Coordinating the Crowd: Inducing Desirable Equilibria in Non-Cooperative Systems

David Mguni
PROWLER.io
Cambridge, UK
davidmg@prowler.io

Joel Jennings
PROWLER.io
Cambridge, UK
joel@prowler.io

Emilio Sison
MIT
Cambridge, MA, USA
esison@mit.edu

Sergio Valcarcel Macua
PROWLER.io
Cambridge, UK
sergio@prowler.io

Sofia Ceppi
PROWLER.io
Cambridge, UK
sofia@prowler.io

Enrique Munoz de Cote
PROWLER.io
Cambridge, UK
enrique@prowler.io

ABSTRACT

Many real-world systems such as taxi systems, traffic networks and smart grids involve self-interested actors that perform individual tasks in a shared environment. However, in such systems, the self-interested behaviour of agents produces welfare inefficient and globally suboptimal outcomes that are detrimental to all — common examples are congestion in traffic networks, demand spikes for resources in electricity grids and over-extraction of environmental resources such as fisheries. We propose an *incentive-design* method which modifies agents' rewards in non-cooperative multi-agent systems that results in independent, self-interested agents choosing actions that produce optimal system outcomes in strategic settings. Our framework combines multi-agent reinforcement learning to simulate (real-world) agent behaviour and black-box optimisation to determine the optimal modifications to the agents' rewards or *incentives* given some fixed budget that results in optimal system performance. By modifying the reward functions and generating agents' equilibrium responses in a sequence of offline Markov games, our method enables optimal incentive structures to be determined offline through iterative updates of the reward functions of a simulated game. Our theoretical results show that our method converges to reward modifications that induce system optimality. We demonstrate the applications of our framework by tackling a challenging problem in economics that involves thousands of selfish agents and a traffic congestion problem.

ACM Reference Format:
David Mguni, Joel Jennings, Emilio Sison, Sergio Valcarcel Macua, Sofia Ceppi, and Enrique Munoz de Cote. 2019. Coordinating the Crowd: Inducing Desirable Equilibria in Non-Cooperative Systems. In *Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019,* IFAAMAS, 9 pages.

1 INTRODUCTION

Complex systems such as traffic networks, smart grids and fleet networks involve autonomous agents that seek to perform individual tasks. One such example is a ride-sharing network such as an Uber fleet which involves many self-interested (freelance) drivers that use the same road network and have access to a common supply of customers. Other examples are road traffic networks used by commuters, electricity grids with households drawing from the network and smart grids. In each of these settings, agents utilise a shared resource to maximise their individual objectives.

Multi-agent systems (MASs) in which the set of agents act non-cooperatively to maximise their own interests are modelled by Markov games (MGs). In MGs, although each agent acts rationally i.e. to maximise its own interests, the lack of coordination produces stable outcomes or *Nash equilibria* (NE) that are vastly suboptimal from a system perspective and undermine firm efficiency [7].

In ride-sharing networks, drivers' self-interested behaviour and preference to locate at certain regions results in inefficient clustering that produces a distribution of taxis that does not match customer locations [16]. This results in a market inefficiency and prevents firms from maximising output. In electricity networks, excessive demand at specific periods leads to demand spikes that overwhelm supply; in traffic networks the actions of self-interested commuters leads to congestion resulting in poor network outcomes.

To alleviate these problems, network designers can employ incentives to modify the strategic behaviour of the self-interested agents. However, in an MAS, these incentives must be carefully calibrated to induce desirable outcomes from the *joint behaviour* of selfish actors in dynamic environments and often, with (budgetary) constraints on the size of incentives or penalties. Additionally, in settings such as smart grids and traffic networks, the design of incentives must also account for adjustments in the system state such as changes in customer demand for taxis; consequently, designing incentives is a formidable challenge [22].

Although in many MAS, the agents' reward functions are known (e.g. minimising commute time, firm profit maximisation) or a sufficiently accurate proxy can be constructed from data, designing incentives remains a challenge. This is due to the fact that changes to the agents' *joint* behaviour (and the resulting system outcomes) after modifications to their rewards is generally difficult to predict.

It is known that in many real-world MASs, human strategic interaction approximates NE strategies. Multi-agent reinforcement learning (MARL) is a powerful tool that enables computerised agents to *learn* strategic behaviour after repeated interactions in unknown systems - this enables MARL to serve as a useful tool to generate a proxy of outcomes in systems with human participants and simulate the behaviour of other computerised agents [9]. As with algorithmic methods in game theory, MARL does not offer

Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), N. Agmon, M. E. Taylor, E. Elkind, M. Veloso (eds.), May 13–17, 2019, Montreal, Canada.

a method of promoting efficient outcomes that maximise social welfare (e.g. minimise travel time in traffic networks) or optimise external objectives (e.g. firm profit) and frequently converges to poor system outcomes [25].

We propose a new technique to tackle the issue of undesirable outcomes in MASs. In our framework, an incentive designer (ID) modifies agents' reward functions in such a way that ensures convergence to efficient outcomes. This modification, as shown in one of our experiments, can represent a toll charge on a traffic network that induces even traffic flow leading to reduced congestion.

Using the known agents' intrinsic goals, our framework firstly uses MARL to learn the NE of simulated MAS and thus generate a proxy for real-world outcomes. This then allows us to model the induced changes in agents' behaviour given modifications to their rewards through incentives. The ID uses Bayesian optimisation in the *simulated environment* to determine the optimal modifications to the agents' rewards to be implemented in the real-world settings. The ID is not required to have a priori knowledge of the system performance metric but requires only the goal of the agents (e.g. arriving at work in the quickest time possible).

We concern ourselves with Markov potential games (MPGs) — a class of MGs that model settings in which agents compete for a common resource such as selfish routing games (transportation networks) [22], spectrum sharing (wireless communications) [32], oligopoly [26], electric power grids [12] and cloud computing [3].

We prove theoretical results that demonstrate that within MPGs, the ID's modifications to the game produces a continuous family of NE. Crucially, this allows the ID to use *black-box optimisation* techniques to find the reward modifications that induce desirable behaviour in the agents. Since the reward modifier influences the potential function - a function that is maximised by all agents' NE strategies, the method can be used to induce the desired behaviour in any number of agents. This is exemplified in one of our experiments in which we modify the rewards of 2,000 agents.

Contributions. i) We propose an algorithmic framework that determines how to modify the rewards (i.e. find incentives) in an MPG environment that lead to optimal system performance. **ii)** We show that the set of MGs with modified rewards are MPGs, and that the equilibrium set is continuous on the reward modifications. As we show, this allows us to prove existence of an optimal reward modifier. We prove convergence to the reward modifier that induces efficient NE and provide an approximation bound when the optimal reward modifier is estimated with a method that has low computational complexity. **iii)** We illustrate the framework in a set of experiments that tackle a logistic problem involving a system with 2,000 agents and a traffic network problem.

Related Work. Our work relates to mechanism design (MD) [19] and its dynamic and learning variants [27]. These incomplete information models analyse the problem of constructing a *mechanism* - a system of rewards and transfers among self-interested agents that have private information about their reward functions. MD seeks to incentivise truth-revealing announcements from the agents. In general, mechanisms that induce the desired agent behaviour for general reward functions cannot be achieved [23]. Therefore, in MD, agents' reward functions are (typically) limited to quasi-linear functions that are known up-front [19]. Our framework permits reward functions beyond quasi-linear functions.

This work relates to leader-follower games - sequential games in which a leader moves in advance of other agent(s) or *follower(s)*, who each select a best response strategy [28]. However, in leader-follower games, the leader cannot induce efficient outcomes i.e. maximise its own objective (e.g. ex. 98.1 in [20]) since the leader's reward is a function over a fixed joint action set.

Our work relates to reward shaping through which a reward is added with the aim of inducing convergence to a more desirable equilibrium [2]. The majority of the reward shaping literature is concerned with *potential based* reward shaping. Potential based reward shaping leaves the NE set unaltered and does not guarantee convergence to more efficient equilibria [6]. A number of papers handle non-potential based rewards shaping e.g. [21], however, such papers are limited to empirical analyses of specific normal form games e.g. the stag hunt game [21]. We tackle the MG case which adds considerable complexity as it requires a method of incentivising *sequences* of state-action pairs (trajectories) in a stochastic setting.

