LIGS: LEARNABLE INTRINSIC-REWARD GENERATION SELECTION FOR MULTI-AGENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Efficient exploration is important for reinforcement learners (RL) to achieve high rewards. In multi-agent systems, *coordinated* exploration and behaviour is critical for agents to jointly achieve optimal outcomes. In this paper, we introduce a new general framework for improving coordination and performance of multi-agent reinforcement learners (MARL). Our framework, named Learnable Intrinsic-Reward Generation Selection algorithm (LIGS) introduces an adaptive learner, Generator that observes the agents and learns to construct intrinsic rewards online that coordinate the agents' joint exploration and joint behaviour. Using a novel combination of reinforcement learning (RL) and switching controls, LIGS determines the best states to learn to add intrinsic rewards which leads to a highly efficient learning process. LIGS can subdivide complex tasks making them easier to solve and enables systems of RL agents to quickly solve environments with sparse rewards. LIGS can seamlessly adopt existing multi-agent RL algorithms and our theory shows that it ensures convergence to joint policies that deliver higher system performance. We demonstrate the superior performance of the LIGS framework in challenging tasks in Foraging and StarCraft II.

1 INTRODUCTION

Cooperative multi-agent reinforcement learning (MARL) has emerged as a powerful tool to enable autonomous agents to solve various tasks such as ride-sharing (Zhou, Luo, et al., 2020) and swarm robotics (Hüttenrauch et al., 2017; Mguni et al., 2018). In multi-agent systems (MAS), maximising system performance often requires agents to coordinate during exploration and learn coordinated joint actions (Matignon et al., 2012). However, in many MAS, the reward signal provided by the environment is not sufficient to guide the agents towards coordinated behaviour (Matignon et al., 2012). Consequently, relying on solely the individual rewards received by the agents may not lead to optimal outcomes (Mguni et al., 2019). This problem is worsened by the fact that MAS can have many stable points some of which lead to arbitrarily bad outcomes (Roughgarden & Tardos, 2007).

As in single agent RL, in MARL inefficient exploration can dramatically decrease sample efficiency. In MAS, a major challenge is how to overcome sample inefficiency from poorly *coordinated exploration*. Unlike single agent RL, in MARL, the collective of agents is typically required to coordinate its exploration to find their optimal joint policies[1]. A second issue is that in many MAS settings of interest, such as video games and physical tasks, rich informative signals of the agents' *joint* performance are not readily available (Hosu & Rebedea, 2016). For example, in StarCraft Micromanagement (Samvelyan et al., 2019), the sparse reward alone (win, lose) gives insufficient information to guide agents toward their optimal joint policy. Consequently, MARL requires large numbers of samples producing a great need for MARL methods that can solve such problems efficiently.

To aid coordinated learning, algorithms such as QMIX (Rashid et al., 2018), MAVEN (Mahajan et al., 2019) and COMA (Foerster et al., 2018), so-called centralised critic and decentralised execution (CT-DE) methods use a centralised critic whose role is to estimate the agents' expected returns. The critic makes use of all available information generated by the system, specifically the global state and the joint action. To enable effective CT-DE, it is critical that the joint greedy action

[1] In single agent RL exploration issues can be mitigated by adjusting exploration rates or the policy variance (Tijsma et al., 2016). However it has been shown that the same is not possible in MARL (Mahajan et al., 2019).

should be equivalent to the collection of individual greedy actions of agents, which is called the IGM (Individual-Global-Max) principle (Son et al., 2019). CT-DE methods are however, prone to convergence to suboptimal joint policies (Wang, Ren, Han, et al., 2020). In particular, existing value factorisations, e.g. QMIX and VDN (Sunehag et al., 2017), cannot ensure an exact guarantee of IGM consistency (Wang, Ren, Liu, et al., 2020). Moreover, CT-DE methods such as QMIX require a monotonicity condition which is violated in scenarios where multiple agents must coordinate but are penalised if only a subset of them do so (see Exp. 2, Sec. 6.1).

To tackle these issues, in this paper we introduce a new MARL framework, LIGS that constructs intrinsic rewards online which guide MARL learners towards their optimal joint policy. LIGS involves an *adaptive* intrinsic reward agent, Generator that selects intrinsic rewards to add according to the history of visited states and joint actions performed by the agents. Generator adaptively guides the agents' exploration and behaviour towards coordination and maximal joint performance. A pivotal feature of LIGS is the novel combination of RL and *switching controls* (Bayraktar & Egami, 2010; Mguni, 2018) which enables it to determine the best set of states to learn to add intrinsic rewards while disregarding less useful states. This enables Generator to quickly learn how to set intrinsic rewards that guide the agents during their learning process. Moreover, the intrinsic rewards added by Generator can significantly deviate from the environment rewards. This enables LIGS to both promote complex *joint exploration* patterns and decompose difficult tasks. Despite this flexibility, special features within LIGS ensure the underlying optimal policies are preserved so that the agents learn to solve the task at hand.

Overall, LIGS has several advantages:
• LIGS has the freedom to introduce rewards that vastly deviate from the environment rewards. With this, LIGS promotes *coordinated exploration* (i.e. visiting unplayed state-joint actions) among the agents enabling them to find joint policies that maximise the system rewards and generates intrinsic rewards to aid solving sparse reward MAS.
• LIGS selects which best states to add intrinsic rewards *adaptively* in response to the agents' behaviour while the agents learn leading to an efficient learning process.
• LIGS's intrinsic rewards preserve the agents' optimal joint policy and ensure that the total *environment* return is (weakly) increased.

To enable the framework to perform successfully, we overcome several challenges: **i)** Firstly, constructing an intrinsic reward can change the underlying problem leading to the agents solving irrelevant tasks (Mannion et al., 2017). We resolve this by endowing the intrinsic reward function with special form which both allows a rich spread of intrinsic rewards while preserving the optimal policy. **ii)** Secondly, introducing intrinsic reward functions can *worsen* the agents' performance (S. Devlin & Kudenko, 2011) and doing so *while training* can lead to convergence issues. We prove LIGS leads to better performing policies and that LIGS's learning process converges and preserves the MARL learners' convergence properties. **iii)** Lastly, adding an agent Generator with its own goal leads to a Markov game (MG) with $N + 1$ agents (Fudenberg & Tirole, 1991). Tractable methods for solving MGs are extremely rare with convergence only in special cases (Yang & Wang, 2020). Nevertheless, using a special set of features in LIGS's design, we prove LIGS converges to a solution in which it learns an intrinsic reward function that improves the agents' performance.

2 Related Work

Multi-agent exploration methods seek to promote coordinated exploration among MARL learners. Maven et. al (Mahajan et al., 2019) propose a hybridisation of value and policy-based methods that uses mutual information to learn a diverse set of behaviours between agents. Though this approach promotes coordinated exploration, it does not encourage exploration of novel states. Other approaches to promote exploration in MARL while assuming aspects of the environment are known in advance and agents can perform perfect communication between themselves (Viseras et al., 2016). Similarly, to promote coordinated exploration in partially observable settings, (Pesce & Montana, 2020) propose end-to-end learning of a communication protocol through a memory device. In general, exploration-based methods provide no performance guarantees nor do they ensure the optimal policy (of the underlying dec-MDP) is preserved. Moreover, many employ heuristics that naively reward exploration to unvisited states without consideration of the environment reward. This can lead to spurious objectives being maximised.

Reward shaping (Harutyunyan et al., 2015) (RS) is a technique which aims to alleviate the problem of sparse and uninformative rewards by supplementing the agent's reward with a prefixed term F. In (Ng et al., 1999) it was established that adding a *shaping reward function* of the form $F(s_{t+1}, s_t) = \gamma\phi(s_{t+1}) - \phi(s_t)$ preserves the optimal policy and in some cases can aid learning. RS has been extended to MAS (S. Devlin & Kudenko, 2011, 2016; S. Devlin et al., 2011; S. M. Devlin & Kudenko, 2012; Mannion et al., 2018; Sadeghlou et al., 2014) in which it is used to promote convergence to efficient social welfare outcomes. Poor choices of F in a MAS can slow the learning process and can induce convergence to poor system performance (S. Devlin & Kudenko, 2011). In MARL, the question of which shaping function to use remains unaddressed. Typically, RS algorithms rely on hand-crafted shaping reward functions that are constructed using domain knowledge, contrary to the goal of autonomous learning (S. Devlin & Kudenko, 2011). As we later describe LIGS, which successfully *learns* an instrinsic reward function F, uses a similar form as PBRS however, F is now augmented to include the actions of another RL agent to learn the intrinsic rewards online. In (Du et al., 2019) an approach towards learning intrinsic rewards is proposed in which a parameterised intrinsic reward is learned using a bilevel approach through a centralised critic. Loosely related are single-agent methods (Dilokthanakul et al., 2019; Kulkarni et al., 2016; Pathak et al., 2017; Zheng et al., 2018) which, in general, introduce heuristic terms to generate intrinsic rewards.

Within these categories, closest to our work is the intrinsic reward approach in (Du et al., 2019). There, the agents' policies and intrinsic rewards are learned with a bilevel approach. In contrast, LIGS performs these operations *concurrently* leading to a faster, more efficient procedure. A crucial point of distinction is that in LIGS, the intrinsic rewards are constructed by an RL agent (Generator) with its own reward function. Consequently, LIGS can generate complex patterns of intrinsic rewards, encourage *joint exploration*. Additionally, LIGS learns intrinsic rewards only at relevant states, this confers high computational efficiency. Lastly, unlike exploration-based methods e.g., (Mahajan et al., 2019), LIGS ensures preservation of the agents' joint optimal policy for the task.

3 PRELIMINARIES

A fully cooperative MAS is modelled by a decentralised-Markov decision process (dec-MDP) (Yang & Wang, 2020). A dec-MDP is an augmented MDP involving a set of $N \geq 2$ agents denoted by $\mathcal{N}$ that independently decide actions to take which they do so simultaneously over many rounds. Formally, a dec-MDP is a tuple $\mathfrak{M} = \langle \mathcal{N}, \mathcal{S}, (\mathcal{A}_i)_{i\in\mathcal{N}}, P, R, \gamma \rangle$ where $\mathcal{S}$ is the finite set of states, $\mathcal{A}_i$ is an action set for agent $i \in \mathcal{N}$ and $R : \mathcal{S} \times \boldsymbol{\mathcal{A}} \to \mathcal{P}(D)$ is the reward function that all agents jointly seek to maximise where D is a compact subset of $\mathbb{R}$ and lastly, $P : \mathcal{S} \times \boldsymbol{\mathcal{A}} \times \mathcal{S} \to [0, 1]$ is the probability function describing the system dynamics where $\boldsymbol{\mathcal{A}} := \times_{i=1}^{N} \mathcal{A}_i$. Each agent $i \in \mathcal{N}$ uses a *Markov policy* $\pi_i : \mathcal{S} \times \mathcal{A}_i \to [0, 1]$ to select its actions. At each time $t \in 0, 1, \ldots$, the system is in state $s_t \in \mathcal{S}$ and each agent $i \in \mathcal{N}$ takes an action $a_t^i \in \mathcal{A}_i$. The *joint action* $\boldsymbol{a}_t = (a_t^1, \ldots, a_t^N) \in \boldsymbol{\mathcal{A}}$ produces an immediate reward $r_i \sim R(s_t, \boldsymbol{a}_t)$ for agent $i \in \mathcal{N}$ and influences the next-state transition which is chosen according to P. The goal of each agent i is to maximise its expected returns measured by its value function $v^{\pi^i, \pi^{-i}}(s) = \mathbb{E}_{\pi^i, \pi^{-i}}[\sum_{t=0}^{\infty} \gamma^t R(s_t, \boldsymbol{a}_t)]$, where Π_i is a compact Markov policy space and $-i$ denotes the tuple of agents excluding agent i.

Intrinsic rewards can strongly induce more efficient learning (and can promote convergence to higher performing policies) (S. Devlin & Kudenko, 2011). We tackle the problem of how to *learn* intrinsic rewards produced by a function $F^\star$ that leads to the agents learning policies that jointly maximise the system performance (through coordinated learning). Naturally, the problem of learning $F^\star$ can be approached by optimising the input $\boldsymbol{\theta} \in \mathbb{R}^m$ of a parameteric function: $\hat{F}(s, \boldsymbol{a}; \boldsymbol{\theta})$; that is, finding $\boldsymbol{\theta}^\star \in \mathbb{R}^m$ for which $F^\star(s, \boldsymbol{a}) = \hat{F}(s, \boldsymbol{a}, \boldsymbol{\theta}^\star)$ given an intrinsic reward $\hat{F}$ where s is the state and $\boldsymbol{a}$ is the agents' joint action. Determining this function is a significant challenge since poor choices can hinder learning and the concurrency of multiple learning processes presents potential convergence issues in a system already populated by multiple learners (Zinkevich et al., 2006). Additionally, we require that the method preserves the optimal joint policy and the underlying dec-MDP. Note that using an optimisation procedure to find $\boldsymbol{\theta}^\star$ directly does not make use of information generated by intermediate state-joint-action-reward tuples of the RL problem which can guide the optimisation.

4 The LIGS Framework

We now describe the details of the LIGS framework and how it learns intrinsic rewards that improve learning and team performance. We then describe the agents' objectives and their learning processes.

To tackle the challenges described above, we introduce Generator an *adaptive* agent with its own objective that determines the best intrinsic rewards to give to the agents at each state. Using observations of the joint actions played by the N agents, the goal of Generator is to construct intrinsic rewards (which the N MARL learners cannot generate themselves) to coordinate exploration and guide the agents towards learning joint policies that maximise their shared rewards. To do this, Generator learns how to choose the values of an intrinsic reward function $F^{\boldsymbol{\theta}}$ at each state. Simultaneously, the N agents perform actions to maximise their rewards using their individual policies. The objective for each agent $i \in \{1, \ldots, N\}$ is given by:

$$v^{\pi^i,\pi^{-i},g}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \left(R + F^{\boldsymbol{\theta}}\right) \Big| s_0 = s\right],$$

where $\boldsymbol{\theta}$ is determined by Generator using the policy $g : \mathcal{S} \times \Theta \to [0, 1]$ and $\Theta \subset \mathbb{R}^p$ is Generator's action set. The intrinsic reward function is given by $F^{\boldsymbol{\theta}}(\cdot) \equiv \phi(s_t, \theta^c_t) - \gamma^{-1}\phi(s_{t-1}, \theta^c_{t-1})$ for any $s_t, s_{t-1} \in \mathcal{S}$ where $\theta^c_t \sim g$ is the action chosen by Generator and $\theta^c_t \equiv 0, \forall t < 0$. The function $\phi : \mathcal{S} \times \Theta \to \mathbb{R}$ is a continuous map that satisfies the condition $\phi(s, 0) \equiv 0, \forall s \in \mathcal{S}$ (for example, ϕ can be a neural network with fixed weights with input (s, θ^c) and Θ can be a set of integers $\{1, \ldots, K\}$). Therefore, Generator determines the output of $F^{\boldsymbol{\theta}}$ (which it does through its choice of θ^c). With this, Generator constructs intrinsic rewards that are tailored for the specific setting.

