL TASKS

The detailed descriptions below are adapted from [4].

C.1 Investment in Product Quality

Model. This model is adapted from [39] and [37] that captures the investment decisions in a fragmented market with a large number of firms. Each firm produces the same kind of product. The state of a firm $s \in \mathcal{S} = \{0, 1, \ldots, 9\}$ denotes the product quality. At each step, each firm decides whether or not to invest in improving the quality of the product. Thus the action space is $\mathcal{A} = \{0, 1\}$. When a firm decides to invest, its product quality increases uniformly at random from its current value to the maximum value 9 if the average quality in the market for that product is below a particular threshold q. If this average quality value is above q, then the product quality gets only half of the improvement compared to the former case. This implies that when the average quality in the economy is below q, it is easier for each firm to improve its quality. When a firm does not invest, its product quality remains unchanged. Formally, the dynamics is given by:

$$s_{t+1} = \begin{cases} s_t + \lfloor \chi_t(10 - s_t) \rfloor, & \text{if } \langle \mu_t \rangle < q \text{ and } a_t = 1 \\ s_t + \lfloor \chi_t(10 - s_t)/2 \rfloor, & \text{if } \langle \mu_t \rangle \geq q \text{ and } a_t = 1 \ . \\ s_t, & \text{if } a_t = 0 \end{cases}$$

An agent incurs a cost due to its investment and earns a positive reward due to its own product quality and a negative reward due to the average product quality, which we denote by

$$\langle \mu_t \rangle \triangleq \sum_{s \in \mathcal{S}} s \cdot \mu_t(s). \tag{C8}$$

The final reward is given as:

$$r(s_t, a_t, \mu_t) = d \cdot s_t / 10 - c \cdot \langle \mu_t \rangle - \alpha \cdot a_t$$

C.1.1 Settings. We set $d = 0.3$, $c = 0.2$, $\alpha = 0.2$ and probability density f for χ_t as $U(0, 1)$. We set the threshold q to 4 and 5, respectively. The initial mean field μ_0 is set as a uniform distribution, i.e, $\mu_0(s) = 1/|\mathcal{S}|$ for all $s \in \mathcal{S}$.

C.2 Malware Spread

C.2.1 Model. The malware spread model is presented in [20, 21] and used as a simulated study for MFG in [37]. This model is representative of several problems with positive externalities, such as flu vaccination and economic models involving the entry and exit of firms. Here, we present a discrete version of this problem: Let $\mathcal{S} = \{0, 1, \ldots, 9\}$ denote the state space (level of infection), where $s = 0$ is the most healthy state and $s = 9$ is the least healthy state. The action space $\mathcal{A} = \{0, 1\}$, where $a = 0$ means DoNothing and $a = 1$ means Intervene. The dynamics is given by

$$s_{t+1} = \begin{cases} s_t + \lfloor \chi_t(10 - s_t) \rfloor, & \text{if } a_t = 0 \\ 0, & \text{if } a_t = 1 \end{cases},$$

where $\{\chi_t\}_{0 \le t \le T}$ is a $[0, 1]$-valued i.i.d. process with probability density f. The above dynamics means the DoNothing action makes the state deteriorate to a worse condition, while the Intervene action resets the state to the most healthy level. Rewards are coupled through the average health level of the population, i.e., $\langle \mu_t \rangle$ as defined in Eq.(C8). An agent incurs a cost $(k + \langle \mu_t \rangle)s_t$, which captures the risk of getting infected, and an additional cost of α for performing the Intervene action. The reward sums over all negative costs:

$$r(s_t, a_t, \mu_t) = -(k + \langle \mu_t \rangle)s_t/10 - \alpha \cdot a_t.$$

C.2.2 Settings. Following [37], we set $k = 0.2$, $\alpha = 0.5$, and the probability density f to the uniform distribution $U(0, 1)$ for the original dynamics. The initial mean field μ_0 is set as a uniform distribution.

C.3 Virus Infection

C.3.1 Model. This is a virus infection used as a case study in [5]. There is a large number of agents in a building. Each can choose between "social distancing" (D) or "going out" (U). If a "susceptible" (S) agent chooses social distancing, they may not become "infected" (I). Otherwise, an agent may become infected with a probability proportional to the number of infected agents. If infected, an agent will recover with a fixed chance every time step. Both social distancing and being infected have an associated negative reward. Formally, let $\mathcal{S} = \{S, I\}, \mathcal{A} = \{U, D\}, r(s, a, \mu_t) = -\mathbb{1}_{\{s=I\}} - 0.5 \cdot \mathbb{1}_{\{s=D\}}$. The transition probability is given by

$$P(s_{t+1} = S | s_t = I, \cdot, \cdot) = 0.3$$
$$P(s_{t+1} = I | s_t = S, a_t = U, \mu_t) = 0.9^2 \cdot \mu_t(I)$$
$$P(s_{t+1} = I | s_t = S, a_t = D, \cdot) = 0.$$

C.3.2 Settings. The initial mean field μ_0 is set as a uniform distribution.