2 PRELIMINARIES

Let $\mathcal{N} \triangleq \{1, \dots, N\}$ denote the (possibly infinite) set of agents where $N \in \mathbb{N} \times \{\infty\}$. An MG is a tuple: $\mathcal{G} = \langle \mathcal{N}, (\gamma_i)_{i\in\mathcal{N}}, \mathcal{S}, (\mathcal{U}^i)_{i\in\mathcal{N}}, P, (R_i)_{i\in\mathcal{N}} \rangle$ which can be described as follows: at each time step $t = 1, 2, \dots T \in \mathbb{N} \times \{\infty\}$, the state of the system is given by $s \in \mathcal{S} \subseteq \mathbb{R}^p$ for some $p \in \mathbb{N}$. The game is equipped with an action set $\boldsymbol{\mathcal{U}} = \times_{i\in\mathcal{N}} \mathcal{U}^i$ – a Cartesian product of each agent's action set $\mathcal{U}^i$. Each set $\mathcal{U}^i$ is a compact, non-empty action set for each agent $i \in \mathcal{N}$. We define by $\mathcal{U}^{-i} = \times_{j\in\mathcal{N}\setminus\{i\}} \mathcal{U}^j$ - the Cartesian product of all agents' action sets except agent i. At each time step, the next state of the game is determined by a probability distribution $P : \mathcal{S} \times \boldsymbol{\mathcal{U}} \times \mathcal{S}$ so that $P(\cdot|s, \boldsymbol{u})$ gives the probability distribution over next states given a current state s when the agents take a joint action $\boldsymbol{u} \in \boldsymbol{\mathcal{U}}$. When the environment is at state s and the agents take action $\boldsymbol{u}$, each agent i receives a reward computed by a Lipschitz function $R_i : \mathcal{S} \times \mathcal{U}^i \times \mathcal{U}^{-i} \to \mathbb{R}$. The term $\gamma_i \in [0, 1[$ is each agent i's discount factor. Each agent has a stochastic policy $\pi^i : \mathcal{S} \times \mathcal{U}^i \to \mathbb{R}^+$ - a conditional distribution over the action set given the current state. Let Π^i be a non-empty set of stochastic policies over $\mathcal{S} \times \mathcal{U}^i$ such that $\pi^i \in \Pi^i$. We denote by $\boldsymbol{\Pi}$ the set of policies for all agents i.e. $\boldsymbol{\Pi} \triangleq \times_{i\in\mathcal{N}} \Pi^i$, where each π^i, and by $\Pi^{-i} \triangleq \times_{j\in\mathcal{N}\setminus\{i\}} \Pi^j$. For simplicity, we assume $\Pi^j = \Pi^i, \forall i \neq j$. The joint policy of all agents is denoted by $\boldsymbol{\pi} = \left(\pi^i\right)_{i\in\mathcal{N}} \in \boldsymbol{\Pi}$, while the joint policy of all but the i-th agent is denoted $\pi^{-i} = \left(\pi^j\right)_{j\in\mathcal{N}\setminus\{i\}}$. We will sometimes write $\boldsymbol{\pi} = \left(\pi^i, \pi^{-i}\right)$ for any $i \in \mathcal{N}$.

Each agent $i \in \mathcal{N}$ uses a *value function*, $v_i^{\boldsymbol{\pi}} : \mathcal{S} \times \boldsymbol{\Pi} \to \mathbb{R}$, as its objective function:

$$v_i^{(\pi_i, \pi_{-i})}(s) = \mathbb{E}\Big[\sum_{t=0}^{T} \gamma_i^t \, R_i(s_t, u_{i,t}, u_{-i,t}) \Big| \boldsymbol{u}_t \sim \boldsymbol{\pi}(\cdot|s_t), s_{t+1} \sim P(\cdot|s_t, \boldsymbol{u}_t), s_0 = s \Big],$$

where $\boldsymbol{u}_t = (u_{i,t}, u_{-i,t})$ is the joint action at time t.

We now give some essential definitions:

Definition 2.1. The policy $\pi^i \in \Pi^i$ is a best-response policy against $\pi^{-i} \in \Pi^{-i}$ if: $\pi^i \in \underset{\widetilde{\pi}^i \in \Pi^i}{\operatorname{argmax}} \, v_i^{(\widetilde{\pi}^i, \pi^{-i})}$.

A Markov-Nash equilibrium (M-NE) is the solution concept for MGs in which every agent plays a best-response against other agents. A M-NE is defined by the following:

Definition 2.2. A strategy $\boldsymbol{\pi} = \left(\pi^i\right)_{i\in\mathcal{N}} \in \boldsymbol{\Pi}$ is an **M-NE** if:

$$v_i^{(\pi^i,\pi^{-i})}(s) \geq v_i^{(\pi'^i,\pi^{-i})}(s), \quad \forall \pi'^i \in \Pi, \forall \pi^{-i} \in \Pi^{-i}, \forall s \in \mathcal{S}, \forall i \in \mathcal{N}. \tag{1}$$

The M-NE condition ensures no agent can improve their rewards by deviating unilaterally from their current strategy. We define $NE\{\mathcal{G}\}$ as the set of M-NE for the game $\mathcal{G}$.

Definition 2.3. An MG is called an *exact MPG* or an **MPG** for short, if there exists a function $\Phi : \mathcal{S} \times \boldsymbol{\Pi} \to \mathbb{R}$ such that:

$$v_i^{(\pi^i,\pi^{-i})}(s) - v_i^{(\pi'^i,\pi^{-i})}(s) = \Phi^{(\pi^i,\pi^{-i})}(s) - \Phi^{(\pi'^i,\pi^{-i})}(s) \quad \forall \pi'^i \in \Pi^i,\ \forall \pi^{-i} \in \Pi^{-i}, \forall s \in \mathcal{S},\ \forall i \in \mathcal{N} \tag{2}$$

Note that $\Phi^{\boldsymbol{\pi}}(s)$ gives the same value for all agents. We use $\mathcal{G}(\boldsymbol{w})$ to denote an MPG. In this paper, we focus exclusively on MPGs.

3 THE FRAMEWORK

We now describe how the ID modifies the MG played by the agents. The problem is arranged into a hierarchy in which the ID chooses the reward function of the game and a *simulated* subgame which models the joint behaviour of the agents. The goal of the ID is to modify the set of agent reward functions for the subgame that induces behaviour that maximises the ID's payoff. Crucially, in the MAS model, the agents are required to behave rationally and hence produce the responses of self-interested agents in an environment with the given reward functions. Using feedback from the simulated subgame in response to changes to the agents' reward functions, the ID can compute precisely the modifications to the agents' rewards that produce desirable equilibria among self-interested agents. The simulated environment avoids the need for costly acquisition of feedback data from real-world environments whilst ensuring the generated agent behaviour is consistent with real-world outcomes.

The MAS model consists of solving the Markov game $\mathcal{G}(\boldsymbol{w}) = \langle \mathcal{N}, (\gamma_i)_{i\in\mathcal{N}}, \mathcal{S}, (\mathcal{U}^i)_{i\in\mathcal{N}}, P, (R_{i,\boldsymbol{w}})_{i\in\mathcal{N}} \rangle$ i.e. finding $\boldsymbol{\pi} \in NE\{\mathcal{G}(\boldsymbol{w})\}$ where the parameter $\boldsymbol{w}$ is chosen by the ID. Now each agent $i \in \mathcal{N}$ has a value function $v_i^{\boldsymbol{\pi},\boldsymbol{w}} : \mathcal{S} \times \boldsymbol{\Pi} \times \boldsymbol{W} \to \mathbb{R}$ given by:

$$v_i^{\boldsymbol{\pi},\boldsymbol{w}}(s) = \mathbb{E}\Big[\sum_{t=0}^{T} \gamma_i^t R_{i,\boldsymbol{w}}(s_t, u_{i,t}, u_{-i,t}) \Big| \boldsymbol{u}_t \sim \boldsymbol{\pi}(\cdot|s_t), s_{t+1} \sim P(\cdot|s_t, \boldsymbol{u}_t), s_0 = s \Big]$$

The most natural alteration to an agent's reward function is for it to be modified additively by a **modifier function** $\Theta : \mathcal{S} \times \mathcal{U}^i \times \mathcal{U}^{-i} \times \boldsymbol{W} \to \mathbb{R}$ such that the agents' modified reward function becomes:

$$R_{i,\boldsymbol{w}}(s_t, u_{i,t}, u_{-i,t}) \triangleq R_i(s_t, u_{i,t}, u_{-i,t}) + \Theta(s_t, u_{i,t}, u_{-i,t}, \boldsymbol{w})$$

where $R_i : \mathcal{S} \times \mathcal{U}^i \times \mathcal{U}^{-i} \to \mathbb{R}$ is an **'intrinsic reward'** that cannot be modified by the ID. This function describes the agents' goals e.g. minimising travel time in their commute. We assume a sufficiently good proxy is available or the function is known to the ID. The function Θ is a modification to each agent's reward function and represents an incentive - for example, it may represent a toll charge in a traffic network or a surcharge in a smart grid which depends on factors such as time of day and the predicted available supply.