As the agent $i \in \mathcal{N}$ policy can be learned using any MARL method, LIGS freely adopts any MARL algorithm for the N agents (see Sec. 9 in the Supp. Material). The transition probability $P : \mathcal{S} \times \boldsymbol{\mathcal{A}} \times \mathcal{S} \to [0, 1]$ takes the state and *only* the actions of the N agents as inputs. Note that unlike reward-shaping methods e.g. (Ng et al., 1999), the function ϕ now contains an action term θ^c which is chosen by Generator which enables the intrinsic reward function to be learned online. The presence of the action θ^c term may spoil the policy invariance result in (Ng et al., 1999). We however prove a policy invariance result (Prop. 1) analogous to that in (Ng et al., 1999) which shows LIGS preserves the optimal policy of $\mathfrak{M}$.
Generator is an RL agent whose objective takes into account the history of states and N agents' joint actions. Define by $R^{\boldsymbol{\theta}}(s, \boldsymbol{a}) := R(s, \boldsymbol{a}) + F^{\boldsymbol{\theta}}$ Generator's objective is:

$$v^{\boldsymbol{\pi},g}_c(s) = \mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty} \gamma^t \left(R^{\boldsymbol{\theta}}(s_t, \boldsymbol{a}_t) + L(s_t, \boldsymbol{a}_t)\right) \Big| s_0 = s\right], \quad \forall s \in \mathcal{S}. \tag{1}$$

The objective encodes Generator's agenda, namely to maximise the agents' joint expected return. Therefore, using its intrinsic rewards, Generator seeks to guide the set of agents toward optimal joint trajectories (potentially away from suboptimal trajectories, c.f. Experiment 2) and enables the agents to learn faster (c.f. StarCraft experiments in Sec. 6). Lastly, $L : \mathcal{S} \times \boldsymbol{\mathcal{A}} \to \mathbb{R}$ rewards Generator when the agents jointly visit novel state-joint-action tuples and tends to 0 as the tuples are revisited. We later prove that with this objective, Generator's optimal policy (for constructing the intrinsic rewards) maximises the expected team (extrinsic) return (Prop. 1).

Since Generator has its own (distinct) objective, the resulting setup is an MG, $\mathcal{G} = \langle \mathcal{N} \times \{c\}, \mathcal{S}, (\mathcal{A}_i)_{i \in \mathcal{N}}, \Theta, P, R^{\boldsymbol{\theta}}, R_c, \gamma \rangle$ where the new elements are $\{c\}$, Generator agent, $R^{\boldsymbol{\theta}} := R + F^{\boldsymbol{\theta}}$, the new team reward function which contains the intrinsic reward $F^{\boldsymbol{\theta}}$, $R_c : \mathcal{S} \times \boldsymbol{\mathcal{A}} \times \Theta \to \mathbb{R}$, the one-step reward for Generator (we give the details of this later). The transition probability $P : \mathcal{S} \times \boldsymbol{\mathcal{A}} \times \mathcal{S} \to [0, 1]$ takes the state and *only* the N agents joint action as inputs.

Switching Control Mechanism

So far Generator's problem involves learning to construct intrinsic rewards at *every* state which can be computationally expensive. We now introduce an important feature to the framework with which it only learns the best intrinsic reward in a subset of states in which intrinsic rewards are most useful. This is in contrast to the problem tackled by the N agents who must compute their optimal actions at all states. To achieve this, we now replace Generator's policy space with a form of policies known as *switching controls*. These policies enable Generator to decide at which states to learn the value of intrinsic rewards. This enables Generator to learn quickly both where to add intrinsic rewards and the magnitudes that improve performance since Generator's magnitude optimisations are performed only at a subset of states. Crucially, with this Generator can learn its policy rapidly enabling it to guide the agents toward coordination and higher performing policies while they train.

At each state Generator first makes a *binary decision* to decide to *switch on* its F for agent $i \in \mathcal{N}$ using a switch I_t which takes values in $\{0, 1\}$. Crucially, now Generator is tasked with learning how to construct the N agents' intrinsic rewards *only* at states that are important for guiding the agents to their joint optimal policy. Both the decision to activate the function F and its magnitudes is determined by Generator. With this, the agent $i \in \mathcal{N}$ objective becomes:

$$v^{\boldsymbol{\pi},g}(s_0, I_0) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \left\{R + F^{\boldsymbol{\theta}} \cdot I_t\right\}\right], \; \forall (s_0, I_0) \in \mathcal{S} \times \{0, 1\}, \tag{2}$$

where $I_{\tau_{k+1}} = 1 - I_{\tau_k}$, which is the switch for the function F which is 0 or 1 and $\{\tau_k\}_{k>0}$ are times that a switch takes place [2] so for example if the switch is first turned on at the state s_5 then turned off at s_7, then $\tau_1 = 5$ and $\tau_2 = 7$ (we will shortly describe these in more detail). At any state, the decision to turn on I is decided by a (categorical) policy $\mathfrak{g}_c : \mathcal{S} \to \{0, 1\}$ which acts according to Generator's objective. In particular, first, Generator makes an observation of the state $s_k \in \mathcal{S}$ and the joint action $\boldsymbol{a}_k$ and using $\mathfrak{g}_c$, Generator decides whether or not to activate the policy g to provide an intrinsic reward whose value is determined by $\theta_k^c \sim g$. With this it can be seen the sequence of times $\{\tau_k\}$ is $\tau_k = \inf\{t > \tau_{k-1} | s_t \in \mathcal{S}, \mathfrak{g}_c(s_t) = 1\}$ so the switching times. $\{\tau_k\}$ *are* ***rules*** *that depend on the state.* Therefore, by learning an optimal $\mathfrak{g}_c$, Generator learns the useful states to switch on F.

As we later describe, the termination times occur according to some external (probabilistic) rule. To induce Generator to selectively choose when to switch on the additional rewards, each switch activation incurs a fixed cost for Generator. In this case, the objective for Generator is:

$$v_c^{\boldsymbol{\pi},g}(s_0, I_0) = \mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty} \gamma^t \left(R^{\boldsymbol{\theta}}(s_t, \boldsymbol{a}_t) + \sum_{k \geq 1} c(I_t, I_{t-1})\delta^t_{\tau_{2k-1}} + L(s_t, \boldsymbol{a}_t)\right)\right], \tag{3}$$

where the new term $c : \{0, 1\}^2 \to \mathbb{R}_{<0}$ is a strictly negative cost function which imposes a cost for each switch activation and is modulated by the Kronecker-delta function $\delta^t_{\tau_{2k-1}}$ which is 1 whenever $t = \tau_{2k-1}$ and 0 otherwise. The cost has two effects: first, it reduces the computational complexity of Generator's problem since Generator now determines *subregions* of $\mathcal{S}$ it should learn the values of F. Second, it ensures the *information-gain* from encouraging the agents to explore a given set of state-action tuples is sufficiently high to merit activating a stream of intrinsic rewards. We set $\tau_0 \equiv 0$, $\theta_{\tau_k} \equiv 0, \forall k \in \mathbb{N}$ $(\theta_{\tau_k+1}, \ldots, \theta_{\tau_{k+1}-1}$ remain non-zero), $\theta_k^c \equiv 0 \;\; \forall k \leq 0$ and denote by $I(t) \equiv I_t$.

There are various possibilities for the *termination* times $\{\tau_{2k}\}$ (recall that $\{\tau_{2k+1}\}$ are the times which the intrinsic reward F is *switched on) using* $\mathfrak{g}_c$. One is for the terminations to occur randomly. Another is to build a construction of $\{\tau_{2k}\}$ that directly incorporates the information gain that a state visit provides — we defer the details of this arrangement to Sec. 14 of the Appendix.

Summary of Events

At a time $t \in 0, 1, \ldots$

1. The N agents makes an observation of the state $s_t \in \mathcal{S}$.
2. The N agents perform a joint action $\boldsymbol{a}_t = (a_t^1, \ldots, a_t^N)$ sampled from $\boldsymbol{\pi} = (\pi^1, \ldots, \pi^N)$.
3. Generator makes an observation of s_t and $\boldsymbol{a}_t$ and draws samples from its polices $(\mathfrak{g}_c, g)$.
4. If $\mathfrak{g}_c(s_t) = 0$:
   ◦ Each agent $i \in \mathcal{N}$ receives a reward $r_i \sim R(s_t, \boldsymbol{a}_t)$ and the system transitions to the next state s_{t+1} and steps 1 - 3 are repeated.
5. If $\mathfrak{g}_c(s_t) = 1$:
   ◦ $F^{\boldsymbol{\theta}}$ is computed using s_t, $\boldsymbol{a}_t$ and the Generator action $\theta^c \sim g$.
   ◦ Each agent $i \in \mathcal{N}$ receives a reward $r_i + F^{\boldsymbol{\theta}}$ and the system transitions to s_{t+1}.
6. At time $t + 1$ if the intrinsic reward terminates then steps 1 - 3 are repeated or if the intrinsic reward has not terminated then step 5 is repeated.

[2] More precisely, $\{\tau_k\}_{k\geq 0}$ are *stopping times* (Øksendal, 2003).

4.1 The Learning Procedure

In Sec. 5, we provide the convergence properties of the algorithm, and give the full code of the algorithm in Sec. 8 of the Appendix. The algorithm consists of the following procedures: Generator updates its policy that determines the values θ at each state and the states to perform a switch while the agents $\{1, \ldots, N\}$ learn their individual policies $\{\pi_1, \ldots, \pi_N\}$. In our implementation, we used proximal policy optimization (PPO) (Schulman et al., 2017) as the learning algorithm for both Generator's intervention policy $\mathfrak{g}_c$ and Generator's policy g. For the N agents we used MAPPO (Yu et al., 2021). For Generator L term we use[3] $L(s_t, \boldsymbol{a}_t) := \|\hat{h} - h\|_2^2$ where $\hat{h}$ is a random initialised network which is the target network which is fixed and h is the *prediction function* that is consecutively updated during training. We constructed $\hat{F}$ using a fixed neural network $f : \mathbb{R}^d \mapsto \mathbb{R}^m$ and a one-hot encoding of the action of Generator. Specifically, $\phi(s_t, \theta_t^c) := f(s_t) \cdot i(\theta_t^c)$ where $i(\theta_t^c)$ is a one-hot encoding of the action θ_t^c picked by Generator. Thus, $\hat{F}(s_t, \theta_t^c; s_{t-1}, \theta_{t-1}^c) = f(s_t) \cdot i(\theta_t^c) - \gamma^{-1} f(s_{t-1}) \cdot i(\theta_{t-1}^c)$. The action set of Generator is $\Theta := \{0, 1, ..., m\}$ where each element is drawn from $\mathbb{N}$, and g is an MLP $g : \mathbb{R}^d \mapsto \mathbb{R}^m$. Extra details are Sec. 8 of the Appendix.

5 Convergence and Optimality of LIGS

We now show that LIGS converges and that the solution ensures a higher performing agent policies than what would be achieved by solving $\mathfrak{M}$ directly. The addition of Generator's RL process which modifies N agents' rewards during learning can produce convergence issues (Zinkevich et al., 2006). Also to ensure the framework is useful, we must verify that the solution of $\mathcal{G}$ corresponds to solving the MDP, $\mathfrak{M}$. To resolve these issues, we first study the stable point solutions of $\mathcal{G}$. Unlike MDPs, the existence of a solution in Markov policies is not guaranteed for MGs (Blackwell & Ferguson, 1968) and is rarely computable (except for special cases such as *team* and *zero-sum* MGs (Shoham & Leyton-Brown, 2008)). MGs also often have multiple stable points that can be inefficient (Mguni et al., 2019); in $\mathcal{G}$ such stable points would lead to a poor performing agent joint policy. We resolve these challenges with the following scheme:

[I] LIGS preserves the optimal solution of $\mathfrak{M}$.
[II] The MG induced by LIGS has a stable point which is the convergence point of MARL.
[III] LIGS yields a team payoff that is (weakly) greater than that from solving $\mathfrak{M}$ directly.
[IV] LIGS converges to the solution with a linear function approximators.

In what follows, we denote by $\boldsymbol{\Pi} := \times_{i \in \mathcal{N}} \Pi_i$. The results are built under Assumptions 1 - 7 (Sec. 15 of the Appendix) which are standard in RL and stochastic approximation theory. We now prove the result **[I]** which shows the solution to $\mathfrak{M}$ is preserved under the influence of LIGS:

Proposition 1 *The following statements hold:*

i) $\max\limits_{\boldsymbol{\pi} \in \boldsymbol{\Pi}} v^{\boldsymbol{\pi}, g}(s, \cdot) = \max\limits_{\boldsymbol{\pi} \in \boldsymbol{\Pi}} v^{\boldsymbol{\pi}}(s)$, $\forall s \in \mathcal{S}, \forall i \in \mathcal{N}, \forall g$ *where* $v^{\boldsymbol{\pi}}(s) = \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{t=0}^{\infty} \gamma^t R\right]$.

ii) The Generator's optimal policy maximises $v^{\boldsymbol{\pi}}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, \boldsymbol{a}_t)\right]$ *for any* $s \in \mathcal{S}$.

Result (i) says that the agents' problem is preserved under Generator's influence. Moreover the agents' (expected) total return is that from the environment (extrinsic rewards). Result (ii) establishes that Generator's optimal policy induces it to maximise the agents' joint (extrinsic) total return. The result is proven by a careful adaptation of the policy invariance result in (Ng et al., 1999) to our MARL switching control setting where the intrinsic-reward is not added at all states.

Building on Prop. 1, we deduce the following result:

Corollary 1 *LIGS preserves the dec-MDP played by the agents. In particular, let* $(\hat{\boldsymbol{\pi}}, g)$ *be a stable point policy profile*[4] *of the MG induced by LIGS,* $\mathcal{G}$ *then* $\hat{\boldsymbol{\pi}}$ *is a solution to the dec-MDP,* $\mathfrak{M}$.

Therefore, the introduction of Generator does not alter the fundamentals of the problem. Our next task is to prove the existence of a stable point of the MG induced by LIGS and show it is a limit point

[3]This is similar to random network distillation (Burda et al., 2018) however the input is over the space $\mathcal{A} \times \mathcal{S}$.
[4]By stable point profile we mean a Markov perfect equilibrium (Fudenberg & Tirole, 1991).

of a sequence of Bellman operations. To do this we prove that a stable solution of $\mathcal{G}$ exists and that $\mathcal{G}$ has a special property that permits its stable point to be found using dynamic programming. We begin by defining key objects for the analysis:

Given a $V^{\boldsymbol{\pi},g} : \mathcal{S} \times \mathbb{N} \to \mathbb{R}, \ \forall \boldsymbol{\pi} \in \boldsymbol{\Pi}$ and g, define by $\mathcal{M}^{\boldsymbol{\pi},g} V^{\boldsymbol{\pi},g}(s_{\tau_k}, I_{\tau_k}) := R(s_{\tau_k}, \boldsymbol{a}_{\tau_k}) + F^{(\theta_{\tau_k},\theta_{\tau_{k-1}})} + c(I_k, I_{k-1}) + \gamma \sum_{s' \in \mathcal{S}} P(s'; \boldsymbol{a}_{\tau_k}, s) V^{\boldsymbol{\pi},g}(s', I(\tau_{k+1})), \ \forall s_{\tau_k} \in \mathcal{S}$ where $\boldsymbol{a}_{\tau_k} \sim \boldsymbol{\pi}(\cdot|s_{\tau_k})$, $\theta_{\tau_k} \sim g(\cdot|s_{\tau_k})$ and τ_k is a Generator switching time. We define the Bellman operator T of $\mathcal{G}$ by $TV^{\boldsymbol{\pi},g}(s_{\tau_k}, I_{\tau_k}) := \max \Big\{ \mathcal{M}^{\boldsymbol{\pi},g} V^{\boldsymbol{\pi},g}(s_{\tau_k}, I_{\tau_k}), R(s_{\tau_k}, \boldsymbol{a}_{\tau_k}) + \gamma \max_{\boldsymbol{a} \in \boldsymbol{\mathcal{A}}} \sum_{s' \in \mathcal{S}} P(s'; \boldsymbol{a}, s_{\tau_k}) V^{\boldsymbol{\pi},g}(s', I_{\tau_k}) \Big\}$.