C.4 Rock-Paper-Scissors

This model is adapted by [5] from the generalized non-zero-sum version of *Rock-Paper-Scissors* game [35]. Each agent can choose between "rock" (R), "paper" (P) and "scissors" (S), and obtains a reward proportional to double the number of beaten agents minus the number of agents beating the agent. Formally, let $\mathcal{S} = \mathcal{A} = \{R, P, S\}$, and for any $a \in \mathcal{A}, \mu_t \in \mathcal{P}(\mathcal{S})$:

$$r(R, a, \mu_t) = 2 \cdot \mu_t(S) - 1 \cdot \mu_t(P),$$
$$r(P, a, \mu_t) = 4 \cdot \mu_t(R) - 2 \cdot \mu_t(S),$$
$$r(S, a, \mu_t) = 6 \cdot \mu_t(P) - 3 \cdot \mu_t(R).$$

The transition function is deterministic: $p(s_{t+1}|s_t, a_t, \mu_t) = \mathbb{1}_{\{s_{t+1}=a_t\}}$.

C.4.1 Settings. The initial mean field μ_0 is set as a uniform distribution.

C.5 Left-Right

C.5.1 Model. This model is used in [5]. A group of agents makes sequential decisions to move "left" or "right". At each step, each agent is at a position (state) either "left", "right" or "center", and can choose to move either "left" or "right", receives a reward according the current population density (mean field) at each position, and with probability one (dynamics) they reach "left" or "right". Once an agent leaves "centre", she can never head back and can only be on the left or right thereafter. Formally, we configure the MFG as follows: $\mathcal{S} = \{C, L, R\}$, $\mathcal{A} = \mathcal{S} \setminus \{C\}$, the reward

$$r(s, a, \mu_t) = -\mathbb{1}_{\{s=L\}} \cdot \mu_t(L) - \mathbb{1}_{\{s=R\}} \cdot \mu_t(R).$$

This reward setting means each agent will incur a negative reward determined by the population density at her current position. The transition function is deterministic that directs an agent to the next state with probability one: $P(s_{t+1}|s_t, a_t, \mu_t) = \mathbb{1}_{\{s_{t+1}=a_t\}}$.

C.5.2 Settings. The initial mean field μ_0 is set as $\mu_0(L) = \mu_0(R) = 0.5$.

D IMPLEMENTATION DETAILS

The experimental settings below are adapted from [4].

D.1 Feature representations.

We use one-hot encoding to represent states and actions. Let $\{1, 2, \ldots, |\mathcal{S}|\}$ denote an enumeration of $\mathcal{S}$ and $\left[s_{[1]}, s_{[2]}, \ldots, s_{[|\mathcal{S}|]}\right]$ denote a vector of length $|\mathcal{S}|$, where each component stands for a state in $\mathcal{S}$. The state j is denoted by $\left[0, \ldots, 0, s_{[j]} = 1, 0, \ldots, 0\right]$. An action is represented in the same manner. A mean field μ is represented by a vector $\left[\mu(s_{[1]}), \mu(s_{[2]}), \ldots, \mu(s_{[|\mathcal{S}|]})\right]$, where $\mu(s_{[i]})$ denotes the proportion of agents that are in the ith state.

D.2 Reward Models and Adaptive Samplers.

The reward mode r_ω takes as input the concatenation of feature vectors of s, a and μ and outputs a scalar as the reward. We adopt the neural network (a four-layer perceptron) with the Adam optimiser and the Leaky ReLU activation function. The sizes of the two hidden layers are both 64. The learning rate is 10^{-4}. The adaptive sampler π^{θ_t} ($0 \le t < T$) takes as input the feature vector of a state and outputs a distribution over the action set. The neural network architecture for each adaptive sampler is a five-layer perceptron with the Adam optimiser and the Leaky ReLU activation function. The sizes of the first two hidden layers are both 64, the size of the third hidden layer is identical to the size of the action set, and the last layer is a softmax layer for generating a distribution over the action set.

D.3 Computation of ERMFNE.

In ERMFNE expert training, we repeat the fixed point iteration to compute the MF flow. We terminate at the ith iteration if the mean squared error over all steps and all state is below or equal to 10^{-10}, i.e.,

$$\frac{1}{(T-1)|\mathcal{S}|} \sum_{t=1}^{T-1} \sum_{s \in \mathcal{S}} \left(\mu_t^{(i)}(s) - \mu_t^{(i-1)}(s)\right)^2 \le 10^{-10}.$$