Note the modifier function includes cases for which $\Theta(\cdot, u_{-i,t},) = \Theta(\cdot, u'_{-i,t})$, $\forall u_{-i,t} \neq u'_{-i,t} \in \mathcal{U}^{-i}$ in which case the modifier function adds rewards that do not depend on actions other than those taken by agent i. We denote the **cumulative sum of incentives** by $\Psi(\boldsymbol{w}, \boldsymbol{\pi}) := \sum_{i\in\mathcal{N}} \sum_{t=0}^{T} \Theta(s_t, u_{i,t}, u_{-i,t}, \boldsymbol{w})$. **The incentive designer's problem** consists of a tuple $P_{\text{ID}} \triangleq \langle \boldsymbol{w}, R_{\text{ID}} \rangle$ where $\boldsymbol{w} \in \boldsymbol{W} \subset \mathbb{R}^l$ ($l \in \mathbb{N}$) is a set of vector of real-valued parameters over a space of parametric uniformly continuous functions and R_{ID} is the reward function for the ID. The ID's problem is to find Θ (i.e. the vector of parameters $\boldsymbol{w}$) that maximises the following:

$$J(\boldsymbol{w}, \boldsymbol{\pi}) := \mathbb{E}\Big[R_{\text{ID}}(\boldsymbol{w}, \boldsymbol{\pi}) - \lambda\Psi(\boldsymbol{w}, \boldsymbol{\pi})\Big],\ \lambda \in \mathbb{R} \tag{3}$$

whilst satisfying the M-NE condition which ensures that the agents play best-response policies. Thus the ID's problem is:

$$\underset{\boldsymbol{w}\in\boldsymbol{W}}{\text{maximise}}\ J(\boldsymbol{w}, \boldsymbol{\pi}) \text{ s.t. } v_i^{(\pi^i,\pi^{-i}),\boldsymbol{w}}(s) \geq v_i^{(\pi'^i,\pi^{-i}),\boldsymbol{w}}(s), \quad \forall i \in \mathcal{N}, \forall \pi'_i \in \Pi^i, \forall \pi^{-i} \in \Pi^{-i}, \forall s \in \mathcal{S}. \tag{4}$$

where J is a Lipschitz continuous function. The formulation describes numerous problems within economics and logistics including revenue management (e.g. ticket pricing), congestion management, and network design (e.g. tolling) [5]. The function Ψ can be interpreted as a system of wealth transfers for example, in the case of freelance taxis, Ψ represents rewards given to drivers for taking jobs at specific times and locations or surcharges to customers, and similarly for smart grid users at peak times. The following condition constrains the transfer of wealth to the set of agents:

Definition 3.1. The choice $\boldsymbol{w} \in \boldsymbol{W}$ is **weakly budget balanced** if there is no net transfer from the ID to the agents: $\Psi(\boldsymbol{w}, \boldsymbol{\pi}) \leq 0$.

We consider two main types of reward function for the ID, depending on the ID's goal:

1. **Trajectory targeted**: The ID's payoff is a function of the state trajectories produced by the agents' policies in the MG; i.e. is, $J(\boldsymbol{w}, \boldsymbol{\pi}) \triangleq \mathbb{E}\Big[R_{\text{ID}}(\boldsymbol{w}, X^{\boldsymbol{\pi}}, \zeta)\Big]$, where $X^{\boldsymbol{\pi}}$ is Markov chain induced by the policy profile $\boldsymbol{\pi} \in \boldsymbol{\Pi}$ in $\mathcal{G}(\boldsymbol{w})$ and ζ is an i.i.d. random variable which captures outcome noise. An example is taxi firm seeking to match the location of a set of freelance taxi drivers with (predicted) customer locations in some region. Here, the ID's objective could be given by a KL divergence between the distribution of taxis at every timestep, $D_t^a(\boldsymbol{w}, \boldsymbol{\pi})$, and the target distribution of demand, $D_t^\star$: $R_{\text{ID}}^{(\text{tra})} = \sum_{t=0}^{T} \text{KL}(D_t^a(\boldsymbol{w}, \boldsymbol{\pi}) \| D_t^\star)$. Other applications of trajectory targeted objectives are firms seeking to smoothen electricity consumption in smart grids through dynamic pricing [5] and modification of firm activity through taxation [17].

2. **Welfare targeted**: The ID's payoff is a function of the agents' joint rewards, that is, $J(\boldsymbol{w}, \boldsymbol{\pi}) \triangleq \mathbb{E}\Big[R_{\text{ID}}(\boldsymbol{w}, h(v_u^{\boldsymbol{\pi},\boldsymbol{w}}), \zeta)\Big]$, for some uniformly continuous function h and $v_a^{\boldsymbol{\pi},\boldsymbol{w}} \triangleq \left(v_i^{\boldsymbol{\pi},\boldsymbol{w}}\right)_{i\in\mathcal{N}}$. One example a traffic network manager that seeks to minimise travel time of all agents. In this cases, the ID is the sum of agents' negative costs (travel times) i.e.: $R_{\text{ID}}^{(\text{soc})} = \sum_{i\in\mathcal{N}} v_i^{\boldsymbol{\pi},\boldsymbol{w}}$, which results in the ID maximising social welfare. Similar examples are resource extraction and oligopoly intervention e.g. fishery problems using optimal taxation [26] in which the ID seeks to maximise firm welfare whilst

seeking to sustain a minimum amount of the resource and worst-case optimisation (maxmin) problems (i.e. by setting $h = -\mathbf{1}$).

The ID problem (4) is a bilevel optimisation problem (mathematical program with equilibrium constraints). Such problems are generally highly non-convex with unconnected feasible regions. For this reason, the problem is generally highly intractable using analytic methods but for simple cases (e.g. linear rewards) [4].

In the next section, we overcome these issues by expressing the NE constraint in terms of the potential function, and show that MARL methods can be applied to compute the set of NE for the MAS model, so that we can ensure feasibility for the ID problem without requiring closed analytic solutions. Crucially, this, as we show, allows us to compute the agents equilibrium policies to an MG the reward function of which, is chosen by the ID. We prove continuity properties of the MPG with respect to the ID's changes to the reward function which allows the ID to produce an iterative sequence of reward functions. We then give a constructive formulation that allows to prove convergence to such an optimal solution for the ID. Finally, we provide an approximation bound when the optimal reward modifier is approximated with a truncated power series. We proceed to explain the details.

4 THEORETICAL ANALYSIS

We now show that $\mathcal{G}(\boldsymbol{w})$ is an MPG, which enables $NE\{\mathcal{G}(\boldsymbol{w})\}$ to be described in terms of local maxima of function (not fixed points).

Proposition 4.1. *There exists a function $\Phi : \mathcal{S} \times \boldsymbol{\Pi} \to \mathbb{R}$ such that each agent's best-response strategy in $\mathcal{G}(\boldsymbol{w})$ maximises Φ.*

Prop. 4.1 reduces the problem of finding the M-NE for $\mathcal{G}(\boldsymbol{w})$ to a single optimal control problem as opposed to finding a fixed point solution which is considerably more difficult. However, it is necessary to show that the game produced after the ID alters the agents' rewards is still potential. Lemma 4.2 establishes that fact:

Lemma 4.2. *The game $\mathcal{G}(\boldsymbol{w})$ is an MPG.*

Proof. To prove the assertion we need to show that the transformation $R_i \to R_{i,\boldsymbol{w}}$ preserves the potential game property.

For any function $\Xi : \mathcal{S} \times \mathcal{U}^i \times \mathcal{U}^{-i}$ define $\Delta\Xi \triangleq \Xi_{i,\boldsymbol{w}}(s_t, u_{i,t}, u_{-i,t}) - \Xi_{i,\boldsymbol{w}}(s_t, u_{i,t}, u_{-i,t})$. We claim that there exists a function $\Phi^{\boldsymbol{\pi},\boldsymbol{w}}(s)$ s.th. $\Delta R_{i,\boldsymbol{w}}(s_t, u_{i,t}, u_{-i,t}) = \Phi^{\boldsymbol{\pi},\boldsymbol{w}}(s)$. This follows directly from the additive form of the reward function modification. Indeed, consider the function $\Phi^{\boldsymbol{\pi},\boldsymbol{w}}(s) \triangleq \Phi^{\boldsymbol{\pi}}(s) + \Theta(s, u^i, u^{-i}, \boldsymbol{w})(s)$. Since $\mathcal{G}_0$ is potential, by (3) and (4) we have that:

$$\begin{aligned}\Delta R_{i,\boldsymbol{w}}(s_t, u_{i,t}, u_{-i,t}) &= \Delta R_i(s_t, u_{i,t}, u_{-i,t}) + \Delta\Theta(s_t, u_{i,t}, u_{-i,t}, \boldsymbol{w}) \\ &= \Delta\Phi^{\boldsymbol{\pi},\boldsymbol{w}}(s). \end{aligned} \tag{5}$$

which completes the proof. □

Proposition 4.3. *S.2 R_{ID} is uniformly continuous in $\boldsymbol{w}$.*

The proof of the proposition is deferred to the appendix.

Corollary 4.4. *The following expression holds*

$$\left\{\operatorname*{argmax}_{\boldsymbol{\pi}\in\boldsymbol{\Pi}} \Phi^{\boldsymbol{\pi},\boldsymbol{w}}(s), \forall s \in \mathcal{S}\right\} \subseteq NE\{\mathcal{G}(\boldsymbol{w})\}. \tag{6}$$

Cor. 4.4 expresses that in playing their best-response strategies $\mathcal{G}(\boldsymbol{w})$, each agent inadvertently maximises $\Phi^{\boldsymbol{\pi},\boldsymbol{w}}$, so the function $\Phi^{\boldsymbol{\pi},\boldsymbol{w}}$ is a potential of $\mathcal{G}(\boldsymbol{w})$.