The following result establishes that the solution of the MG $\mathcal{G}$, can be computed using RL methods:

Theorem 1 *Let $V : \mathcal{S} \times \boldsymbol{\mathcal{A}} \to \mathbb{R}$, then for any $s \in \mathcal{S}$ the game $\mathcal{G}$ has a stable point given by $\lim_{k\to\infty} T^k V^{\boldsymbol{\pi}} = \sup_{\hat{\boldsymbol{\pi}} \in \boldsymbol{\Pi}} V^{\hat{\boldsymbol{\pi}}} = V^{\boldsymbol{\pi}^\star}$ where $\boldsymbol{\pi}^\star \in \boldsymbol{\Pi}$ is a stable joint policy of $\mathcal{G}$.*

Theorem 1 proves that the MG $\mathcal{G}$ (which is the game that is induced when Generator plays with the N agents) has a stable point which is the limit of a dynamic programming method. In particular, it proves the that the stable point of $\mathcal{G}$ is the limit point of the sequence $T^1V, T^2V, \ldots,$. Crucially, (by Corollary 1) the limit point corresponds to the solution of the dec-MDP $\mathcal{M}$. Theorem 1 is proven by firstly proving that $\mathcal{G}$ has a dual representation as an MDP whose solution corresponds to the stable point of the MG. Theorem 1 enables us to tackle the problem of finding the solution to $\mathcal{G}$ using distributed learning methods i.e. MARL to solve $\mathcal{G}$. Moreover, Prop. 1 indicates by computing the stable point of $\mathcal{G}$ leads to a solution of $\mathfrak{M}$. These results combined prove **[II]**.

Our next result characterises Generator policy $\mathfrak{g}_c$ and the optimal times to activate F. The result yields a key aspect of our algorithm for executing Generator activations of intrinsic rewards:

Proposition 2 *The policy $\mathfrak{g}_c$ is given by the following: $\mathfrak{g}_c(s_t, I_t) = H(\mathcal{M}^{\boldsymbol{\pi},g} V^{\boldsymbol{\pi},g} - V^{\boldsymbol{\pi},g})(s_t, I_t), \ \forall (s_t, I_t) \in \mathcal{S} \times \{0,1\}$, where $V^{\boldsymbol{\pi},g}$ is the solution in Theorem 1 and H is the Heaviside function, moreover Generator's switching times are $\tau_k = \inf\{\tau > \tau_{k-1} | \mathcal{M}^{\boldsymbol{\pi},g} V^{\boldsymbol{\pi},g} = V^{\boldsymbol{\pi},g}\}$.*

In general, introducing intrinsic rewards or shaping rewards may undermine learning and worsen overall performance. We now prove that the LIGS framework introduces an intrinsic reward which yields better performance for the N agents as compared to solving $\mathfrak{M}$ directly (**[III]**).

Theorem 2 *Each agent's expected return $v^{\boldsymbol{\pi},g}$ whilst playing $\mathcal{G}$ is (weakly) higher than the expected return for $\mathfrak{M}$ (without Generator) i.e. $v^{\boldsymbol{\pi},g}(s,\cdot) \geq v^{\boldsymbol{\pi}}(s), \ \forall s \in \mathcal{S}, \ \forall i \in \mathcal{N}$.*

Theorem 2 shows that Generator's influence leads to an improvement in the system performance for the N agents. Note that by Prop. 1, Theorem 2 compares the environment (extrinsic) rewards accrued by the agents so that the presence of Generator increases the total expected environment rewards. We complete our analysis by extending Theorem 1 to capture (linear) function approximators which proves **[IV]**. We first define a *projection* Π by: $\Pi\Lambda := \arg\min_{\bar{\Lambda} \in \{\Phi r | r \in \mathbb{R}^p\}} \left\| \bar{\Lambda} - \Lambda \right\|$ for any function Λ.

Theorem 3 *LIGS converges to the stable point of $\mathcal{G}$, moreover, given a set of linearly independent basis functions $\Phi = \{\phi_1, \ldots, \phi_p\}$ with $\phi_k \in L_2, \forall k$. LIGS converges to a limit point $r^\star \in \mathbb{R}^p$ which is the unique solution to $\Pi\mathfrak{F}(\Phi r^\star) = \Phi r^\star$ where $\mathfrak{F}\Lambda := \hat{R} + \gamma P \max\{\mathcal{M}\Lambda, \Lambda\}$. Moreover, $r^\star$ satisfies: $\|\Phi r^\star - Q^\star\| \leq (1-\gamma^2)^{-1/2} \|\Pi Q^\star - Q^\star\|$.*

The theorem establishes the convergence of LIGS to a stable point (of $\mathcal{G}$) with the use of linear function approximators. The second statement bounds the proximity of the convergence point by the smallest approximation error that can be achieved given the choice of basis functions.

6 Experiments

We performed a series of experiments to test if LIGS: **1.** learns to efficiently coordinate the agents' joint exploration **2.** Optimises convergence points by inducing coordination. **3.** Handles sparse

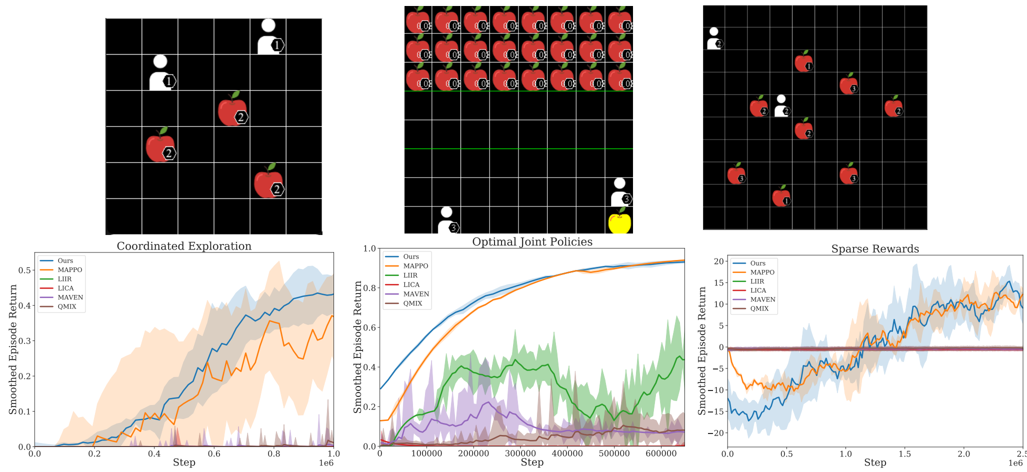

Figure 1: *Left.* Coordinated Exploration. *Centre*. Optimal joint policies. *Right.* Sparse rewards.

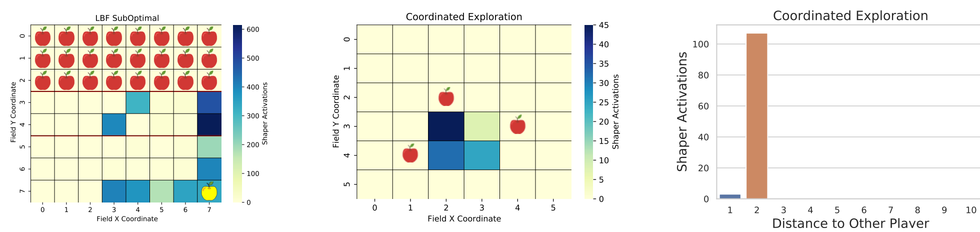

Figure 2: *Left.* Heatmap of Exp. 2 showing where Generator adds rewards. *Centre.* Corresponding heatmap for Exp. 1. *Right.* Plot of distance to other agent when Generator activates rewards in Exp 1

rewards environments. In all tasks, we compared the performance of LIGS against leading MARL solvers MAPPO (Yu et al., 2021), QMIX (Rashid et al., 2018); intrinsic reward MARL algorithms LIIR (Du et al., 2019), LICA (Zhou, Liu, et al., 2020) and a leading MARL exploration algorithm MAVEN (Mahajan et al., 2019). We then compared LIGS against these baselines in Cooperative Foraging Tasks (Papoudakis et al., 2020) and StarCraft Micromanagement II (Samvelyan et al., 2019).

6.1 Cooperative Foraging Tasks

Experiment 1: Coordinated exploration. We tested our first claim that LIGS promotes coordinated exploration among agents. To investigate this, we used a version of the level-based foraging environment (Papoudakis et al., 2020) as follows: there are n agents each with level a_i. Moreover, there are 3 apples with level K such that $\sum_{i=1}^{N} a_i = K$. The only way to collect the reward is if all agents collectively enact the collect action when they are beside an apple. This is a challenging joint-exploration problem since to obtain the reward, the agents must collectively explore joint actions across the state space (rapidly) to discover that simultaneously executing collect near an apple produces rewards. To increase the difficulty, we added a penalty for the agents failing to coordinate in collecting the apples. For example, if only one agent uses the collect action near an apple, it gets a negative reward. This results in a non-monotonic reward structure. Fig. 1 shows the performance curves. Fig, 1 shows LIGS demonstrates superior performance over the baselines.

Experiment 2: Optimal joint policies. We next tested our second claim that LIGS can promote convergence to joint policies that achieve higher system rewards. To do this, we constructed a challenging experiment in which the agents must avoid converging to suboptimal policies that deliver positive but low rewards. In this experiment, the grid is divided horizontally in three sections; top, middle and bottom. All grid locations in the top section give a small reward r/n to the agent visiting them where n is the number of tiles in the each section. The middle section does not give any rewards. The bottom section rewards the agents depending on their relative positions. If one agent is at the top and the other at the bottom, the agent at the bottom receives a reward $-r/n$ each time the other agent receives a reward. If both agents are at the bottom, then one of the tiles in this section will give a

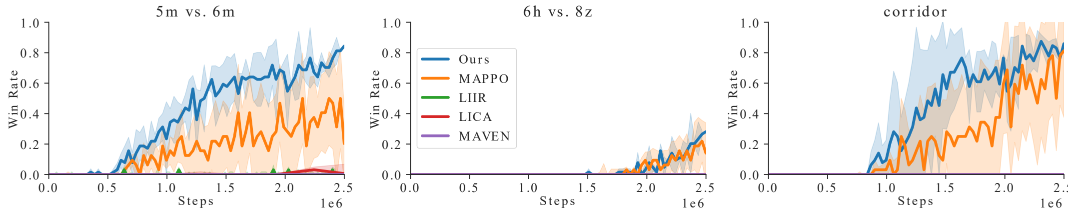

Figure 3: *Median win rate over the course of learning on SMAC.* LIGS outperforms the baselines on all maps. LIIR, LICA, and MAVEN are generally not visible as their win rate is negligible.

reward $R, r/2 < R < r$ to both agents. The bottom section gives no reward otherwise. The agents start in the middle section and as soon as they cross to one section they cannot return to the middle. As is shown in Fig. 1, LIGS learns to acquire rewards rapidly in comparison to the baselines with MAPPO requiring around 400k episodes to match the rewards produced by LIGS.

Experiment 3: Sparse rewards. We tested our claim that LIGS can promote learning in MAS with sparse rewards. We simulate a sparse reward setting using a competitive game between two teams of agents. One team is controlled by LIGS while the other actions of the agents belonging to the other team are determined by a fixed policy. The goal is to collect the apple faster than the opposing team. Collecting the apple results in a reward of 1, and rewards are 0 otherwise. This is a challenging sparse reward since informative reward signals occur only apple when the apple is collected. As is shown in Fig. 1 both LIGS and MAPPO perform well on the sparse rewards environment, whilst the other baselines are all unable to learn any behaviour on this environment.

We next investigated the workings of the LIGS framework. We studied the locations where Generator added intrinsic rewards in Experiments 1 and 2. As is shown in the heatmap visualisation in Fig. 2, for Experiment 2, we observe that Generator learns to add intrinsic rewards that guide the agents towards the optimal reward (bottom left) and away from the suboptimal rewards at the top (where some other baselines converge). This supports our claim that LIGS learns to guide the agents towards jointly optimal policies. Moreover, as Fig. 2 shows, LIGS's switching mechanism means that Generator only adds intrinsic rewards as the most useful locations for guiding the agents towards their target. For Experiment 1, Fig. 2 shows that Generator learns to guide the agents towards the apple which delivers the high rewards. Crucially Fig. 2 (Right) demonstrates a striking benefit of the LIGS framework, namely it only activates the intrinsic rewards around the apple when *both* agents are at most 2 cells away from the apple. Since the agents receive positive rewards only when they arrive at the apple simultaneously, this ensures the agents are encouraged to coordinate their arrival and receive the maximal positive rewards and avoids encouraging arrivals at times that lead to penalties.

6.2 Learning Performance in StarCraft Multi-Agent Challenge

To study the performance of LIGS in highly complex environments, we compared its performance with the aforementioned baselines on three StarCraft Multi-Agent Challenge (SMAC) maps (Samvelyan et al., 2019) *5m vs. 6m* (hard), *6h vs. 8z*, and *Corridor*. These maps vary in a range of MARL attributes such as number of units to control, environment reward density, unit action sets, and (partial)-observability. In Fig. 3, we report our results showing 'Win Rate' vs 'Steps'. These curves are generated by computing the median win rate (vs the opponent) of the agent at regular intervals during learning. We ran 3 of each algorithm (further setup details are in the Supp. material Sec 13). LIGS outperforms the baselines in all maps. In *5m vs. 6m*, the baselines do not approach the performance of LIGS. In *Corridor* MAPPO requires over an extra million steps to match the performance of LIGS. In *6h vs. 8z*, LIGS slightly outperforms the baselines. In summary, LIGS shows performance gains over all baselines in SMAC maps which encompass diverse MAS attributes.

7 Conclusion

We introduced LIGS, a novel framework for generating intrinsic rewards which significantly boosts performance of MARL algorithms. Central to LIGS is a powerful adaptive learning mechanism that generates intrinsic rewards according to the task and the MARL learners' joint behaviour. Our experiments show LIGS induces superior performance in MARL algorithms in a range of tasks.

REFERENCES

Bayraktar, E., & Egami, M. (2010). On the one-dimensional optimal switching problem. *Mathematics of Operations Research*, *35*(1), 140–159.

Benveniste, A., Métivier, M., & Priouret, P. (2012). *Adaptive algorithms and stochastic approximations* (Vol. 22). Springer Science & Business Media.

Blackwell, D., & Ferguson, T. S. (1968). The big match. *The Annals of Mathematical Statistics*, *39*(1), 159–163.

Burda, Y., Edwards, H., Storkey, A., & Klimov, O. (2018). Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*.

Devlin, S., & Kudenko, D. (2011). Theoretical considerations of potential-based reward shaping for multi-agent systems. *The 10th International Conference on Autonomous Agents and Multiagent Systems*, 225–232.

Devlin, S., & Kudenko, D. (2016). Plan-based reward shaping for multi-agent reinforcement learning. *The Knowledge Engineering Review*, *31*(1), 44–58.

Devlin, S., Kudenko, D., & Grześ, M. (2011). An empirical study of potential-based reward shaping and advice in complex, multi-agent systems. *Advances in Complex Systems*, *14*(02), 251–278.

Devlin, S. M., & Kudenko, D. (2012). Dynamic potential-based reward shaping. *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems*, 433–440.

Dilokthanakul, N., Kaplanis, C., Pawlowski, N., & Shanahan, M. (2019). Feature control as intrinsic motivation for hierarchical reinforcement learning. *IEEE transactions on neural networks and learning systems*, *30*(11), 3409–3418.

Du, Y., Han, L., Fang, M., Liu, J., Dai, T., & Tao, D. (2019). Liir: Learning individual intrinsic reward in multi-agent reinforcement learning.

Foerster, J. N., Farquhar, G., Afouras, T., Nardelli, N., & Whiteson, S. (2018). Counterfactual multi-agent policy gradients. *Thirty-second AAAI conference on artificial intelligence*.

Fudenberg, D., & Tirole, J. (1991). Tirole: Game theory. *MIT Press*, *726*, 764.

Harutyunyan, A., Devlin, S., Vrancx, P., & Nowé, A. (2015). Expressing arbitrary reward functions as potential-based advice. *Proceedings of the AAAI Conference on Artificial Intelligence*, *29*.

Hosu, I.-A., & Rebedea, T. (2016). Playing atari games with deep reinforcement learning and human checkpoint replay. *arXiv preprint arXiv:1607.05077*.