Having reduced the problem of finding $NE\{\mathcal{G}(\boldsymbol{w}\}$ to an optimal control problem, we now establish that the ID's problem is a *constrained optimisation problem*:

Theorem 4.5. *ID's problem is equivalent to:*

$$\operatorname*{maximise}_{\boldsymbol{w}\in\boldsymbol{W},\boldsymbol{\pi}\in\boldsymbol{\Pi}} J(\boldsymbol{w},\boldsymbol{\pi}) \;\text{ s.t. } \nabla_\pi \Phi^{\boldsymbol{\pi},\boldsymbol{w}} = 0, \;\; \nabla^2 \Phi^{\boldsymbol{\pi},\boldsymbol{w}} \preceq 0. \tag{7}$$

Sketch. The proof consists of the following components; proving that $\Phi \in \mathcal{C}^1$ and that ID's problem can be rewritten as a constrained optimisation problem and the set of constraints of the problem are expressed by (7). By *Rademacher's lemma* we have that if Φ is Lipschitz continuous on some open subset of its domain then Φ is differentiable almost everywhere (in that set). Since Φ is defined over $\mathcal{S} \subseteq \mathbb{R}^p$, we can construct an open subset for which Rademacher's lemma holds. To deduce the remainder, we note that by Corollary 4.4, $NE\{\mathcal{G}(\boldsymbol{w})\}$ coincide with the set of the local maxima of Φ. The result then follows by noting that conditions (7) are first and second order conditions for local maxima of Φ. □

Theorem 4.5 establishes that the ID's problem reduces to a constrained optimisation problem where the feasibility set is given by the set of points that are local maxima of Φ. In the next section, we show that we can apply MARL to constrain the set of points in $\boldsymbol{W}$ to lie within the feasibility set.

We now prove that $NE\{\mathcal{G}(\boldsymbol{w})\}$ is continuous on $\boldsymbol{w}$ — this enables the ID to generate an iterative sequence of games and permits use of black-box optimisation to solve the ID's problem. We firstly study the effect of modifying $\boldsymbol{w}$ on $NE\{\mathcal{G}(\boldsymbol{w})\}$. To establish a formal notion of continuity of $NE\{\mathcal{G}(\boldsymbol{w})\}$ w.r.t $\boldsymbol{w}$, we introduce *essentiality*:

Definition 4.6. Given metric space $\mathbf{X}$, let $B_\alpha(\boldsymbol{x}) \triangleq \{\boldsymbol{y} \in \mathbf{X} : \|\boldsymbol{x} - \boldsymbol{y}\| < \alpha\}$ denote the open ball with radius $\alpha > 0$ around $\boldsymbol{x} \in \mathbf{X}$. Then $\boldsymbol{x} \in NE\{\mathcal{G}(\boldsymbol{w})\}$ is **essential** in $\boldsymbol{w}$ if for any $\epsilon > 0$, $\exists\delta > 0$: $\boldsymbol{w}' \in B_\epsilon(\boldsymbol{w}) \implies \boldsymbol{x}' \in B_\delta(\boldsymbol{x})$, for any $\boldsymbol{x}' \in NE\{\mathcal{G}(\boldsymbol{w}')\}$.

The following results establish the continuity in ID's reward under changes in $\boldsymbol{w}$ which underpin the existence of a solution for ID's problem and a method for computing the solution. We begin by demonstrating that small changes in ID's action lead to small changes in the game, that is, the game itself is continuous in $\boldsymbol{w}$.

Proposition 4.7. *$NE\{\mathcal{G}(\boldsymbol{w})\}$ is an essential set in $\boldsymbol{w}$.*

Proof. We begin the proof by proving that the value function for each agent $i \in \mathcal{N}$ is Lipschitz continuous w.r.t. $\boldsymbol{w}$:

$$\begin{aligned}\left|v_i^{(\pi_i^\star,\pi_{-i}^\star),\boldsymbol{w}}(s_t) - v_i^{(\pi_i^\star,\pi_{-i}^\star),\boldsymbol{w}'}(s_t)\right| &= \Big|\mathbb{E}\Big[\max_{\pi\in\Pi}[R_i(s_t,u_{i,t},u_{-i,t},\boldsymbol{w}) \\ &\quad + \gamma\sum_{s'\in S} p(s'|s,\boldsymbol{a}) v_i^{(\pi_i^\star,\pi_{-i}^\star)}(s',u_{i,t},u_{-i,t},\boldsymbol{w})\Big|\boldsymbol{u}_t \sim \boldsymbol{\pi}(\cdot|s_t)\Big]\Big| \\ &\quad - \Big|\mathbb{E}\Big[\max_{\pi\in\Pi}[R_i(s_t,u_{i,t},u_{-i,t},\boldsymbol{w}') \\ &\quad + \gamma\sum_{s'\in S} P(s'|s,\boldsymbol{a}) v_i^{(\pi_i^\star,\pi_{-i}^\star)}(s',u_{i,t},u_{-i,t},\boldsymbol{w}')\Big|\boldsymbol{u}_t \sim \boldsymbol{\pi}(\cdot|s_t)\Big]\Big| \\ \leq \max_{\boldsymbol{\pi}\in\boldsymbol{\Pi}} &\Big|\mathbb{E}\left[R_i(s_t,u_{i,t},u_{-i,t},\boldsymbol{w}) - R_i(s_t,u_{i,t},u_{-i,t},\boldsymbol{w}')\Big|\boldsymbol{u}_t \sim \boldsymbol{\pi}(\cdot|s_t)\right]\Big| \\ &\quad + \gamma\sum_{s'\in S} P(s'|s,\boldsymbol{u})\Big|\mathbb{E}_{\boldsymbol{\pi}}\left[v_i^{(\pi_i^\star,\pi_{-i}^\star)}(s',u_{i,t},u_{-i,t},\boldsymbol{w})\right. \\ &\quad \left. - v_i^{(\pi_i^\star,\pi_{-i}^\star)}(s',u_{i,t},u_{-i,t},\boldsymbol{w})\Big|\boldsymbol{u}_t \sim \boldsymbol{\pi}(\cdot|s_t)\right]\Big|. \end{aligned} \tag{8}$$

Recall that $\gamma < 1$, we therefore find that

$$\left|v_i^{(\pi_i^\star,\pi_{-i}^\star),\boldsymbol{w}}(s_t) - v_i^{(\pi_i^\star,\pi_{-i}^\star),\boldsymbol{w}'}(s_t)\right| \leq (1-\gamma)^{-1}\max_{\boldsymbol{\pi}\in\Pi}\left|\mathbb{E}_{\boldsymbol{\pi}}\left[R_i(s_t,u_{i,t},u_{-i,t},\boldsymbol{w}) - R_i(s_t,u_{i,t},u_{-i,t},\boldsymbol{w}')\middle|\boldsymbol{u}_t\sim\boldsymbol{\pi}(\cdot|s_t)\right]\right| \leq c\|\boldsymbol{w}-\boldsymbol{w}'\|,$$

where $c \triangleq L_{R_i}(1+\gamma)^{-1}$ and $L_{R_i} > 0$ is the Lipschitz constant for the function R_i, which proves that the function $v_i^{\cdot,\boldsymbol{w}}$ is Lipschitzian in $\boldsymbol{w}$. Hence, *a fortiori*, the function $v_i^{\cdot,\boldsymbol{w}}$ is uniformly continuous w.r.t. $\boldsymbol{w}$ hence we have that $\forall\epsilon>0\exists\delta>0$ s.th $\|\boldsymbol{w}-\boldsymbol{w}'\|<\epsilon \implies |v_i^{(\pi_i^\star,\pi_{-i}^\star),\boldsymbol{w}}(\cdot) - v_i^{(\pi_i^\star,\pi_{-i}^\star),\boldsymbol{w}'}(\cdot)| < \delta$. The remainder of the proof follows using the potential property (Definition 2) and Lemma A.1. □

To solve the ID's problem, it is necessary to establish the existence of an optimal reward modifier $\boldsymbol{w}^\star \in \boldsymbol{W}$ that solves ID's problem, i.e. a $\boldsymbol{w}^\star \in \arg\max J(\boldsymbol{w},\boldsymbol{\pi})$ which induces an efficient NE.

Theorem 4.8. *For $\mathcal{G}(\boldsymbol{w})$ there exists a value $\boldsymbol{w}^\star \in \boldsymbol{W}$ that maximises ID's reward function R_{ID}.*

Sketch. First we note that by Prop. 4.3, we note that the function J is Lipschitz continuous w.r.t. the variable $\boldsymbol{w}$. The proof then follows from the compactness of $\boldsymbol{\Pi}, \boldsymbol{W}$ and the continuity of J, indeed since J is a continuous map from compact sets by the properties of continuous maps we can deduce that the image of J is compact. Moreover, by extreme value theorem we deduce the existence of a maximum value of $\boldsymbol{w}$ within the set $\boldsymbol{W}$. □

Previous results hold for an arbitrarily expressive modifier function Θ. In practice, it is computationally efficient to express Θ using a representation with few parameters. The following bounds ID's loss when Θ is approximated by a truncated power series:

Theorem 4.9. *Let $\boldsymbol{w}^\epsilon(n) \in \boldsymbol{W}$ approximate solution to ID's problem for $\mathcal{G}(\boldsymbol{w})$ which is generated by an $n-$order series expansion, define ID's approximation loss by $\mathcal{L} \triangleq J(\boldsymbol{w}^\star,\boldsymbol{\pi}) - J(\boldsymbol{w}^\epsilon(n),\boldsymbol{\pi})$, then $\mathcal{L}$ has the following bound: $\mathcal{L} \leq \max\left|D^{N+1}J(\boldsymbol{w}',\boldsymbol{\pi}(\boldsymbol{w}'))\right|$.*

The solution $\boldsymbol{w}^\star$ is closely approximated by a truncated series expansion (other expansions e.g. neural networks are possible) reducing the number of parameters to be computed.

5 PRESERVING THE NASH EQUILIBRIA.

We can modify the framework to tackle the case in which the ID modifies the rewards to maximise some efficiency criterion subject to the condition that M-NE set of the game is preserved. Inducing convergence to the highest welfare equilibria within a fixed M-NE set is known as *equilibrium selection* (ES) and represents a major challenge in GT and MARL [10].The ID framework can be used to address ES within the context of MPGs.

Let $\mu_k(\boldsymbol{W}) \triangleq \{\boldsymbol{w}\in\boldsymbol{W} : \Psi(\cdot,\boldsymbol{w}) = k | k\in\mathbb{R}\}$. Since $\mathcal{G}(\mu_k(\boldsymbol{W}))$ is just the MG in which the agents' rewards are modified by at most a constant, it is straightforward to deduce that the NE set is preserved. A particular case of this is potential-based reward shaping in which each agent's value function is given by the following:

$$v_i^{\boldsymbol{\pi},\boldsymbol{w}}(s) = \mathbb{E}\left[\sum_{t=0}^{T}\gamma_i^{t-1}\left\{\gamma R_i(s_t,u_{i,t},u_{-i,t}) + F(s_t,u_{i,t},u_{-i,t},\boldsymbol{w})\right\}\right]$$

where $F(s_t,u_{i,t},u_{-i,t},\boldsymbol{w}) := \gamma_i\Theta(s_t,u_{i,t},u_{-i,t},\boldsymbol{w}) - \Theta(s_{t-1},u_{i,t-1},u_{-i,t-1},\boldsymbol{w})$. When $T=\infty$, since $0\leq\gamma_i<1$ the potential-based modifier produces the telescoping sum: $\sum_{t\geq 0}\Delta\Theta(s_t,u_{i,t},u_{-i,t},\boldsymbol{w}) = \Theta(s_0,u_{i,0},u_{-i,0},\boldsymbol{w}) \equiv c$ where c is some constant independent of the agents' policies. We therefore see that $NE\{\mathcal{G}(\boldsymbol{w})\}$ is preserved since from Definition 2, we can see that the addition of constants to the agents' reward functions preserves the M-NE condition. The general case does not restrict to potential based reward shaping. Moreover, unlike current potential-based shaping methods for which the function F is fixed and may *lead to convergence to less desirable equilibria* [6], now the function $F(\cdot,\boldsymbol{w})$ is determined as a solution to the ID's problem in $\boldsymbol{w}$.

By similar reasoning as previous, we deduce the following:

$$\underset{\boldsymbol{w}\in\boldsymbol{W},\boldsymbol{\pi}\in\boldsymbol{\Pi}}{\text{maximise}}\ J(\boldsymbol{w},\boldsymbol{\pi}) \text{ s.t. } \nabla_\pi\Phi^{\boldsymbol{\pi},\mu_0(\boldsymbol{W})} = 0,\ \ \nabla_\pi^2\Phi^{\boldsymbol{\pi},\mu_0(\boldsymbol{W})} \leq 0. \quad (9)$$

Hence, the M-NE constraint is defined over the M-NE set before the ID alters the game. The formulation ensures that the agents' rewards are modified in a way that the agents play efficient policies, the constraint ensures the joint policy remains in the original NE set. We now formalise the description of efficiency.

Definition 5.1. The strategy profile $\boldsymbol{\pi}\in\boldsymbol{\Pi}$ is a **welfare optimal strategy profile** of $\mathcal{G}(\boldsymbol{w})$ if: $\sum_{i\in\mathcal{N}} v_i^{\pi^i,\pi^{-i},\boldsymbol{w}} \geq \sum_{i\in\mathcal{N}} v_i^{\pi'^i,\pi^{-i},\boldsymbol{w}}$.

Definition 5.2. For a given $\boldsymbol{w}\in\boldsymbol{W}$, $\boldsymbol{\pi}\in\boldsymbol{\Pi}$ is a **Pareto efficient (PE)** strategy profile of $\mathcal{G}(\boldsymbol{w})$ if: *i)* $v_i^{\pi^i,\pi^{-i},\boldsymbol{w}} \geq v_i^{\pi'^i,\pi'^{-i},\boldsymbol{w}}, \forall i\in\mathcal{N}$, *ii)* $v_i^{\pi^i,\pi^{-i},\boldsymbol{w}} > v_i^{\pi'^i,\pi^{-i},\boldsymbol{w}}$ for some $i\in\mathcal{N}$.

PE implies that no agent increases their reward whenever some other strategy profile $\boldsymbol{\pi}'\in\boldsymbol{\Pi}$ is played and, at least one agent is strictly best off under $\boldsymbol{\pi}$ so that all agents prefer the PE outcome. PE is a criterion for a welfare maximising ID. We say that strategy profile $\boldsymbol{\pi}$ is **payoff dominant** if $\boldsymbol{\pi}\in NE\{\mathcal{G}(\boldsymbol{w})\}$ and $\boldsymbol{\pi}$ is PE.

Proposition 5.3. *Let $\boldsymbol{w}\in\boldsymbol{W}$ be a solution to ID's ES problem, then $\exists\,\boldsymbol{\pi}\in NE\{\mathcal{G}_0\}$ which is a payoff dominant policy profile of $\mathcal{G}_0$.*

The issue of how to compute $\boldsymbol{w}^\star$ remains; we now describe its computation using black-box optimisation and MARL.

6 SOLUTION METHOD

The method uses MARL to generate a model of the strategic (equilibrium) behaviour among the agents for a given value of $\boldsymbol{w}_k$ that determines the modification of the agents' rewards. The value of value of $\boldsymbol{w}_k$ is then updated. The specifics are as follows: the function R_{ID}, its gradient, the function h and each $v_i^{\boldsymbol{\pi},\boldsymbol{w}}$ are all unknown to the ID (however a suitable proxy for the intrinsic reward, R_i is known), who solely observes its realised rewards for each candidate $\boldsymbol{w}$ which suggests a black-box optimisation method. The unknown payoff, J, is treated as a random function with some prior belief over the space of functions. After observing the value of $J(\boldsymbol{w}_k,\boldsymbol{\pi})$ for some $\boldsymbol{w}_k\in\boldsymbol{W}$, the belief is updated to form a posterior distribution which is used to construct an acquisition function (e.g., expected improvement) that indicates which parameter $\boldsymbol{w}_{k+1}$ should be evaluated next, guiding exploration over $\boldsymbol{W}$. We use MARL to solve the game $\mathcal{G}(\boldsymbol{w})$ allowing the agents (of the simulated game) to observe only their individual (modified) rewards after their joint policy $\boldsymbol{\pi}$ is played. The agents sample trajectories of experience

tuples $(s_t, \boldsymbol{u}_t, (R_{i,\boldsymbol{w}_k}(s_t, \boldsymbol{u}_t))_{i\in\mathcal{N}}, s_{t+1})$, which are used to estimate the joint value function, $v_i^{\boldsymbol{\pi},\boldsymbol{w}}$. Then, they update their policies by performing stochastic gradient ascent.

The optimisation objective in (7) is nested; the ID chooses $\boldsymbol{w}$ of $\mathcal{G}(\boldsymbol{w})$ and the agents select a joint policy which generates a reward signal for the ID. Simultaneous updates of both the ID parameters and the agents' policies, in general, lack converge guarantees due to non-stationarity. Therefore, in order to compute the solution iteratively, after an initial choice by the ID, we let the MARL algorithm run until convergence which fulfils the M-NE constraint for the ID's problem (c.f. Prop. 6.1); the ID receives feedback from the outcome of the game $\mathcal{G}(\boldsymbol{w})$, then updates its choice of $\boldsymbol{w}$. This results in an *inner-outer loop method*. Each step performed by the ID is computationally costly. As such, gradient-based algorithms require a substantial number of iterations to converge to a solution. We therefore use a sample-efficient optimisation algorithm, namely Bayesian optimisation which also allows scaling of the framework. BO also has strong theoretical guarantees for non-convex problems [24] and can handle large dimensional problems [8, 30]. Inner-outer loop methods are widely used in single agent problems to tune hyperparameters of learning algorithms [14].