Hüttenrauch, M., Šošić, A., & Neumann, G. (2017). Guided deep reinforcement learning for swarm systems. *arXiv preprint arXiv:1709.06011*.

Jaakkola, T., Jordan, M. I., & Singh, S. P. (1994). Convergence of stochastic iterative dynamic programming algorithms. *Advances in neural information processing systems*, 703–710.

Kulkarni, T. D., Narasimhan, K., Saeedi, A., & Tenenbaum, J. (2016). Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation. *Advances in neural information processing systems*, *29*, 3675–3683.

Macua, S. V., Zazo, J., & Zazo, S. (2018). Learning parametric closed-loop policies for markov potential games. *arXiv preprint arXiv:1802.00899*.

Mahajan, A., Rashid, T., Samvelyan, M., & Whiteson, S. (2019). Maven: Multi-agent variational exploration. *arXiv preprint arXiv:1910.07483*.

Mannion, P., Devlin, S., Duggan, J., & Howley, E. (2018). Reward shaping for knowledge-based multi-objective multi-agent reinforcement learning. *The Knowledge Engineering Review*, *33*.

Mannion, P., Devlin, S., Mason, K., Duggan, J., & Howley, E. (2017). Policy invariance under reward transformations for multi-objective reinforcement learning. *Neurocomputing*, *263*, 60–73.

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

Mguni, D., Wu, Y., Du, Y., Yang, Y., Wang, Z., Li, M., Wen, Y., Jennings, J., & Wang, J. (2021). Learning in nonzero-sum stochastic games with potentials. *ICML*.

Mguni, D. (2018). A viscosity approach to stochastic differential games of control and stopping involving impulsive control. *arXiv preprint arXiv:1803.11432*.

Mguni, D. (2019). Cutting your losses: Learning fault-tolerant control and optimal stopping under adverse risk. *arXiv preprint arXiv:1902.05045*.

Mguni, D., Jennings, J., & de Cote, E. M. (2018). Decentralised learning in systems with many, many strategic agents. *Thirty-Second AAAI Conference on Artificial Intelligence*.

Mguni, D., Jennings, J., Sison, E., Valcarcel Macua, S., Ceppi, S., & Munoz de Cote, E. (2019). Coordinating the crowd: Inducing desirable equilibria in non-cooperative systems. *Proceedings of the 18th international conference on autonomous agents and multiagent systems* (pp. 386–394). International Foundation for Autonomous Agents; Multiagent Systems.

Ng, A. Y., Harada, D., & Russell, S. (1999). Policy invariance under reward transformations: Theory and application to reward shaping. *ICML*, *99*, 278–287.

Øksendal, B. (2003). Stochastic differential equations. *Stochastic differential equations* (pp. 65–84). Springer.

Papoudakis, G., Christianos, F., Schäfer, L., & Albrecht, S. V. (2020). Comparative evaluation of multi-agent deep reinforcement learning algorithms. *arXiv preprint arXiv:2006.07869*.

Pathak, D., Agrawal, P., Efros, A. A., & Darrell, T. (2017). Curiosity-driven exploration by self-supervised prediction. *International conference on machine learning*, 2778–2787.

Pesce, E., & Montana, G. (2020). Improving coordination in small-scale multi-agent deep reinforcement learning through memory-driven communication. *Machine Learning*, 1–21.

Rashid, T., Samvelyan, M., Schroeder, C., Farquhar, G., Foerster, J., & Whiteson, S. (2018). Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. *International Conference on Machine Learning*, 4295–4304.

Roughgarden, T., & Tardos, E. (2007). Introduction to the inefficiency of equilibria. *Algorithmic game theory*, *17*, 443–459.

Sadeghlou, M., Akbarzadeh-T, M. R., & Naghibi-S, M. B. (2014). Dynamic agent-based reward shaping for multi-agent systems. *2014 Iranian Conference on Intelligent Systems (ICIS)*, 1–6.

Samvelyan, M., Rashid, T., De Witt, C. S., Farquhar, G., Nardelli, N., Rudner, T. G., Hung, C.-M., Torr, P. H., Foerster, J., & Whiteson, S. (2019). The starcraft multi-agent challenge. *arXiv preprint arXiv:1902.04043*.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. *CoRR*, *abs/1707.06347*.

Shoham, Y., & Leyton-Brown, K. (2008). *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press.

Son, K., Kim, D., Kang, W. J., Hostallero, D. E., & Yi, Y. (2019). Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. *International Conference on Machine Learning*, 5887–5896.

Sunehag, P., Lever, G., Gruslys, A., Czarnecki, W. M., Zambaldi, V., Jaderberg, M., Lanctot, M., Sonnerat, N., Leibo, J. Z., Tuyls, K., et al. (2017). Value-decomposition networks for cooperative multi-agent learning. *arXiv preprint arXiv:1706.05296*.

Tijsma, A. D., Drugan, M. M., & Wiering, M. A. (2016). Comparing exploration strategies for q-learning in random stochastic mazes. *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1–8.

Tsitsiklis, J. N., & Van Roy, B. (1999). Optimal stopping of markov processes: Hilbert space theory, approximation algorithms, and an application to pricing high-dimensional financial derivatives. *IEEE Transactions on Automatic Control*, *44*(10), 1840–1851.

Viseras, A., Wiedemann, T., Manss, C., Magel, L., Mueller, J., Shutin, D., & Merino, L. (2016). Decentralized multi-agent exploration with online-learning of gaussian processes. *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 4222–4229.

Wang, J., Ren, Z., Han, B., Ye, J., & Zhang, C. (2020). Towards understanding linear value decomposition in cooperative multi-agent q-learning. *arXiv preprint arXiv:2006.00587*.

Wang, J., Ren, Z., Liu, T., Yu, Y., & Zhang, C. (2020). Qplex: Duplex dueling multi-agent q-learning. *arXiv preprint arXiv:2008.01062*.

Yang, Y., & Wang, J. (2020). An overview of multi-agent reinforcement learning from game theoretical perspective. *arXiv preprint arXiv:2011.00583*.

Yang, Y., Wen, Y., Wang, J., Chen, L., Shao, K., Mguni, D., & Zhang, W. (2020). Multi-agent determinantal q-learning. *International Conference on Machine Learning*, 10757–10766.

Yu, C., Velu, A., Vinitsky, E., Wang, Y., Bayen, A., & Wu, Y. (2021). The surprising effectiveness of mappo in cooperative, multi-agent games. *arXiv preprint arXiv:2103.01955*.

Zheng, Z., Oh, J., & Singh, S. (2018). On learning intrinsic rewards for policy gradient methods. *arXiv preprint arXiv:1804.06459*.

Zhou, M., Liu, Z., Sui, P., Li, Y., & Chung, Y. Y. (2020). Learning implicit credit assignment for multi-agent actor-critic. *arXiv e-prints*, arXiv–2007.

Zhou, M., Luo, J., Villella, J., Yang, Y., Rusu, D., Miao, J., Zhang, W., Alban, M., Fadakar, I., Chen, Z., et al. (2020). Smarts: Scalable multi-agent reinforcement learning training school for autonomous driving. *arXiv preprint arXiv:2010.09776*.

Zinkevich, M., Greenwald, A., & Littman, M. (2006). Cyclic equilibria in markov games. *Advances in Neural Information Processing Systems*, *18*, 1641.

Appendix

Table of Contents

8 Algorithm

Algorithm 1: Learnable **I**ntrinsic-Reward **G**eneration **S**election algorithm (LIGS)

Input: Environment E

Initial agent policies $\boldsymbol{\pi}_0 = (\pi_0^1, \dots \pi_0^N)$ with parameters $\theta_{\pi_0^1}, \dots \theta_{\pi_N^1}$, Initial Generator switch policy $\mathfrak{g}_{c_0}$ with parameters $\theta_{\mathfrak{g}_{c_0}}$, Initial Generator action policy g_0 with parameters θ_{g_0}, Randomly initialised fixed neural network $\phi(\cdot, \cdot)$, Neural networks h (fixed) and $\hat{h}$ for Augmented RND with parameter $\theta_{\hat{h}}$, Buffer B, Number of rollouts N_r, rollout length T, Number of mini-batch updates N_u, Switch cost c, discount factor γ, learning rate α.

Output: Optimised agent policies $\boldsymbol{\pi}^\star = (\pi^{\star,1}, \dots, \pi^{\star,N})$

$\boldsymbol{\pi} = (\pi^1, \dots, \pi^N), g, \mathfrak{g}_c \leftarrow \boldsymbol{\pi}_0, g_0, \mathfrak{g}_{c_0}$
for $n = 1, N_r$ **do**
  // Collect rollouts
  for $t = 1, T$ **do**
    Get environment states s_t from E
    Sample $\boldsymbol{a}_t = (a_t^1, \dots, a_t^N)$ from $(\pi^1(s_t), \dots, \pi^N(s_t))$
    Apply action $\boldsymbol{a}_t$ to environment E, get rewards $\boldsymbol{r}_t = (r_t^1, \dots, r_t^N)$ and next state s_{t+1}
    Sample q_t from $\mathfrak{g}_c(s_t)$ **// Switching control**
    if $q_t = 1$ **then**
      Sample θ_t^c from $g(s_t)$
      Sample θ_{t+1}^c from $g(s_{t+1})$
      $f_t^i = \gamma\phi(s_{t+1}, \theta_{t+1}^c) - \phi(s_t, \theta_t^c)$ **// Calculate** $F(s_t, \theta_t^c, s_{t+1}, \theta_{t+1}^c)$
    else
      $\theta_t^c, f_t^i = 0, 0$ **// Dummy values**
    Append $(s_t, \boldsymbol{a}_t, g_t, \theta_t^c, \boldsymbol{r}_t, f_t^i, s_{t+1})$ to B
  for $u = 1, N_u$ **do**
    Sample data $(s_t, \boldsymbol{a}_t, g_t, \theta_t^c, \boldsymbol{r}_t, f_t^i, s_{t+1})$ from B
    if $g_t = 1$ **then**
      Set reward to $\boldsymbol{r}_t^s = \boldsymbol{r}_t + f_t^i$
    else
      Set reward to $\boldsymbol{r}_t^s = \boldsymbol{r}_t$
    // Update Augmented RND
    $\text{Loss}_{\text{RND}} = ||h(s_t, \boldsymbol{a}_t) - \hat{h}(s_t, \boldsymbol{a}_t)||^2$
    $\theta_{\hat{h}} \leftarrow \theta_{\hat{h}} - \alpha\nabla\text{Loss}_{\text{RND}}$
    // Update Generator
    $l_t = ||h(s_t, \boldsymbol{a}_t) - \hat{h}(s_t)||^2$ **// Compute** $L(s_t, \boldsymbol{a}_t)$
    $c_t = cg_t$
    Compute Loss_g using $(s_t, a_t, g_t, c_t, \boldsymbol{r}_t, f_t^i, l_t, s_{t+1})$ using PPO loss **// Section 4.1**
    Compute $\text{Loss}_{\mathfrak{g}_c}$ using $(s_t, a_t, g_t, c_t, \boldsymbol{r}_t, f_t^i, l_t, s_{t+1})$ using PPO loss **// Section 4.1**
    $\theta_g \leftarrow \theta_g - \alpha\nabla\text{Loss}_g$
    $\theta_{\mathfrak{g}_c} \leftarrow \theta_{\mathfrak{g}_c} - \alpha\nabla\text{Loss}_{\mathfrak{g}_c}$
    // Update agent j**, for each** $j \in 1, \dots, N$
    Compute Loss_{π^j} using $(s_t, \boldsymbol{a}_t, r_t^{j,s} := r_t^j + f_t^i, s_{t+1})$ using PPO loss **// Section 4.1**
    $\theta_{\pi^j} \leftarrow \theta_{\pi^j} - \alpha\nabla\text{Loss}_{\pi^j}$

9 ABLATION STUDY: PLUG & PLAY

In order to validate our claim that LIGS freely adopts RL learners, we tested the ability of LIGS to boost performance in a complex coordination task using independent Proximal policy optimization algorithm (IPPO) (Schulman et al., 2017) as the base learner. In this experiment, two agents are spawned at opposite sides of the grid. The red agent is spawned in the left hand side and the blue agent is spawned in the right hand side of the grid in Fig. 4. The goal of the agents is to arrive at their corresponding goal states (indicated by the coloured square, where the colour corresponds to the agent whose goal state it is) at the other side of the grid. Upon arriving at their goal state the agents receive their reward. However, the task is made difficult by the fact that only one agent can pass through the corridor at a time. Therefore, in this setup, the only way for the agents to complete the task is for the agents to successfully coordinate, i.e. one agent is required to allow the other agent to pass through before attempting to traverse the corridor.

It is known that independent learners in general, struggle to solve such tasks since their ability to coordinate systems of RL learners is lacking (Yang et al., 2020). This is demonstrated in Fig. 4 (right) which displays the performance curve of for IPPO which fails to score above 0. As claimed, when incorporated into the LIGS framework, the agents succeed in coordinating to solve the task. This is indicated by the performance of IPPO + LIGS (blue).

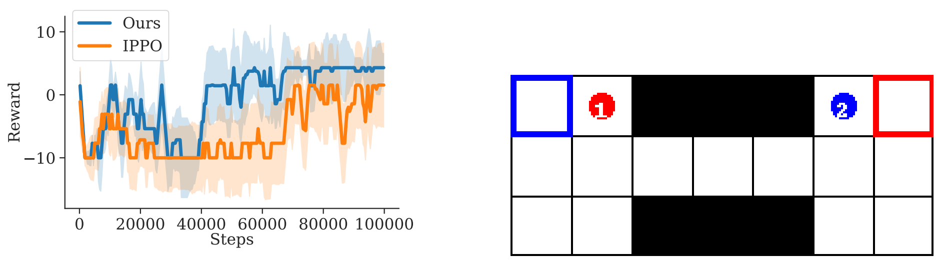

Figure 4: *Left.* Performance curves for IPPO and IPPO with LIGS. *Right.* Coordination environment.

10 Results on Additional StarCraft Micromanagement II Maps

Here we present results on two more maps: MMM2 and 3s5z vs 3s6z As can be seen in Fig. 5, LIGS outperforms the leading baseline MAPPO in all maps yielding a 50% improvement in win rate over MAPPO in MMM2.

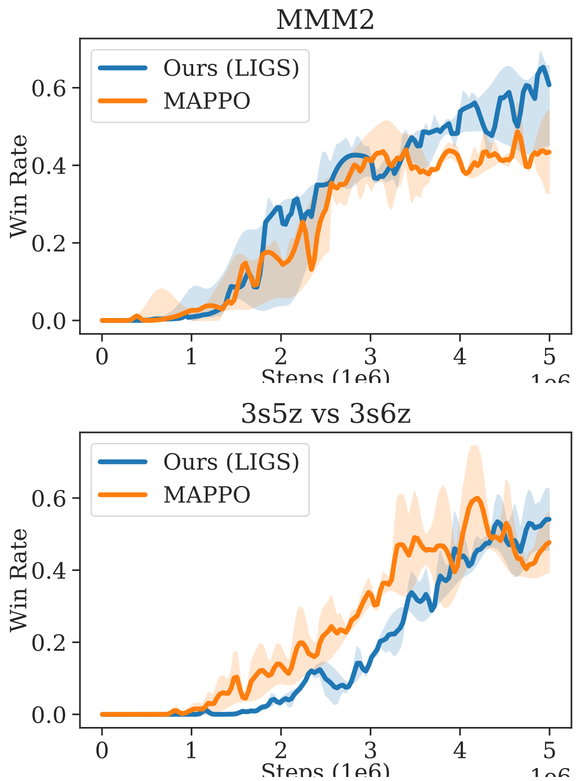

Figure 5: Performance of LIGS in additional SMAC maps

11 Flexibility of LIGS to Accommodate different Exploration Bonus Terms L

To demonstrate the robustness of our method to different choices of exploration bonus terms in Generator's objective, we conducted an Ablation study on the L-term (c.f. Equation 3) where we replaced the RND L term with a basic count-based exploration bonus. To exemplify the high degree of flexibility, we replaced the RND with a simple exploration bonus term $L(s) = \frac{1}{\text{Count(s)}+1}$ for any given state $s \in \mathcal{S}$ where $\text{Count}(s)$ refers to a simple count of the number of times the state s has been visited. We conducted the Ablation study on all three Foraging environments presented in Sec. 6.1. We note that despite the simplicity of the count-based measure, generally the performance of both versions of LIGS is comparable and in fact the count-based variant is superior to the RND version for the joint exploration environment.