Inputs: Maximum number of BO evaluations K, and maximum number of MARL iterations M.

Initialise ID's dataset $\mathcal{D}_0 = \{\}$ and reward modifier parameter $\boldsymbol{w}_0$.
for $k = 0, \ldots, K$ **do**
  Initialise agents' strategy profile $\boldsymbol{\pi}_0$.
  for $m = 0, \ldots, M$ **do**
    Agents sample data from the environment following strategy profile $\boldsymbol{\pi}_m$.
    Estimate joint value function (critic) $v_i^{\boldsymbol{\pi}_m,\boldsymbol{w}_k}$.
    Update joint policy (actor) $\boldsymbol{\pi}_{m+1}$.
  end for
  Estimate ID's payoff function $J(\boldsymbol{w}_k, \boldsymbol{\pi}_M)$.
  Select new $\boldsymbol{w}_{k+1}$ guided by current data $\mathcal{D}_k$ using BO with expected improvement criterion.
  Augment dataset $\mathcal{D}_{k+1} = \{\mathcal{D}_k, (\boldsymbol{w}_k, J(\boldsymbol{w}_k, \boldsymbol{\pi}_M))\}$.
end for
Return $\boldsymbol{w}_T$.

Algorithm 1: The ID framework

6.1 Discussion on the method

Convergence. In order to ensure the algorithm converges to an optimal solution for the ID both the inner and outer loop are required to converge. Theorem 4.8 guarantees the existence of a solution for $\boldsymbol{w}^\star$. Convergence of the inner loop is required to obtain the equilibria of the simulated MPG. Consequently, the method is subject to conditions under which MARL methods converge. Hence, the method is subject to conditions under which MARL methods converge. MARL methods have been shown in general, to have strong convergence guarantees to M-NE solutions for MPGs [11, 13, 29]. The following proposition provides this guarantee:

Proposition 6.1 (Convergence). *Algorithm 1 converges to a stable point, moreover the set of stable points of algorithm 1 correspond to M-NE for the MPG.*

Another consideration is the growth in decision complexity of the ID's problem with the number of parameters over which the BO is performed. This depends on the size of the state space of the MAS model. Theorem 3, however proves that approximate solutions are computable with fewer parameters for a given error bound.

7 EXPERIMENTS

7.1 Optimising a Traffic Network

The following experiment illustrates the application of the method to a traffic network problem. We consider road traffic network examples, one of which is a subsection of the city of London. In this setting, each agent seeks to traverse the graph from a source node (labelled 1) to a goal node (labelled 8) - this, for example can represent agents performing a commute. The agents incur costs which represent the travel time. When traversing an edge, each agent incurs a unit cost plus an additional cost which is a convex function of the number of agents traversing the edge at that time - the latter cost represents additional time delays due to traffic congestion.

The goal of each agent is to minimise its own costs. It is well-known that in such systems (e.g. Braess' Paradox, Pigou example), the agents' selfish behaviour of leads to congestion on 'more desirable' paths leading poor system efficiency [22].

The problem is modelled as a *selfish routing game* (SRG) - a widely studied potential game [22] that models traffic networks. In this setting, agents pursuing their individual objectives produce outcomes that result in high travel times for all [31]. In this problem, a set of N self-interested agents direct its *commodity flow* through a network $G = (V, E)$ where V is the set of nodes and $E \subseteq V \times V$ is the set of edges of G. Each agent seeks to direct a single commodity e.g., a taxi firm directing only its fleet. When traversing an edge their commodity produces congestion incurring a negative externality (cost) on all agents. Each agent's commodity is infinitely divisible so that at each node the agents may split their commodity flow over each outgoing edge. Each agent's goal is to direct its commodity through paths that minimise its own costs.

A central planner (CP) seeks to minimise delays due to congestion by devising a dynamic system of toll charges that induces an even commodity flow over a given subset of edges of the network $\hat{E} \subseteq E$ at all times. The CP's problem is to maximise $R_{\text{ID}}(\boldsymbol{w}) = -\sum_{t=1}^{T}[\sum_{l\in\hat{E}}(f^\star(t) - f_l(t))^2]^{1/2}$ where $f_l(t, \boldsymbol{w})$ is the flow on edge $l \in E$ at time t and $f^\star(t) \triangleq (|\hat{E}|)^{-1}\sum_{l\in\hat{E}} f_l(t)$. To induce changes in the agents' commodity flows, the CP adds to $R_{i,e}$ the function $\Theta(f_e)$ which is a power series of order 5.

We consider two cases, we firstly provide an intuitive example known as *Braess' example*, a widely studied problem that clearly demonstrates the inefficiencies of traffic networks [22]. We then apply the method to a subsection of the traffic network in the city of London, UK. We show that our framework finds an optimal system of tolls that leads to maximal system efficiency.

7.1.1 Braess' Example: Fig. 1 shows a diagrammatic illustration of the Nash equilibrium agent flow (the size of the flow of agents through an edge is represented by the edge width) through the network after convergence without ID. As is shown in Fig. 1a), selfish

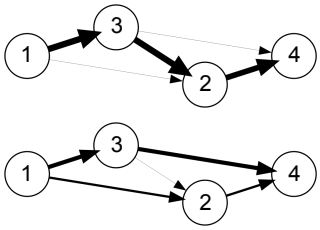

Figure 1: (Top) Braess' example — all agents direct their commodity flow through the middle edge (3→ 2). (Bottom) The (distributed) commodity flows with the ID's toll added.

agents play an M-NE strategy in which they route all their commodity through the middle edge (3→ 2) leading to high congestion costs. As is shown in Figs. 1a) and 1b), when an ID is included, it learns how to set tolls (costs) on the middle edge that induce equal flow over the graph which maximises social welfare.

7.1.2 Extended City Case: We test our method in a complex network consisting of 8 nodes and 13 edges which represents a subsection of the London road network. We show that our method produces socially optimal (M-NE) outcomes. The ID is able to isolate the 3 roads edges to apply tolls in only 150 outer loop iterations.

Our method shows that the ID was able to isolate three nodes to apply a toll which led to a reduction in congestion (indicated in Fig. 2) through the network in only 150 iterations of BO (outer loop). Fig. 2 c) shows the social welfare function (which is the sum of all agents' returns) after 6,000 iterations of the MARL algorithm (inner loop) without the ID (orange curve) and with the ID (blue curve), and demonstrates a significant increase in social welfare. This technique is a first example of reinforcement learning in an SRG that handles large networks and populations of users. This is in contrast to current methods in which agents choose *paths* resulting in exponential scaling in decision complexity with graph size [18].

7.2 Supply & demand matching with thousands of agents

Consider 2,000 agents each seeking to locate themselves at desirable points in space over some time horizon. The desirability of a region changes with time and decreases with the number of agents located in their neighbourhood. The resulting NE distribution is in general, highly inefficient (and may not conform to external objectives) due to agent clustering [15]. The problem is a dynamic generalisation of *the El Faro bar problem* and encapsulates *spectrum sharing problems in wireless communications* [1]. The problem models spatio-economics problems such as firms locating their supply with dynamic demand e.g. freelance taxis. To handle large strategic populations, we use a mean field game framework [15].

A formal description is as follows: the game has a finite set of agents $\mathcal{N} \triangleq \{1, \dots, N\}$, where $N \in \mathbb{N}$. At time $t < T$, the state of the system is $\boldsymbol{x}_t = (x_{i,t})_{i \in \mathcal{N}} \in \mathcal{S}$ where $x_{i,t}$ denotes the location of agent i at time t and $\mathcal{S} \subseteq \mathbb{R}^2$. Each agent i selects action $u_{i,t} \in \mathbb{R}^2$ - a vector movement towards some location $x_{i,t+1} \in \mathcal{S}$. The transition dynamics are given by $x_{i,t+1} = \alpha x_{i,t} + \beta u_{i,t} + \epsilon_{i,t}$,

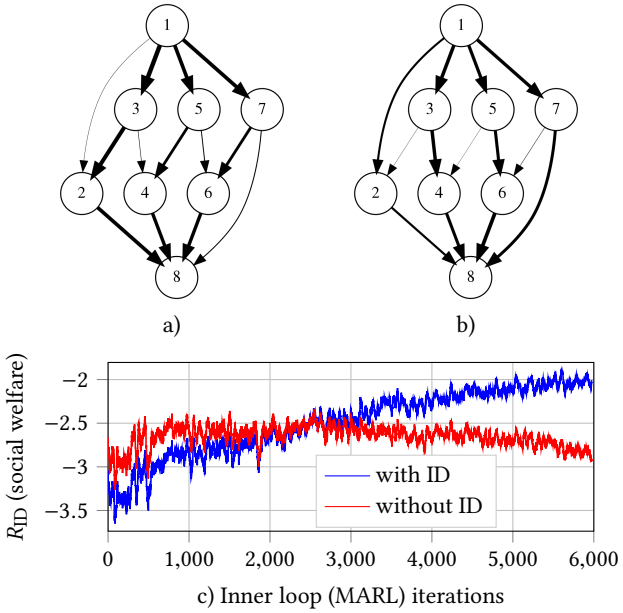

Figure 2: a) Network flow without ID. b) Network flow with ID. Width of the edges represent the size of the flow produced by the agents after converging to an NE with their rewards modified by the ID. c) Comparison of social welfare (SW) with iterations of agents' MARL algorithm (inner loop of Algorithm 1) without ID (red curve) and with ID (blue curve) after $K = 150$ iterations of Bayesian optimisation. Without incentive, the agents converge to an NE that is mindless of the SW, while the inclusion of the incentives leads to a significant increase in SW.

where α, β are scalars, and $\epsilon_{i,t} \sim \mathcal{N}(0, \Sigma)$, for some covariance matrix Σ. The agents' joint action produces a distribution M^a_{t+1} of agents over $\mathcal{S}$. Let $m^a_{x_t} \in \mathbb{P}(\mathcal{H})$ be the density of agents at some location $x_t \in \mathcal{S}$ at time $t \in [0, T]$, where $\mathbb{P}(\mathcal{H})$ denotes the space of probability measures. Each point in $\mathcal{S}$ has some level of *desirability* $\Gamma : \mathcal{S} \times \mathbb{P}(\mathcal{H}) \to \mathbb{R}$ which is determined by the agent's location and the density of agents at that point. Each agent's reward, R_i is given by: $R_i(\boldsymbol{x}_t, m^a_{\boldsymbol{x},u_i}) = \mathbb{E}\Big[\sum_{t=0}^{T} \Gamma(\boldsymbol{x}_t, m^a_{\boldsymbol{x}_t}) - \frac{1}{2} u_{i,t}^\top K u_{i,t}\Big]$, where $\Gamma(\boldsymbol{x}_t, m^a_{\boldsymbol{x}_t}) := (\boldsymbol{x}_t - \tilde{\boldsymbol{x}}_t)^2 - \alpha(m^a_{\boldsymbol{x}_t})^2$,, the expectation is taken over the state-action trajectory induced by the system dynamics and joint policy $\boldsymbol{\pi} \in \boldsymbol{\Pi}$. The term Ψ, rewards the agent for locating closer to $\tilde{\mathbf{x}}_t \in \mathcal{S}$ at time $t \leq T$ whilst penalising the agent for remaining in areas highly concentrated with agents. The quadratic term levies a movement penalty control cost. A principal $\mathbf{P}$ aims to incentivise the self-interested agents to adopt a target distribution $M^\star_t$ at each time step $t \leq T$. $\mathbf{P}$'s objective J is given by a KL divergence between M^a_t and $M^\star_t$ i.e. $J(\boldsymbol{w}, \boldsymbol{\pi}) = \mathbb{E}[\sum_{t=0}^{T} \mathrm{KL}(M^a_t(\boldsymbol{w}, \boldsymbol{\pi}) \| M^\star_t)]$. To incentivise the agents to adopt its desired distribution, $\mathbf{P}$ adds a reward modifier function Θ which is parameterised by $\boldsymbol{w} \in \boldsymbol{W}$. We test our method both *one-shot* and *dynamic* scenarios.

In the **one-shot game** the ID seeks to induce an agent distribution (shown by the left heat map in Fig. 3) - this differs from the

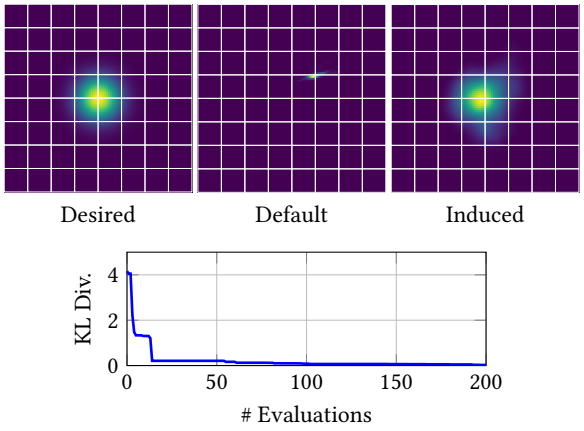

Figure 3: One shot case. (Top) Heat maps of the ID's preferred distribution $M^\star$, the default agent behaviour, and the agents' distribution with modified rewards. (Bottom) Average KL divergences per evaluation of the ID's BO outer loop (averaged over 100 independent tests per evaluation for 4 independent runs).

distribution obtained when agents' maximise only their intrinsic reward function (central heat map in Fig. 3). When the modifier function Θ is added to the agents' rewards, the average KL divergence converges almost to zero which demonstrating a close match of the agents' distribution (right heat-map in Fig. 3) with the desired one.[1]

In the **dynamic game** the ID's desired distribution changes over time. In our experiment, $M_t^\star$ for $t = 0, 1, 2$ are as shown by the heat maps in the top row of Fig. 4 (left), while the bottom row presents the agents' distributions achieved with the ID framework.

8 CONCLUSION

In this paper, we introduce an incentive designer (ID) framework - a technique that enables self-interested adaptive learners to converge to efficient Nash equilibria in Markov games. By adding a modifier function to the agents' rewards, our method learns to modify the rewards of self-interested agents to induce efficient, desirable equilibrium outcomes. We prove a continuity property in the ID's modifications to the game which permits a broad range of black-box optimisation techniques to be applied.

9 APPENDIX

Lemma A.1. *Let A and B be sets and let $f : \mathbb{A} \times \mathbb{B} \to \mathbb{R}$ and $h : \mathbb{A} \times \mathbb{B} \to \mathbb{R}$ be two real-valued maps s.th. the following expression holds $\forall a \in \mathbb{A}, b \in \mathbb{B}$ and for some constant c: $|f(a,b) - h(a,b)| < c, \implies |\max_{a\in\mathbb{A}, b\in\mathbb{B}} f(a,b) - \max_{a\in\mathbb{A}, b\in\mathbb{B}} h(a,b)| < c$*

Proof. By (A.1) we have that $f(a,b) < c + h(a,b)$. Applying the max operator and taking absolute values yields: $\max_{a\in A, b\in B} f(a,b) < c + \max_{a\in A, b\in B} h(a,b) \implies |\max_{a\in A, b\in B} f(a,b) - \max_{a\in A, b\in B} h(a,b)| < c$ □

[1]The small discrepancy from 0 is due to Gaussian approximation of the agent density.

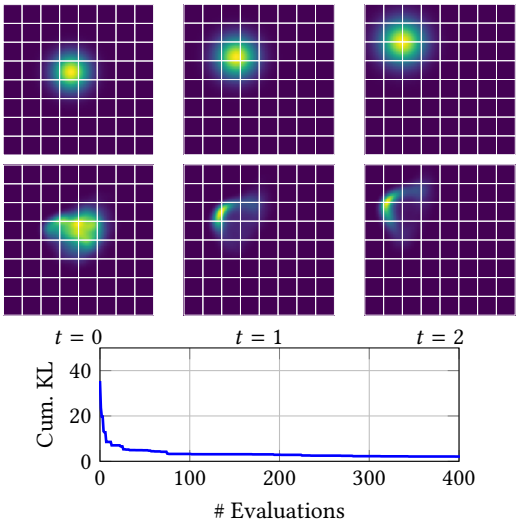

Figure 4: Dynamic case. (Top) Heat maps represent (first row) the ID's preferred distribution $M_t^\star$, (second row) the induced agent distribution M_t^a at time-steps $t = 0, 1, 2$. (Bottom) Average episodic cumulative KL divergences per evaluation of the ID's BO outer loop (averaged over 100 independent tests per evaluation for 4 independent runs). Without ID, the agents behave similar to the default behaviour displayed in Fig. 3-Top middle.

Proof of Proposition 4.3:

To prove the proposition, we consider the two cases (trajectory targeted and welfare targeted) of the ID's goal separately.

Case I: Welfare Targeted

The agents' reward functions $R_{i,\boldsymbol{w}}$ are Lipschitz continuous in $\boldsymbol{w}$. This follows from the fact that the composite function $g_1 \circ (g_2 \circ (\ldots \circ (g_n(\cdot)\ldots))$ of $n < \infty$ Lipschitzian functions $g_1, g_2, \ldots, g_n$ is itself Lipschitzian (moreover we can then apply Rademacher's lemma to ascertain differentiability almost everywhere).