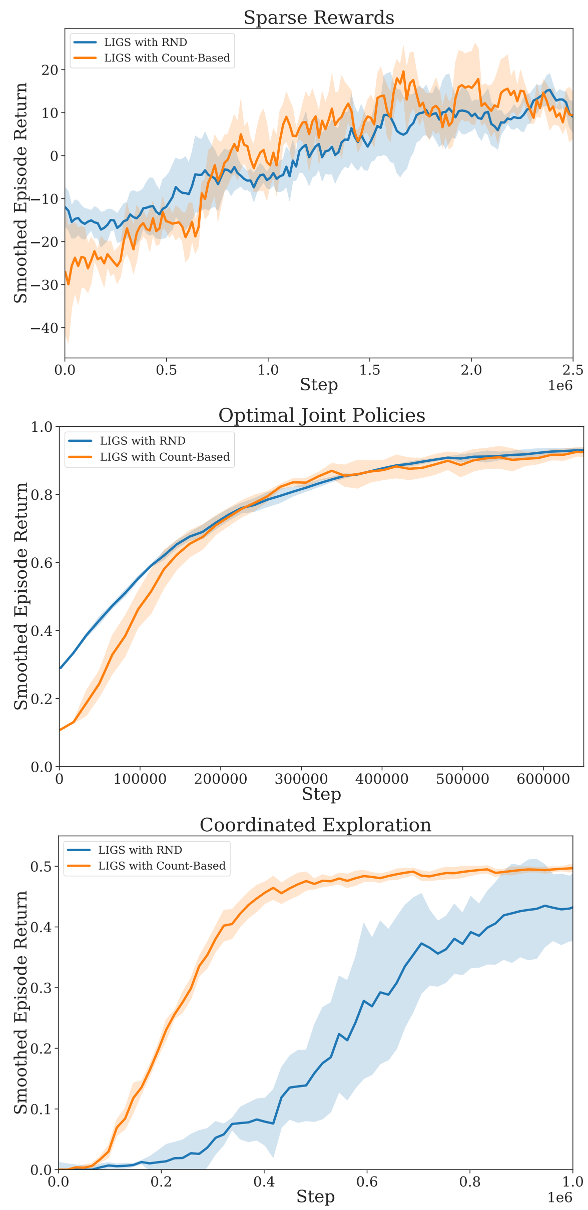

Figure 6: Performance of LIGS compared with the exploration bonus replaced by count-based method on the three tasks in the Foraging environment.

12 FURTHER EXPERIMENT DEMONSTRATING LIGS IMPROVED USE OF EXPLORATION BONUSES.

As we have shown above, LIGS can accommodate a variety of exploration bonuses and perform well. Here, we did a experiment to further justify using LIGS against simpler exploration bonus methods. We compared LIGS against and MAPPO with an RND intrinsic reward in the agents' objectives (MAPPO+RND) and vanilla MAPPO. Fig. 7 shows performance of these two methods on coordination environment shown in Fig. 4. We note that LIGS markedly outperforms both MAPPO+RND and vanilla MAPPO. Due to the added benefit of switching controls and intrinsic reward selection performed by Generator, we observe that LIGS is able to significantly augment the benefits of applying RND directly to the agents' objectives.

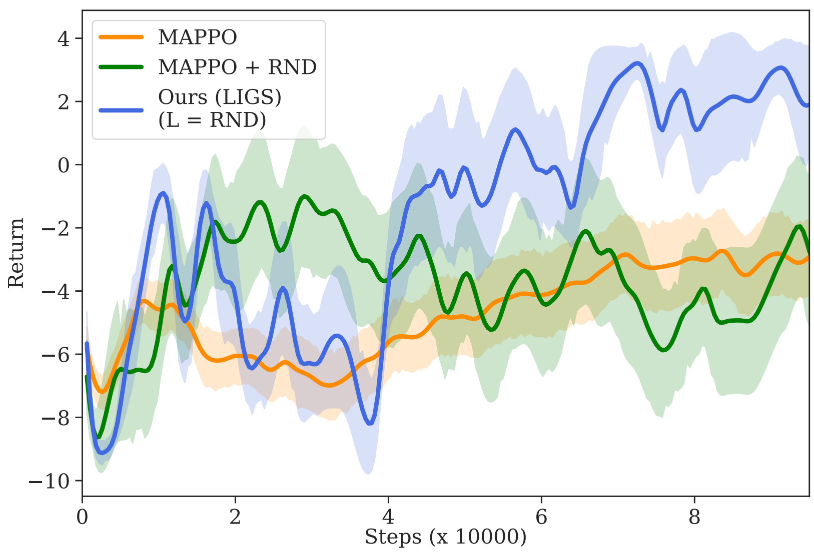

Figure 7: Performance curves for LIGS, MAPPO with RND intrinsic rewards and vanilla MAPPO. The additional machinery of switching-controls and intrinsic reward selection allows LIGS to make better use of exploration bonuses. In this case, LIGS demonstrates significant improvement over MAPPO with RND intrinsic rewards.

13 FURTHER IMPLEMENTATION DETAILS

Details of Generator and F (intrinsic-reward)

Object	Description
f	Fixed feed forward NN that maps $\mathbb{R}^d \mapsto \mathbb{R}^m$ [512, ReLU, 512, ReLU, 512, m]
Θ	Discrete action set which is size of output of f, i.e., Θ is set of integers $\{1, ..., m\}$
g	Fixed feed forward NN that maps $\mathbb{R}^d \mapsto \mathbb{R}^m$ [512, ReLU, 512, ReLU, 512, m]
Intrinsic-reward function ϕ	$\phi(s, \theta^c) = f(s) \cdot \theta^c$
F	$\gamma\phi(s_{t+1}, \theta^c_{t+1})$ - $\phi(s_t, \theta^c_t)$, $\quad \gamma = 0.95$

d=Dimensionality of states; $m \in \mathbb{N}$ - tunable free parameter.

In all experiments we used the above form of F as follows: a state s_t is input to the g network and the network outputs logits p_t. We softmax and sample from p_t to obtain the action θ^c_t. This action is one-hot encoded. Then, the action θ^c_t is multiplied with $f(s_t)$ to compute the second term of F. A similar process is used to compute the first term. In this way the policy of Generator chooses the intrinsic-reward.

13.1 HYPERPARAMETER SETTINGS

In the table below we report all hyperparameters used in our experiments. Hyperparameter values in square brackets indicate ranges of values that were used for performance tuning.

Clip Gradient Norm	1
γ_E	0.99
λ	0.95
Learning rate	1x10^{-4}
Number of minibatches	4
Number of optimisation epochs	4
Number of parallel actors	16
Optimisation algorithm	Adam
Rollout length	128
Sticky action probability	0.25
Use Generalized Advantage Estimation	True
Coefficient of extrinsic reward	[1, 5]
Coefficient of intrinsic reward	[1, 2, 5, 10, 20, 50]
Generator discount factor	0.99
Probability of terminating option	[0.5, 0.75, 0.8, 0.9, 0.95]
L function output size	[2, 4, 8, 16, 32, 64, 128, 256]

14 GENERATOR TERMINATION TIMES

There are various possibilities for the *termination* times $\{\tau_{2k}\}$ (recall that $\{\tau_{2k+1}\}$ are the times which the F is *switched on* using $\mathfrak{g}_c$). One is for Generator to determine the sequence. Another is to build a construction of $\{\tau_{2k}\}$ that directly incorporates the information gain that a state visit provides: let $w : \Omega \to \{0, 1\}$ be a random variable with $\Pr(w = 1) = p$ and $\Pr(w = 0) = 1 - p$ where $p \in]0, 1]$. Then for any $k = 1, 2, \ldots,$ and denote by $\Delta L(s_{\tau_k}) := L(s_{\tau_k}) - L(s_{\tau_k - 1})$, then we can set:

$$I(s_{\tau_{2k+1}+j}) = \begin{cases} I(s_{\tau_{2k+1}}), & \text{if } w\Delta L(s_{\tau_k+j}) > 0, \\ I(s_{\tau_{2k+2}}), & w\Delta L(s_{\tau_k+j}) \leq 0. \end{cases} \tag{4}$$

To explain, since $\{\tau_{2k}\}_{k\geq 0}$ are the times at which F is switched off then if F is deactivated at exactly after j time steps then $I(s_{\tau_{2k+1}+l}) = I(s_{\tau_{2k+1}})$ for any $0 \leq l < j$ and $I(s_{\tau_{2k+1}+j}) = I(s_{\tau_{2k+2}})$. We now see that (4) terminates F when either the random variable w attains a 0 or when $\Delta L(s_{\tau_k+j}) \leq 0$ which occurs when the exploration bonus in the current state is lower than that of the previous state.

15 NOTATION & ASSUMPTIONS

We assume that $\mathcal{S}$ is defined on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and any $s \in \mathcal{S}$ is measurable with respect to the Borel σ-algebra associated with $\mathbb{R}^p$. We denote the σ-algebra of events generated by $\{s_t\}_{t\geq 0}$ by $\mathcal{F}_t \subset \mathcal{F}$. In what follows, we denote by $(\mathcal{V}, \|\|)$ any finite normed vector space and by $\mathcal{H}$ the set of all measurable functions. Where it will not cause confusion (and with a minor abuse of notation) for a given function h we use the shorthand $h^{(\pi^i,\pi^{-i})}(s) = h(s, \pi^i, \pi^{-i}) \equiv \mathbb{E}_{\pi^i,\pi^{-i}}[h(s, a^i, a^{-i})]$.

The results of the paper are built under the following assumptions which are standard within RL and stochastic approximation methods:

Assumption 1 The stochastic process governing the system dynamics is ergodic, that is the process is stationary and every invariant random variable of $\{s_t\}_{t\geq 0}$ is equal to a constant with probability 1.

Assumption 2 The constituent functions of the agents' objectives R, F and L are in L_2.

Assumption 3 For any positive scalar c, there exists a scalar μ_c such that for all $s \in \mathcal{S}$ and for any $t \in \mathbb{N}$ we have: $\mathbb{E}\left[1 + \|s_t\|^c | s_0 = s\right] \leq \mu_c(1 + \|s\|^c)$.

Assumption 4 There exists scalars C_1 and c_1 such that for any function J satisfying $|J(s)| \leq C_2(1 + \|s\|^{c_2})$ for some scalars c_2 and C_2 we have that: $\sum_{t=0}^{\infty} |\mathbb{E}\left[J(s_t)|s_0 = s\right] - \mathbb{E}[J(s_0)]| \leq C_1 C_2(1 + \|s_t\|^{c_1 c_2})$.

Assumption 5 There exists scalars c and C such that for any $s \in \mathcal{S}$ we have that: $|J(s, \cdot)| \leq C(1 + \|s\|^c)$ for $J \in \{r_i, F, L\}$.

We also make the following finiteness assumption on set of switching control policies for Generator:

Assumption 6 For any policy $\mathfrak{g}_c$, the total number of interventions is $K < \infty$.

We lastly make the following assumption on L which can be made true by construction:

Assumption 7 Let $n(s)$ be the state visitation count for a given state $s \in \mathcal{S}$. For any $\boldsymbol{a} \in \boldsymbol{\mathcal{A}}$, the function $L(s, \boldsymbol{a}) = 0$ for any $n(s) \geq M$ where $0 < M \leq \infty$.

16 Proof of Technical Results

We begin the analysis with some preliminary lemmata and definitions which are useful for proving the main results.

Definition 1 *A.1 An operator $T : \mathcal{V} \to \mathcal{V}$ is said to be a* ***contraction*** *w.r.t a norm $\|\cdot\|$ if there exists a constant $c \in [0,1[$ such that for any $V_1, V_2 \in \mathcal{V}$ we have that:*

$$\|TV_1 - TV_2\| \leq c\|V_1 - V_2\|. \tag{5}$$

Definition 2 *A.2 An operator $T : \mathcal{V} \to \mathcal{V}$ is* ***non-expansive*** *if $\forall V_1, V_2 \in \mathcal{V}$ we have:*

$$\|TV_1 - TV_2\| \leq \|V_1 - V_2\|. \tag{6}$$

Lemma 1 *For any $f : \mathcal{V} \to \mathbb{R}, g : \mathcal{V} \to \mathbb{R}$, we have that:*

$$\left\| \max_{a \in \mathcal{V}} f(a) - \max_{a \in \mathcal{V}} g(a) \right\| \leq \max_{a \in \mathcal{V}} \|f(a) - g(a)\|. \tag{7}$$

Proof 1 *We restate the proof given in (Mguni, 2019):*

$$f(a) \leq \|f(a) - g(a)\| + g(a) \tag{8}$$

$$\implies \max_{a \in \mathcal{V}} f(a) \leq \max_{a \in \mathcal{V}} \{\|f(a) - g(a)\| + g(a)\} \leq \max_{a \in \mathcal{V}} \|f(a) - g(a)\| + \max_{a \in \mathcal{V}} g(a). \tag{9}$$

Deducting $\max_{a \in \mathcal{V}} g(a)$ from both sides of (9) yields:

$$\max_{a \in \mathcal{V}} f(a) - \max_{a \in \mathcal{V}} g(a) \leq \max_{a \in \mathcal{V}} \|f(a) - g(a)\|. \tag{10}$$

After reversing the roles of f and g and redoing steps (8) - (9), we deduce the desired result since the RHS of (10) is unchanged.

Lemma 2 *A.4 The probability transition kernel P is non-expansive, that is:*

$$\|PV_1 - PV_2\| \leq \|V_1 - V_2\|. \tag{11}$$

Proof 2 *The result is well-known e.g. (Tsitsiklis & Van Roy, 1999). We give a proof using the Tonelli-Fubini theorem and the iterated law of expectations, we have that:*

$$\|PJ\|^2 = \mathbb{E}\left[(PJ)^2[s_0]\right] = \mathbb{E}\left(\left[\mathbb{E}\left[J[s_1]|s_0\right]\right)^2\right] \leq \mathbb{E}\left[\mathbb{E}\left[J^2[s_1]|s_0\right]\right] = \mathbb{E}\left[J^2[s_1]\right] = \|J\|^2,$$

where we have used Jensen's inequality to generate the inequality. This completes the proof.

Proof of Prop. 1

Proof 3 (Proof of Prop. 1) *To prove the proposition it suffices to prove that the term $\sum_{t=0}^{T} \gamma^t F(\theta_t, \theta_{t-1}) I(t)$ converges to 0 in the limit as $T \to \infty$. As in classic potential-based reward shaping (Ng et al., 1999), central to this observation is the telescoping sum that emerges by construction of F.*

First recall $v^{\boldsymbol{\pi},g}(s, I_0)$, for any $(s, I_0) \in \mathcal{S} \times \{0,1\}$ is given by:

$$v_i^{\boldsymbol{\pi},g}(s, I_0) = \mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty} \gamma^t \left\{R(s_t, \boldsymbol{a}_t) + \hat{F}(s_t, \theta_t^c; s_{t-1}, \theta_{t-1}^c) I_t\right\}\right] \tag{12}$$

$$= \mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, \boldsymbol{a}_t) + \sum_{t=0}^{\infty} \gamma^t \hat{F}(s_t, \theta_t^c; s_{t-1}, \theta_{t-1}^c) I_t\right] \tag{13}$$

$$= \mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, \boldsymbol{a}_t)\right] + \mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty} \gamma^t \hat{F}(s_t, \theta_t^c; s_{t-1}, \theta_{t-1}^c)) I_t\right] \tag{14}$$

Hence it suffices to prove that $\mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty}\gamma^t\hat{F}(s_t,\theta^c_t;s_{t-1},\theta^c_{t-1}))I_t\right]=0$.

Recall there a number of time steps that elapse between τ_k *and* τ_{k+1}*, now*

$$
\begin{aligned}
&\sum_{t=0}^{\infty}\gamma^t\hat{F}(s_t,\theta^c_t;s_{t-1},\theta^c_{t-1}))I(t)\\
&=\sum_{t=\tau_1}^{\tau_2}\gamma^t\phi(s_t,\theta^c_t)-\gamma^{t-1}\phi(s_{t-1},\theta^c_{t-1})+\sum_{t=\tau_3}^{\tau_4}\gamma^t\phi(s_t,\theta^c_t)-\gamma^{t-1}\phi(s_{t-1},\theta^c_{t-1})\\
&\quad+\ldots+\sum_{t=\tau_{(2k-1)}}^{\tau_{2k}}\gamma^t\phi(s_t,\theta^c_t)-\gamma^{t-1}\phi(s_{t-1},\theta^c_{t-1})+\ldots+\\
&=\sum_{t=\tau_1+1}^{\tau_2}\gamma^t\phi(s_t,\theta^c_t)-\gamma^{t-1}\phi(s_{t-1},\theta^c_{t-1})+\gamma^{\tau_1}\phi(s_{\tau_1},\theta^c_{\tau_1})\\
&\quad+\sum_{t=\tau_3+1}^{\tau_4}\gamma^t\phi(s_t,\theta^c_t)-\gamma^{t-1}\phi(s_{t-1},\theta^c_{t-1})+\gamma^{\tau_3}\phi(s_{\tau_3},\theta^c_{\tau_3})\\
&\quad+\ldots+\sum_{t=\tau_{(2k-1)}+1}^{\tau_{2k}}\gamma^t\phi(s_t,\theta^c_t)-\gamma^{t-1}\phi(s_{t-1},\theta^c_{t-1})+\gamma^{2k+1}\phi(s_{\tau_{2k+1}},\theta^c_{\tau_{2k+1}})+\ldots+\\
&=\sum_{t=\tau_1}^{\tau_2-1}\gamma^{t+1}\phi(s_{t+1},\theta^c_{t+1})-\gamma^t\phi(s_t,\theta^c_t)+\gamma^{\tau_1}\phi(s_{\tau_1},\theta^c_{\tau_1})\\
&+\sum_{t=\tau_3}^{\tau_4-1}\gamma^{t+1}\phi(s_{t+1},\theta^c_{t+1})-\gamma^t\phi(s_t,\theta^c_t)+\gamma^{\tau_3}\phi(s_{\tau_3},\theta^c_{\tau_3})\\
&\quad+\ldots+\sum_{t=\tau_{(2k-1)}}^{\tau_{2K-1}}\gamma^t\phi(s_t,\theta^c_t)-\gamma^{t-1}\phi(s_{t-1},\theta^c_{t-1})+\gamma^{\tau_{2k-1}}\phi(s_{\tau_{2k-1}},\theta^c_{\tau_{2k-1}})+\ldots+\\
&=\sum_{k=1}^{\infty}\sum_{t=\tau_{2k-1}}^{\tau_{2K-1}}\gamma^{t+1}\phi(s_{t+1},\theta^c_{t+1})-\gamma^t\phi(s_t,\theta^c_t)-\sum_{k=1}^{\infty}\gamma^{\tau_{2k-1}}\phi(s_{\tau_{2k-1}},\theta^c_{\tau_{2k-1}})\\
&=\sum_{k=1}^{\infty}\gamma^{\tau_{2k}}\phi(s_{\tau_{2k}},\theta^c_{\tau_{2k}})-\sum_{k=1}^{\infty}\gamma^{\tau_{2k-1}}\phi(s_{\tau_{2k-1}},\theta^c_{\tau_{2k-1}})\\
&=\sum_{k=1}^{\infty}\gamma^{\tau_{2k}}\phi(s_{\tau_{2k}},0)-\sum_{k=1}^{\infty}\gamma^{\tau_{2k-1}}\phi(s_{\tau_{2k-1}},0)=0
\end{aligned}
$$

where we have used the fact that by construction $\theta^c_t\equiv 0$ *whenever* $t=\tau_1,\tau_2,\ldots$ *and by construction* $\phi(s,0)\equiv 0$ *for any* s.

With this we readily deduce that $v^{\boldsymbol{\pi},g}(s)=\mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty}\gamma^t R(s_t,\boldsymbol{a}_t)\right]$ *which is a measure of environment rewards only from which statement (i) can be readily deduced.*

For part (ii) we note first that it is easy to see that $v_c^{\boldsymbol{\pi},g}(s_0,I_0)$ *is bounded above, indeed using the key result in the proof of part (i) and the properties of* c *we have that*

$$
\begin{aligned}
v_c^{\boldsymbol{\pi},g}(s_0,I_0)&=\mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty}\gamma^t\left(R^{\boldsymbol{\theta}}+\sum_{k\geq 1}c(I_t,I_{t-1})\delta^t_{\tau_{2k-1}}+L_n(s_t,\boldsymbol{a}_t)\right)\right] &(15)\\
&=\mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty}\gamma^t\left(R+\sum_{k\geq 1}c(I_t,I_{t-1})\delta^t_{\tau_{2k-1}}+L_n(s_t,\boldsymbol{a}_t)\right)+\sum_{t=0}^{\infty}\gamma^t F^{\boldsymbol{\theta}}I_t\right] &(16)
\end{aligned}
$$

$$\leq \mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty}\gamma^t\left(R+L_n(s_t,\boldsymbol{a}_t)\right)\right] \tag{17}$$

$$\leq \left|\mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty}\gamma^t\left(R+L_n(s_t,\boldsymbol{a}_t)\right)\right]\right| \tag{18}$$

$$\leq \mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty}\gamma^t\left\|R+L_n\right\|\right] \tag{19}$$

$$\leq \sum_{t=0}^{\infty}\gamma^t\left(\|R\|+\|L_n\|\right) \tag{20}$$

$$= \frac{1}{1-\gamma}\left(\|R\|+\|L_n\|\right) \tag{21}$$

using the triangle inequality, the definition of $R^{\boldsymbol{\theta}}$ and the (upper-)boundedness of L and R (Assumption 5). We now note that by the dominated convergence theorem we have that $\forall(s_0,I_0)\in\mathcal{S}\times\{0,1\}$

$$\lim_{n\to\infty} v_c^{\boldsymbol{\pi},g}(s_0,I_0) = \lim_{n\to\infty}\mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty}\gamma^t\left(R^{\boldsymbol{\theta}}+\sum_{k\geq 1}c(I_t,I_{t-1})\delta^t_{\tau_{2k-1}}+L_n(s_t,a_t)\right)\right] \tag{22}$$

$$= \mathbb{E}_{\boldsymbol{\pi},g}\lim_{n\to\infty}\left[\sum_{t=0}^{\infty}\gamma^t\left(R^{\boldsymbol{\theta}}+\sum_{k\geq 1}c(I_t,I_{t-1})\delta^t_{\tau_{2k-1}}+L_n(s_t,a_t)\right)\right] \tag{23}$$

$$= \mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty}\gamma^t\left(R^{\boldsymbol{\theta}}+\sum_{k\geq 1}c(I_t,I_{t-1})\delta^t_{\tau_{2k-1}}\right)\right] \tag{24}$$

$$= \mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty}\gamma^t\left(R+\sum_{k\geq 1}c(I_t,I_{t-1})\delta^t_{\tau_{2k-1}}\right)\right] = c\frac{K}{1-\gamma}+v^{\boldsymbol{\pi}}(s_0), \tag{25}$$

where K is the total number of switch activations or interventions and where again we have used the key result in the proof of (i) and Assumption 6 in the last step, after which we deduce (ii) since $v^{\boldsymbol{\pi}}$ and $v_c^{\boldsymbol{\pi},g}$ differ by only a constant.

Note that by (ii) we heron may consider the quantity for the Generator *expected return:*

$$\hat{v}_c^{\boldsymbol{\pi},g}(s_0,I_0) = \mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty}\gamma^t\left(R+\sum_{k\geq 1}c(I_t,I_{t-1})\delta^t_{\tau_{2k-1}}\right)\right]. \tag{26}$$

PROOF OF THEOREM 1

Proof 4 *Theorem 1 is proved by firstly showing that when the agents jointly maximise the same objective there exists a fixed point equilibrium of the game when all agents use Markov policies and* Generator *uses switching control. The proof then proceeds by showing that the MG $\mathcal{G}$ admits a dual representation as an MG in which all agents $\mathcal{N}\times\{c\}$ jointly maximise the same objective which has MPE that can be computed by solving an MDP. Thereafter, we use both results to prove the existence of a fixed point for the game as a limit point of a sequence generated by successively applying the Bellman operator to a test function.*

Therefore, the scheme of the proof is summarised with the following steps:

I) *Prove that the solution to Markov Team games (that is games in which all agents maximise an identical objective) in which one of the agents uses switching control is the limit point of a sequence of Bellman operators (acting on some test function).*

II) *Prove that for the MG $\mathcal{G}$ that is there exists a function $B^{\boldsymbol{\pi},g}:\mathcal{S}\times\{0,1\}\to\mathbb{R}$ such that*[5] $v_i^{\boldsymbol{\pi},g}(z)-v_i^{\boldsymbol{\pi}',g}(z)=B^{\boldsymbol{\pi},g}(z)-B^{\boldsymbol{\pi}',g}(z),\quad\forall z\equiv(s,I_0)\in\mathcal{S}\times\{0,1\},\forall i\in\mathcal{N}\times\{0\}.$

III) *Prove that the MG, $\mathcal{G}$ has a dual representation as a Markov Team Game which admits a representation as an MDP.*

PROOF OF PART **I**

Our first result proves that the operator T is a contraction operator. First let us recall that the switching time τ_k is defined recursively $\tau_k=\inf\{t>\tau_{k-1}|s_t\in A,\tau_k\in\mathcal{F}_t\}$ where $A=\{s\in\mathcal{S},m\in M|\mathfrak{g}_c(m|s_t)>0\}$. To this end, we show that the following bounds holds:

Lemma 3 *The Bellman operator T is a contraction, that is the following bound holds:*

$$\|T\psi-T\psi'\|\leq\gamma\|\psi-\psi'\|.$$

Proof 5 *Recall we define the Bellman operator T_ψ of $\mathcal{G}$ acting on a function $\Lambda:\mathcal{S}\times\mathbb{N}\to\mathbb{R}$ by*

$$T_\psi\Lambda(s_{\tau_k},I(\tau_k)):=\max\left\{\mathcal{M}^{\boldsymbol{\pi},g}\Lambda(s_{\tau_k},I(\tau_k)),\left[\psi(s_{\tau_k},\boldsymbol{a})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\sum_{s'\in\mathcal{S}}P(s';\boldsymbol{a},s_{\tau_k})\Lambda(s',I(\tau_k))\right]\right\} \tag{27}$$

In what follows and for the remainder of the script, we employ the following shorthands:

$$\mathcal{P}^{\boldsymbol{a}}_{ss'}=:\sum_{s'\in\mathcal{S}}P(s';\boldsymbol{a},s),\quad\mathcal{P}^{\boldsymbol{\pi}}_{ss'}=:\sum_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}|s)\mathcal{P}^{\boldsymbol{a}}_{ss'},\quad\mathcal{R}^{\boldsymbol{\pi}}(z_t):=\sum_{\boldsymbol{a}_t\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_t|s)\hat{R}(z_t,\boldsymbol{a}_t,\theta_t,\theta_{t-1})$$

To prove that T is a contraction, we consider the three cases produced by (27), that is to say we prove the following statements:

i) $\left|\Theta(z_t,\boldsymbol{a},\theta^c_t,\theta^c_{t-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_t}\psi(s',\cdot)-\left(\Theta(z_t,\boldsymbol{a},\theta^c_t,\theta^c_{t-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_t}\psi'(s',\cdot)\right)\right|\leq\gamma\|\psi-\psi'\|$

ii) $\|\mathcal{M}^{\boldsymbol{\pi},g}\psi-\mathcal{M}^{\boldsymbol{\pi},g}\psi'\|\leq\gamma\|\psi-\psi'\|,$ *(and hence $\mathcal{M}$ is a contraction).*

iii) $\left\|\mathcal{M}^{\boldsymbol{\pi},g}\psi-\left[\Theta(\cdot,\boldsymbol{a})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}\psi'\right]\right\|\leq\gamma\|\psi-\psi'\|.$ *where $z_t\equiv(s_t,I_t)\in\mathcal{S}\times\{0,1\}$.*

We begin by proving i).

Indeed, for any $\boldsymbol{a}\in\boldsymbol{\mathcal{A}}$ and $\forall z_t\in\mathcal{S}\times\{0,1\},\forall\theta_t,\theta_{t-1}\in\Theta,\forall s'\in\mathcal{S}$ we have that

$$\begin{aligned}&\left|\Theta(z_t,\boldsymbol{a},\theta^c_t,\theta^c_{t-1})+\gamma\mathcal{P}^{\pi}_{s's_t}\psi(s',\cdot)-\left[\Theta(z_t,\boldsymbol{a},\theta^c_t,\theta^c_{t-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\ \mathcal{P}^{\boldsymbol{a}}_{s's_t}\psi'(s',\cdot)\right]\right|\\&\leq\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\left|\gamma\mathcal{P}^{\boldsymbol{a}}_{s's_t}\psi(s',\cdot)-\gamma\mathcal{P}^{\boldsymbol{a}}_{s's_t}\psi'(s',\cdot)\right|\\&\leq\gamma\|P\psi-P\psi'\|\\&\leq\gamma\|\psi-\psi'\|,\end{aligned}$$

again using the fact that P is non-expansive and Lemma 1.

We now prove ii).

For any $\tau\in\mathcal{F}$, define by $\tau'=\inf\{t>\tau|s_t\in A,\tau\in\mathcal{F}_t\}$. Now using the definition of $\mathcal{M}$ we have that for any $s_\tau\in\mathcal{S}$

$$|(\mathcal{M}^{\boldsymbol{\pi},g}\psi-\mathcal{M}^{\boldsymbol{\pi},g}\psi')(s_\tau,I(\tau))|$$

[5]This property is analogous to the condition in Markov potential games (Macua et al., 2018; Mguni et al., 2021)

$$\leq \max_{\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1}\in\boldsymbol{\mathcal{A}}\times\Theta^2}\Big|\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+c(I_\tau,I_{\tau-1})+\gamma\mathcal{P}^{\boldsymbol{\pi}}_{s's_\tau}\mathcal{P}^{\boldsymbol{a}}\psi(s_\tau,I(\tau'))$$
$$-\big(\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+c(I_\tau,I_{\tau-1})+\gamma\mathcal{P}^{\boldsymbol{\pi}}_{s's_\tau}\mathcal{P}^{\boldsymbol{a}}\psi'(s_\tau,I(\tau'))\big)\Big|$$
$$=\gamma\left|\mathcal{P}^{\boldsymbol{\pi}}_{s's_\tau}\mathcal{P}^{\boldsymbol{a}}\psi(s_\tau,I(\tau'))-\mathcal{P}^{\boldsymbol{\pi}}_{s's_\tau}\mathcal{P}^{\boldsymbol{a}}\psi'(s_\tau,I(\tau'))\right|$$
$$\leq\gamma\left\|P\psi-P\psi'\right\|$$
$$\leq\gamma\left\|\psi-\psi'\right\|,$$

using the fact that P is non-expansive. The result can then be deduced easily by applying max on both sides.