Specifically, we have that $R_{\text{ID}}(\boldsymbol{w}, h(v^{\cdot,\boldsymbol{w}})) - R_{\text{ID}}(\boldsymbol{w'}, h(v^{\cdot,\boldsymbol{w'}})) \leq L_{R_{\text{ID}}}\|\boldsymbol{w} - \boldsymbol{w'}\| + \left(h(v^{\cdot,\boldsymbol{w}}) - (v^{\cdot,\boldsymbol{w'}})\right) \leq L'\|\boldsymbol{w} - \boldsymbol{w'}\|$ where $L' \triangleq L_{R_{\text{ID}}} + L_h$ and $L_{R_{\text{ID}}}$ and L_h are the Lipschitz constants of R_{ID} and h (resp.). Since $J(\boldsymbol{w}, \boldsymbol{\pi}) \triangleq \mathbb{E}\left[R_{\text{ID}}(\boldsymbol{w}, h(v_a^{\boldsymbol{\pi},\boldsymbol{w}}), \zeta)\right]$ and h is uniformly continuous, it follows J is expressible as a composite function of uniformly continuous functions and hence is itself uniformly continuous (since it is in fact Lipschitz continuous). To complete the proof it remains only to consider the trajectory targeted case.

Case II: Trajectory Targeted

Consider a sequence $\{\boldsymbol{w_n}\}$ s.th. $\boldsymbol{w_n} \to \boldsymbol{w}$ as $n \to \infty$, then $\exists c, d > 0$:

$$\mathbb{E}\left[|J(\boldsymbol{w}, X^{\boldsymbol{\pi}(\boldsymbol{w})}) - J(\boldsymbol{w_n}, X^{\boldsymbol{\pi}(\boldsymbol{w_n})})|\right] \leq c|\boldsymbol{w} - \boldsymbol{w_n}| + d|X^{\boldsymbol{\pi}(\boldsymbol{w})} - X^{\boldsymbol{\pi}(\boldsymbol{w_n})}|,$$

using the Lipschitzianity of J. Since $X^{\boldsymbol{\pi}(\boldsymbol{w_n})} \to X^{\boldsymbol{\pi}(\boldsymbol{w})}$ as $n \to \infty$, then by (10) and by the dominated convergence theorem we deduce that $\exists M \in \mathbb{N}$ s.th. for $n \geq M$:

$$\mathbb{E}\left[J(\boldsymbol{w}, X^{\boldsymbol{\pi}(\boldsymbol{w})}) - J(\boldsymbol{w_n}, X^{\boldsymbol{\pi}(\boldsymbol{w_n})})\right] < c\delta$$

for some constants $c > 0$ and $\delta > 0$ s.th $\delta \to 0$ as $n \to \infty$.

REFERENCES

[1] S. Ahmad, C. Tekin, M. Liu, R. Southwell, and J. Huang. 2010. Spectrum sharing as spatial congestion games. *arXiv CoRR* abs/1011.5384 (2010).

[2] Monica Babes, Enrique Munoz De Cote, and Michael L Littman. 2008. Social reward shaping in the prisoner's dilemma. In *Proc. 7th Int. joint Conf. on Auton. agents and multiagent systems-Volume 3*. International Foundation for Autonomous Agents and Multiagent Systems, 1389–1392.

[3] Xu Chen, Lei Jiao, Wenzhong Li, and Xiaoming Fu. 2016. Efficient multi-user computation offloading for mobile-edge cloud computing. *IEEE/ACM Transactions on Networking* 5 (2016), 2795–2808.

[4] B. Colson, P. Marcotte, and G. Savard. 2007. An overview of bilevel optimization. *Annals of operations research* 153, 1 (2007), 235–256.

[5] André de Palma and Robin Lindsey. 2011. Traffic congestion pricing methodologies and technologies. *Transportation Research Part C: Emerging Technologies* 19, 6 (2011), 1377–1399.

[6] Sam Devlin and Daniel Kudenko. 2011. Theoretical considerations of potential-based reward shaping for multi-agent systems. In *The 10th Int. Conf. on Auton. Agents and Multiagent Systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 225–232.

[7] Pradeep Dubey. 1986. Inefficiency of Nash equilibria. *Mathematics of Operations Research* 11, 1 (1986), 1–8.

[8] Nicolas Durrande. 2001. *Étude de classes de noyaux adaptées à la simplification et à l'interprétation des modèles d'approximation. Une approche fonctionnelle et probabiliste.* Ph.D. Dissertation. Ecole Nationale Supérieure des Mines de Saint-Etienne.

[9] Jordi Grau-Moya, Felix Leibfried, and Haitham Bou-Ammar. 2018. Balancing Two-Player Stochastic Games with Soft Q-Learning. *arXiv:1802.03216* (2018).

[10] John C Harsanyi, Reinhard Selten, et al. 1988. A general theory of equilibrium selection in games. *MIT Press Books* 1 (1988).

[11] Johannes Heinrich, Marc Lanctot, and David Silver. 2015. Fictitious self-play in extensive-form games. In *Int. Conf. on Machine Learning*. 805–813.

[12] Christian Ibars, Monica Navarro, and Lorenza Giupponi. 2010. Distributed demand management in smart grid with a congestion game. In *Smart grid communications (SmartGridComm), 2010 1st IEEE Int. Conf.* IEEE, 495–500.

[13] David S. Leslie and Edmund J. Collins. 2006. Generalised weakened fictitious play. *Games and Economic Behavior* 56 (2006), 285 – 298.

[14] Dougal Maclaurin, David Duvenaud, and Ryan Adams. 2015. Gradient-based hyperparameter optimization through reversible learning. In *Int. Conf. on Mach. Learning*. 2113–2122.

[15] D. Mguni, J. Jennings, and E. Munoz de Cote. 2018. Decentralised Learning in Systems with Many, Many Strategic Agents. In *Proc. of 31st AAAI Conf. on AI (AAAI-18)*.

[16] Fei Miao, Shuo Han, Shan Lin, John A Stankovic, Desheng Zhang, Sirajum Munir, Hua Huang, Tian He, and George J Pappas. 2016. Taxi dispatch with real-time sensing data in metropolitan areas: A receding horizon control approach. *IEEE Transactions on Automation Science and Engineering* 13, 2 (2016), 463–478.

[17] Amyaz A Moledina, Jay S Coggins, Stephen Polasky, and Christopher Costello. 2003. Dynamic environmental policy with strategic firms: prices versus quantities. *Journal of Environmental Economics and Management* 45, 2 (2003), 356–376.

[18] Luca Moscardelli. 2013. Convergence Issues in Congestion Games. *Bulletin of EATCS* 3, 111 (2013).

[19] Noam Nisan and Amir Ronen. 2001. Algorithmic mechanism design. *Games and Economic Behavior* 35, 1-2 (2001), 166–196.

[20] M. J. Osborne and A. Rubinstein. 1994. *A Course in Game Theory*. MIT Press.

[21] Alexander Peysakhovich and Adam Lerer. 2017. Prosocial learning agents solve generalized Stag Hunts better than selfish ones. *arXiv preprint arXiv:1709.02865* (2017).

[22] Tim Roughgarden. 2005. *Selfish routing and the price of anarchy*. Vol. 174. MIT press Cambridge.

[23] M. A. Satterthwaite. 1975. Strategy-proofness and Arrow's conditions: Existence and correspondence theorems for voting procedures and social welfare functions. *Journal of Economic Theory* 10, 2 (1975), 187 – 217.

[24] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. 2016. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proc. IEEE* 104, 1 (2016), 148–175.

[25] Yoav Shoham, Rob Powers, and Trond Grenager. 2003. *Multi-agent reinforcement learning: a critical survey*. Technical Report. Technical report, Stanford University.

[26] Margaret E Slade. 1994. What does an oligopoly maximize? *The Journal of Industrial Economics* (1994), 45–61.

[27] Pingzhong Tang. 2017. Reinforcement mechanism design. In *Early Career Highlights at Proc. 26th Int. Joint Conf. on AI (IJCAI, pages 5146–5150.*

[28] Kurian Tharakunnel and Siddhartha Bhattacharyya. 2007. Leader-Follower semi-Markov decision problems: theoretical framework and approximate solution. In *Approximate Dynamic Programming and Reinforcement Learning, 2007. ADPRL 2007. IEEE Int. Symp. on*. IEEE, 111–118.

[29] S. Valcarcel Macua, J. Zazo, and S. Zazo. 2018. Learning Parametric Closed-Loop Policies for Markov Potential Games. In *To appear in Proc. of the 6th Int. Conf. on Learning Representations (ICLR)*.

[30] Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando de Freitas. 2016. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research* 55 (2016), 361–387.

[31] Hyejin Youn, Michael T Gastner, and Hawoong Jeong. 2008. Price of anarchy in transportation networks: efficiency and optimality control. *Phys. Rev. Lett.* 101, 12 (2008), 128701.

[32] S. Zazo, S. Valcarcel Macua, M. SÃąnchez-FernÃąndez, and J. Zazo. 2016. Dynamic Potential Games With Constraints: Fundamentals and Applications in Communications. *IEEE Transactions on Signal Processing* 64, 14 (July 2016), 3806–3821.