We now prove iii). We split the proof of the statement into two cases:

Case 1:

$$\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau,I(\tau))-\left(\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s',I(\tau))\right)<0. \tag{28}$$

We now observe the following:

$$\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau,I(\tau))-\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s',I(\tau))$$
$$\leq\max\left\{\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\mathcal{P}^{\boldsymbol{\pi}}_{s's_\tau}\mathcal{P}^{\boldsymbol{a}}\psi(s',I(\tau)),\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau,I(\tau))\right\}$$
$$-\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s',I(\tau))$$
$$\leq\Bigg|\max\left\{\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\mathcal{P}^{\boldsymbol{\pi}}_{s's_\tau}\mathcal{P}^{\boldsymbol{a}}\psi(s',I(\tau)),\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau,I(\tau))\right\}$$
$$-\max\left\{\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s',I(\tau)),\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau,I(\tau))\right\}$$
$$+\max\left\{\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s',I(\tau)),\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau,I(\tau))\right\}$$
$$-\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s',I(\tau))\Bigg|$$
$$\leq\Bigg|\max\left\{\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi(s',I(\tau)),\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau,I(\tau))\right\}$$
$$-\max\left\{\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s',I(\tau)),\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau,I(\tau))\right\}\Bigg|$$
$$+\Bigg|\max\left\{\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s',I(\tau)),\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau,I(\tau))\right\}$$
$$-\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s',I(\tau))\Bigg|$$
$$\leq\gamma\max_{a\in\mathcal{A}}\left|\mathcal{P}^{\boldsymbol{\pi}}_{s's_\tau}\mathcal{P}^{\boldsymbol{a}}\psi(s',I(\tau))-\mathcal{P}^{\boldsymbol{\pi}}_{s's_\tau}\mathcal{P}^{\boldsymbol{a}}\psi'(s',I(\tau))\right|$$
$$+\left|\max\left\{0,\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau,I(\tau))-\left(\Theta(z_\tau,\boldsymbol{a}_\tau,\theta^c_\tau,\theta^c_{\tau-1})+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s',I(\tau))\right)\right\}\right|$$
$$\leq\gamma\left\|P\psi-P\psi'\right\|$$
$$\leq\gamma\|\psi-\psi'\|,$$

where we have used the fact that for any scalars a,b,c we have that $|\max\{a,b\}-\max\{b,c\}|\leq|a-c|$ and the non-expansiveness of P.

Case 2:

$$\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau, I(\tau)) - \left(\Theta(z_\tau, \boldsymbol{a}_\tau, \theta^c_\tau, \theta^c_{\tau-1}) + \gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}} \mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s', I(\tau))\right) \geq 0.$$

For this case, first recall that for any $\tau \in \mathcal{F}$, $-c(I_\tau, I_{\tau-1}) > \lambda$ for some $\lambda > 0$.

$$\begin{aligned}
&\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau, I(\tau)) - \left(\Theta(z_\tau, \boldsymbol{a}_\tau, \theta^c_\tau, \theta^c_{\tau-1}) + \gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}} \mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s', I(\tau))\right) \\
&\leq \mathcal{M}^{\boldsymbol{\pi},g}\psi(s_\tau, I(\tau)) - \left(\Theta(z_\tau, \boldsymbol{a}_\tau, \theta^c_\tau, \theta^c_{\tau-1}) + \gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}} \mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s', I(\tau))\right) - c(I_\tau, I_{\tau-1}) \\
&\leq \Theta(z_\tau, \boldsymbol{a}_\tau, \theta^c_\tau, \theta^c_{\tau-1}) + c(I_\tau, I_{\tau-1}) + \gamma\mathcal{P}^{\boldsymbol{\pi}}_{s's_\tau}\mathcal{P}^{\boldsymbol{a}}\psi(s', I(\tau')) \\
&\qquad\qquad - \left(\Theta(z_\tau, \boldsymbol{a}_\tau, \theta^c_\tau, \theta^c_{\tau-1}) + c(I_\tau, I_{\tau-1}) + \gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}} \mathcal{P}^{\boldsymbol{a}}_{s's_\tau}\psi'(s', I(\tau))\right) \\
&\leq \gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}} \left|\mathcal{P}^{\boldsymbol{\pi}}_{s's_\tau}\mathcal{P}^{\boldsymbol{a}}\left(\psi(s', I(\tau')) - \psi'(s', I(\tau))\right)\right| \\
&\leq \gamma\left|\psi(s', I(\tau')) - \psi'(s', I(\tau))\right| \\
&\leq \gamma\left\|\psi - \psi'\right\|,
\end{aligned}$$

again using the fact that P is non-expansive. Hence we have succeeded in showing that for any $\Lambda \in L_2$ we have that

$$\left\|\mathcal{M}^{\boldsymbol{\pi},g}\Lambda - \max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\left[\psi(\cdot, a) + \gamma\mathcal{P}^{\boldsymbol{a}}\Lambda'\right]\right\| \leq \gamma\left\|\Lambda - \Lambda'\right\|. \tag{29}$$

Gathering the results of the three cases gives the desired result.

PROOF OF PART **II**

*To prove Part **II**, we prove the following result:*

Proposition 3 *For any $\boldsymbol{\pi} \in \boldsymbol{\Pi}$ and for any Generator policy g, there exists a function $B^{\boldsymbol{\pi},g} : \mathcal{S} \times \{0,1\} \to \mathbb{R}$ such that*

$$v_i^{\boldsymbol{\pi},g}(z) - v_i^{\boldsymbol{\pi}',g}(z) = B^{\boldsymbol{\pi},g}(z) - B^{\boldsymbol{\pi}',g}(z), \quad \forall z \equiv (s, I_0) \in \mathcal{S} \times \{0,1\} \tag{30}$$

where in particular the function B is given by:

$$B^{\boldsymbol{\pi},g}(s_0, I_0) = \mathbb{E}_{\boldsymbol{\pi},g}\left[\sum_{t=0}^{\infty}\gamma^t\left(R^{\boldsymbol{\theta}} + \sum_{k\geq 1} c(I_t, I_{t-1})\delta^t_{\tau_{2k-1}}\right)\right], \tag{31}$$

for any $(s_0, I_0) \in \mathcal{S} \times \{0,1\}$.

Proof 6 *Note that by the deduction of (ii) in Prop 1, we immediately observe that*

$$\hat{v}_c^{\boldsymbol{\pi},g}(s_0, I_0) = B^{\boldsymbol{\pi},g}(s_0, I_0), \quad \forall (s_0, I_0) \in \mathcal{S} \times \{0,1\}. \tag{32}$$

We therefore immediately deduce that for any two Generator policies g and g' the following expression holds $\forall (s_0, I_0) \in \mathcal{S} \times \{0,1\}$:

$$\hat{v}_c^{\boldsymbol{\pi},g}(s_0, I_0) - \hat{v}_c^{\boldsymbol{\pi},g'}(s_0, I_0) = B^{\boldsymbol{\pi},g}(s_0, I_0) - B^{\boldsymbol{\pi},g'}(s_0, I_0). \tag{33}$$

Our aim now is to show that the following expression holds $\forall (s_0, I_0) \in \mathcal{S} \times \{0,1\}$:

$$v_i^{\boldsymbol{\pi},g}(I_0, s_0) - v_i^{\boldsymbol{\pi}',g}(I_0, s_0) = B^{\boldsymbol{\pi},g}(I_0, s_0) - B^{\boldsymbol{\pi}',g}(I_0, s_0), \quad \forall i \in \mathcal{N}$$

For the finite horizon case, the result is proven by induction on the number of time steps until the end of the game. Unlike the infinite horizon case, for the finite horizon case the value function and policy have an explicit time dependence.

We consider the case of the proposition at time $T-1$ that is we evaluate the value functions at the penultimate time step. In this case, we have that:

$$
\begin{aligned}
&\mathbb{E}_{s_{T-1}\sim d_\theta}\left[B^{\boldsymbol{\pi},g}_{T-1}(I_{T-1},s_{T-1}) - B^{\boldsymbol{\pi}',g}_{T-1}(I_{T-1},s_{T-1})\right] \\
&= \mathbb{E}_{s_{T-1}\sim d_\theta}\Bigg[\sum_{\boldsymbol{a}_{T-1}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-1};s_{T-1})\left[R(s_{T-1},\boldsymbol{a}_{T-1}) + \sum_{k\geq 0}\sum_{j=T-1}^{\infty} c(I_j,I_{j-1})\delta^j_{\tau_k}\right] \\
&\qquad + \gamma\sum_{s_T\in S}\sum_{\boldsymbol{a}_{T-1}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-1};s_{T-1})P(s_T;\boldsymbol{a}_{T-1})B^{\boldsymbol{\pi},g}_{T}(I_T,s_T) \\
&\qquad - \Bigg(\sum_{\boldsymbol{a}'_{T-1}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}'(\boldsymbol{a}'_{T-1};s_{T-1})\left[R(s_{T-1},\boldsymbol{a}'_{T-1}) + \sum_{k\geq 0}\sum_{j=T-1}^{\infty} c(I_j,I_{j-1})\delta^j_{\tau_k}\right] \\
&\qquad + \gamma\sum_{s_T\in S}\sum_{\boldsymbol{a}'_{T-1}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}'(\boldsymbol{a}'_{T-1};s_{T-1})P(s_T;\boldsymbol{a}'_{T-1})B^{\boldsymbol{\pi}',g}_{T}(I_T,s_T)\Bigg)\Bigg] \\
&= \mathbb{E}_{s_{T-1}\sim d_\theta}\Bigg[\sum_{\boldsymbol{a}_{T-1}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-1};s_{T-1})R(s_{T-1},\boldsymbol{a}_{T-1}) \\
&\qquad - \sum_{\boldsymbol{a}'_{T-1}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}'(\boldsymbol{a}'_{T-1};s_{T-1})R(s_{T-1},\boldsymbol{a}'_{T-1}) \\
&\qquad + \gamma\Bigg[\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a}_{T-1}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-1};s_{T-1})P(s_T;\boldsymbol{a}_{T-1})B^{\boldsymbol{\pi},g}_{T}(I_T,s_T) \\
&\qquad - \sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a}'_{T-1}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}'(\boldsymbol{a}'_{T-1};s_{T-1})P(s_T;\boldsymbol{a}'_{T-1})B^{\boldsymbol{\pi}',g}_{T}(I_T,s_T)\Bigg]\Bigg].
\end{aligned}
\tag{34}
$$

We now observe that for any $\boldsymbol{\pi}\in\boldsymbol{\Pi}$ and for any g we have that $B^{\boldsymbol{\pi},g}_T(I_T,s_T) = \mathbb{E}\left[\sum_{\boldsymbol{a}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_T;s_T)\left[R(s_T,\boldsymbol{a}_T)+\sum_{k\geq 0}\sum_{j=T}^{\infty}c(I_j,I_{j-1})\delta^j_{\tau_k}\right]\right]$, moreover we have that for any $\boldsymbol{\pi}\in\boldsymbol{\Pi}$ and for any g

$$
\begin{aligned}
&\mathbb{E}\Bigg[\sum_{\boldsymbol{a}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_T;s_T)\left[R(s_T,\boldsymbol{a}_T)+\sum_{k\geq 0}\sum_{j=T}^{\infty}c(I_j,I_{j-1})\delta^j_{\tau_k}\right] \\
&\qquad - \sum_{\boldsymbol{a}'_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}'(\boldsymbol{a}_T;s_T)\left[R(s_T,\boldsymbol{a}'_T)+\sum_{k\geq 0}\sum_{j=T}^{\infty}c(I_j,I_{j-1})\delta^j_{\tau_k}\right]\Bigg] \\
&=\mathbb{E}\left[\sum_{\boldsymbol{a}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_T;s_T)R(s_T,\boldsymbol{a}_T) - \sum_{\boldsymbol{a}'_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}'(\boldsymbol{a}'_T;s_T)R(s_T,\boldsymbol{a}_T)\right] \\
&\quad + \mathbb{E}\left[\sum_{\boldsymbol{a}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_T;s_T)\sum_{k\geq 0}\sum_{j=T}^{\infty}c(I_j,I_{j-1})\delta^j_{\tau_k} - \sum_{\boldsymbol{a}'_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}'(\boldsymbol{a}'_T;s_T)\sum_{k\geq 0}\sum_{j=T}^{\infty}c(I_j,I_{j-1})\delta^j_{\tau_k}\right]
\end{aligned}
$$

Hence we find that

$$
\begin{aligned}
&\mathbb{E}_{s_{T-1}\sim d_\theta}\left[B^{\boldsymbol{\pi},g}_{T-1}(I_{T-1},s_{T-1}) - B^{\boldsymbol{\pi}',g}_{T-1}(I_{T-1},s_{T-1})\right] \\
&= \mathbb{E}_{s_{T-1}\sim d_\theta}\Bigg[\sum_{\boldsymbol{a}_{T-1}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-1};s_{T-1})R(s_{T-1},\boldsymbol{a}_{T-1}) \\
&\qquad - \sum_{\boldsymbol{a}'_{T-1}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}'(\boldsymbol{a}'_{T-1};s_{T-1})R(s_{T-1},\boldsymbol{a}'_{T-1})
\end{aligned}
\tag{35}
$$

$$+\gamma\Bigg(\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a}_{T-1}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-1};s_{T-1})P(s_T;\boldsymbol{a}_{T-1})\boldsymbol{\pi}(\boldsymbol{a}_T;s_T)R(s_T,\boldsymbol{a}_T) \tag{36}$$

$$-\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a'}_{T-1}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a'}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-1};s_{T-1})P(s_T;\boldsymbol{a'}_{T-1})\boldsymbol{\pi'}(\boldsymbol{a'}_T;s_T)R(s_T,\boldsymbol{a}_T) \tag{37}$$

$$+\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a}_{T-1}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a}_T\in\boldsymbol{\mathcal{A}}}\sum_{k\geq 0}\sum_{j=T}^{\infty}\boldsymbol{\pi}(\boldsymbol{a}_{T-1};s_{T-1})P(s_T;\boldsymbol{a}_{T-1})\boldsymbol{\pi}(\boldsymbol{a}_T;s_T)c(I_j,I_{j-1})\delta^j_{\tau_k} \tag{38}$$

$$-\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a'}_{T-1}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a'}_T\in\boldsymbol{\mathcal{A}}}\sum_{k\geq 0}\sum_{j=T}^{\infty}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-1};s_{T-1})P(s_T;\boldsymbol{a'}_{T-1})\boldsymbol{\pi'}(\boldsymbol{a'}_T;s_T)c(I_j,I_{j-1})\delta^j_{\tau_k}\Bigg)\Bigg] \tag{39}$$

Now

$$\mathbb{E}_{s_{T-1}\sim d_\theta}\Bigg[\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a}_{T-1}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a}_T\in\boldsymbol{\mathcal{A}}}\sum_{k\geq 0}\sum_{j=T}^{\infty}\boldsymbol{\pi}(\boldsymbol{a}_{T-1};s_{T-1})P(s_T;\boldsymbol{a}_{T-1})\boldsymbol{\pi}(\boldsymbol{a}_T;s_T)c(I_j,I_{j-1})\delta^j_{\tau_k}$$

$$-\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a'}_{T-1}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a'}_T\in\boldsymbol{\mathcal{A}}}\sum_{k\geq 0}\sum_{j=T}^{\infty}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-1};s_{T-1})P(s_T;\boldsymbol{a'}_{T-1})\boldsymbol{\pi'}(\boldsymbol{a'}_T;s_T)c(I_j,I_{j-1})\delta^j_{\tau_k}\Bigg]$$

$$=\mathbb{E}_{s_{T-1}\sim d_\theta}\Bigg[\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a}_{T-1}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-1};s_{T-1})P(s_T;\boldsymbol{a}_{T-1})\boldsymbol{\pi}(\boldsymbol{a}_T;s_T)\sum_{k\geq 0}\sum_{j=T}^{\infty}c(I_j,I_{j-1})\delta^j_{\tau_k} \tag{40}$$

$$-\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a'}_{T-1}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a'}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-1};s_{T-1})P(s_T;\boldsymbol{a'}_{T-1})\boldsymbol{\pi'}(\boldsymbol{a'}_T;s_T)\sum_{k\geq 0}\sum_{j=T}^{\infty}c(I_j,I_{j-1})\delta^j_{\tau_k}\Bigg] \tag{41}$$

$$=\mathbb{E}_{s_{T-1}\sim d_\theta}\Bigg[\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a}_{T-1}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-1};s_{T-1})P(s_T;\boldsymbol{a}_{T-1})\boldsymbol{\pi}(\boldsymbol{a}_T;s_T)\sum_{k\geq 0}\sum_{j=T}^{\infty}c(I_j,I_{j-1})\delta^j_{\tau_k} \tag{42}$$

$$-\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a'}_{T-1}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a'}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-1};s_{T-1})P(s_T;\boldsymbol{a'}_{T-1})\boldsymbol{\pi'}(\boldsymbol{a'}_T;s_T)\sum_{k\geq 0}\sum_{j=T}^{\infty}c(I_j,I_{j-1})\delta^j_{\tau_k}\Bigg] \tag{43}$$

$$=K\mathbb{E}_{s_{T-1}\sim d_\theta}\Bigg[\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a}_{T-1}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-1};s_{T-1})P(s_T;\boldsymbol{a}_{T-1})\boldsymbol{\pi}(\boldsymbol{a}_T;s_T) \tag{44}$$

$$-\sum_{s_T\in\mathcal{S}}\sum_{\boldsymbol{a'}_{T-1}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a'}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-1};s_{T-1})P(s_T;\boldsymbol{a'}_{T-1})\boldsymbol{\pi'}(\boldsymbol{a'}_T;s_T)\Bigg] \tag{45}$$

$$=K\left(\sum_{\boldsymbol{a}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_T)-\sum_{\boldsymbol{a'}_T\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi'}(\boldsymbol{a'}_T)\right)=0 \tag{46}$$

using Assumption 6.

Hence we find that

$$\begin{aligned}&\mathbb{E}_{s_{T-1}\sim d_\theta}\left[B^{\boldsymbol{\pi},g}_{T-1}(I_{T-1},s_{T-1})-B^{\boldsymbol{\pi'},g}_{T-1}(I_{T-1},s_{T-1})\right]\\&=\mathbb{E}_{s_{T-1}\sim d_\theta}\Bigg[\sum_{\boldsymbol{a}_{T-1}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-1};s_{T-1})R(s_{T-1},\boldsymbol{a}_{T-1})\\&\qquad-\sum_{\boldsymbol{a'}_{T-1}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-1};s_{T-1})R(s_{T-1},\boldsymbol{a'}_{T-1})\end{aligned} \tag{47}$$

$$+ \gamma \Bigg(\sum_{s_T \in \mathcal{S}} \sum_{\boldsymbol{a}_{T-1} \in \boldsymbol{\mathcal{A}}} \sum_{\boldsymbol{a}_T \boldsymbol{\pi}(\boldsymbol{a}_{T-1}; s_{T-1}) P(s_T; \boldsymbol{a}_{T-1}) \in \boldsymbol{\mathcal{A}}} \boldsymbol{\pi}(\boldsymbol{a}_T; s_T) R(s_T, \boldsymbol{a}_T) \tag{48}$$

$$- \sum_{s_T \in \mathcal{S}} \sum_{\boldsymbol{a'}_{T-1} \in \boldsymbol{\mathcal{A}}} \sum_{\boldsymbol{a'}_T \in \boldsymbol{\mathcal{A}}} \boldsymbol{\pi'}(\boldsymbol{a'}_{T-1}; s_{T-1}) P(s_T; \boldsymbol{a'}_{T-1}) \boldsymbol{\pi'}(\boldsymbol{a'}_T; s_T) R(s_T, \boldsymbol{a}_T) \Bigg) \Bigg] \tag{49}$$

$$= \mathbb{E}_{s_{T-1} \sim d_\theta} \left[v^{\boldsymbol{\pi}, g}_{i,T-1}(s_{T-1}) - v^{\boldsymbol{\pi'}, g}_{i,T-1}(s_{T-1}) \right] \tag{50}$$

Hence, we have succeeded in proving that the expression (30) holds for $T - k$ when $k = 1$.

Our next goal is to prove that the expression holds for any $0 < k \leq T$.

Note that for any $T \geq k > 0$, we can write $B^{\boldsymbol{\pi}, g}_{T-k}$ as

$$B^{\boldsymbol{\pi}, g}_{T-k}(I_0, s_0) = \mathbb{E}_{\boldsymbol{\pi}} \Bigg[R(s_{T-k}, \boldsymbol{a}_{T-k}) + \sum_{k \geq 0} \sum_{j=T-j}^{\infty} c(I_j, I_{j-1}) \delta^j_{\tau_k} \tag{51}$$

$$+ \gamma \sum_{s_{k+1} \in S} P(s'; s_{T-k}, \boldsymbol{a}_{T-k}) B^{\boldsymbol{\pi}, g}_{T-(k+1)}(I_{T-(k+1)}, s_{T-(k+1)}) \Bigg]. \tag{52}$$

Now we consider the case when we evaluate the expression (30) for any $0 < k \leq T$. Our inductive hypothesis is the the expression holds for some $0 < k \leq T$, that is for any $0 < k \leq T$ we have that:

$$\sum_{s_{T-k} \in \mathcal{S}} \sum_{\boldsymbol{a}_{T-(k+1)} \in \boldsymbol{\mathcal{A}}} \boldsymbol{\pi}(\boldsymbol{a}_{T-(k+1)}; s_{T-(k+1)}) P(s_{T-k}; \boldsymbol{a}_{T-(k+1)}) v^{\boldsymbol{\pi}, g}_{i,T-k}(I_{T-k}, s_{T-k})$$

$$- \sum_{s_{T-k} \in \mathcal{S}} \sum_{\boldsymbol{a'}_{T-(k+1)} \in \boldsymbol{\mathcal{A}}} \boldsymbol{\pi'}(\boldsymbol{a'}_{T-(k+1)}; s_{T-(k+1)}) P(s_{T-k}; \boldsymbol{a'}_{T-(k+1)}) v^{\boldsymbol{\pi'}, g}_{i,T-k}(I_{T-k}, s_{T-k}) \tag{53}$$

$$= \sum_{s_{T-k} \in \mathcal{S}} \sum_{\boldsymbol{a}_{T-(k+1)} \in \boldsymbol{\mathcal{A}}} \boldsymbol{\pi}(\boldsymbol{a}_{T-(k+1)}; s_{T-(k+1)}) P(s_{T-k}; \boldsymbol{a}_{T-(k+1)}) B^{\boldsymbol{\pi}, g}_{T-k}(I_{T-k}, s_{T-k}) \tag{54}$$

$$- \sum_{s_{T-k} \in \mathcal{S}} \sum_{\boldsymbol{a'}_{T-(k+1)} \in \boldsymbol{\mathcal{A}}} \boldsymbol{\pi'}(\boldsymbol{a'}_{T-(k+1)}; s_{T-(k+1)}) P(s_{T-k}; \boldsymbol{a'}_{T-(k+1)}) B^{\boldsymbol{\pi'}, g}_{T-k}(I_{T-k}, s_{T-k}). \tag{55}$$

It remains to show that the expression holds for $k + 1$ time steps prior to the end of the horizon. The result can be obtained using the dynamic programming principle and the base case ($k = 1$) result, indeed we have that

$$\mathbb{E}_{s_{T-(k+1)} \sim d_\theta} \left[B^{\boldsymbol{\pi}, g}_{T-(k+1)}(I_{T-(k+1)}, s_{T-(k+1)}) - B^{\boldsymbol{\pi'}, g}_{T-(k+1)}(I_{T-(k+1)}, s_{T-(k+1)}) \right]$$

$$= \mathbb{E}_{s_{T-(k+1)} \sim d_\theta} \Bigg[\sum_{\boldsymbol{a}_{T-(k+1)} \in \boldsymbol{\mathcal{A}}} \boldsymbol{\pi}(\boldsymbol{a}_{T-(k+1)}; s_{T-(k+1)}) \left[R(s_{T-(k+1)}, \boldsymbol{a}_{T-(k+1)}) + \sum_{k \geq 0} \sum_{j=T-1}^{\infty} c(I_j, I_{j-1}) \delta^j_{\tau_k} \right]$$

$$+ \gamma \sum_{s_{T-k} \in S} \sum_{\boldsymbol{a}_{T-(k+1)} \in \boldsymbol{\mathcal{A}}} \boldsymbol{\pi}(\boldsymbol{a}_{T-(k+1)}; s_{T-(k+1)}) P(s_{T-k}; \boldsymbol{a}_{T-(k+1)}) B^{\boldsymbol{\pi}, g}_{T-k}(I_{T-k}, s_{T-k})$$

$$- \Bigg(\sum_{\boldsymbol{a'}_{T-(k+1)} \in \boldsymbol{\mathcal{A}}} \boldsymbol{\pi'}(\boldsymbol{a'}_{T-(k+1)}; s_{T-(k+1)}) \Bigg[R(s_{T-(k+1)}, \boldsymbol{a'}_{T-(k+1)}) + \sum_{k \geq 0} \sum_{j=T-1}^{\infty} c(I_j, I_{j-1}) \delta^j_{\tau_k} \Bigg]$$

$$+ \gamma \sum_{s_{T-k} \in S} \sum_{\boldsymbol{a'}_{T-(k+1)} \in \boldsymbol{\mathcal{A}}} \boldsymbol{\pi'}(\boldsymbol{a'}_{T-(k+1)}; s_{T-(k+1)}) P(s_{T-k}; \boldsymbol{a'}_{T-(k+1)}) B^{\boldsymbol{\pi'}, g}_{T-k}(I_{T-k}, s_{T-k}) \Bigg] \Bigg)$$

$$= \mathbb{E}_{s_{T-(k+1)}\sim d_\theta}\Bigg[\sum_{\boldsymbol{a}_{T-(k+1)}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-(k+1)};s_{T-(k+1)})R(s_{T-(k+1)},\boldsymbol{a}_{T-(k+1)}) - \sum_{\boldsymbol{a'}_{T-(k+1)}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-(k+1)};s_{T-(k+1)})R(s_{T-(k+1)},\boldsymbol{a'}_{T-(k+1)}) + \gamma\Bigg[\sum_{s_{T-k}\in\mathcal{S}}\sum_{\boldsymbol{a}_{T-(k+1)}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-(k+1)};s_{T-(k+1)})P(s_{T-k};\boldsymbol{a}_{T-(k+1)})B^{\boldsymbol{\pi},g}_{T-k}(I_{T-k},s_{T-k}) - \sum_{s_{T-k}\in\mathcal{S}}\sum_{\boldsymbol{a'}_{T-(k+1)}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-(k+1)};s_{T-(k+1)})P(s_{T-k};\boldsymbol{a'}_{T-(k+1)})B^{\boldsymbol{\pi'},g}_{T-k}(I_{T-k},s_{T-k})\Bigg]\Bigg]. \tag{56}$$

$$= \mathbb{E}_{s_{T-(k+1)}\sim d_\theta}\Bigg[\sum_{\boldsymbol{a}_{T-(k+1)}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-(k+1)};s_{T-(k+1)})R(s_{T-(k+1)},\boldsymbol{a}_{T-(k+1)}) - \sum_{\boldsymbol{a'}_{T-(k+1)}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-(k+1)};s_{T-(k+1)})R(s_{T-(k+1)},\boldsymbol{a'}_{T-(k+1)}) + \gamma\Bigg[\sum_{s_{T-k}\in\mathcal{S}}\sum_{\boldsymbol{a}_{T-(k+1)}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-(k+1)};s_{T-(k+1)})P(s_{T-k};\boldsymbol{a}_{T-(k+1)})v^{\boldsymbol{\pi},g}_{i,T-k}(I_{T-k},s_{T-k}) - \sum_{s_{T-k}\in\mathcal{S}}\sum_{\boldsymbol{a'}_{T-(k+1)}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-(k+1)};s_{T-(k+1)})P(s_{T-k};\boldsymbol{a'}_{T-(k+1)})v^{\boldsymbol{\pi'},g}_{i,T-k}(I_{T-k},s_{T-k})\Bigg]\Bigg].$$

$$= \mathbb{E}_{s_{T-(k+1)}\sim d_\theta}\left[v^{\boldsymbol{\pi},g}_{i,T-(k+1)}(I_{T-(k+1)},s_{T-(k+1)}) - v^{\boldsymbol{\pi'},g}_{T-(k+1)}(I_{T-(k+1)},s_{i,T-(k+1)})\right] \tag{57}$$

using the inductive hypothesis and where we have used the fact that

$$\mathbb{E}_{s_{T-(k+1)}\sim d_\theta}\Bigg[\sum_{s_{T-k}\in\mathcal{S}}\sum_{\boldsymbol{a}_{T-(k+1)}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a}_{T-k}\in\boldsymbol{\mathcal{A}}}\sum_{k\geq 0}\sum_{j=T}^{\infty}\boldsymbol{\pi}(\boldsymbol{a}_{T-(k+1)};s_{T-(k+1)})P(s_{T-k};\boldsymbol{a}_{T-(k+1)})\boldsymbol{\pi}(\boldsymbol{a}_{T-k};s_{T-k})c(I_j,I_{j-1})\delta^j_{\tau_k} - \sum_{s_{T-k}\in\mathcal{S}}\sum_{\boldsymbol{a'}_{T-(k+1)}\in\boldsymbol{\mathcal{A}}}\sum_{\boldsymbol{a'}_{T-k}\in\boldsymbol{\mathcal{A}}}\sum_{k\geq 0}\sum_{j=T}^{\infty}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-(k+1)};s_{T-(k+1)})P(s_{T-k};\boldsymbol{a'}_{T-(k+1)})\boldsymbol{\pi'}(\boldsymbol{a'}_{T-k};s_{T-k})c(I_j,I_{j-1})\delta^j_{\tau_k}\Bigg]$$

$$= K\left(\sum_{\boldsymbol{a}_{T-k}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi}(\boldsymbol{a}_{T-k}) - \sum_{\boldsymbol{a'}_{T-k}\in\boldsymbol{\mathcal{A}}}\boldsymbol{\pi'}(\boldsymbol{a'}_{T-k})\right) = 0 \tag{58}$$

via similar reasoning as before and after which which we deduce the result in the finite case.

For the infinite horizon case, we must prove that there exists a measurable function $B : \boldsymbol{\Pi}\times\mathcal{S}\to\mathbb{R}$ such that the following holds for any $i\in\mathcal{N}$ and $\forall\pi_i,\pi'_i\in\Pi_i,\forall\pi_{-i}\in\Pi_{-i}$ and $\forall s\in\mathcal{S}$:

$$\mathbb{E}_{s\sim P}\left[\left(v^{\boldsymbol{\pi},g}_i - v^{\boldsymbol{\pi'},g}_i\right)(z)\right] = \mathbb{E}_{s\sim P}\left[\left(B^{\boldsymbol{\pi},g} - B^{\boldsymbol{\pi'},g}\right)(z)\right]. \tag{59}$$

The result is proven by contradiction.

To this end, let us firstly assume $\exists c\neq 0$ such that

$$\mathbb{E}_{s\sim P}\left[\left(v^{\boldsymbol{\pi},g}_i - v^{\boldsymbol{\pi'},g}_i\right)(z)\right] - \mathbb{E}_{s\sim P}\left[\left(B^{\boldsymbol{\pi},g}_i - B^{\boldsymbol{\pi'},g}_i\right)(z)\right] = c.$$

Let us now define the following quantities for any $s\in\mathcal{S}$ and for each $\pi_i\in\Pi_i$ and $\pi_{-i}\in\Pi_{-i}$ and $\forall i\in\mathcal{N}$:

$$v^{\boldsymbol{\pi},g}_{i,T'}(z) := \sum_{t=0}^{T'}\mu(s_0)\pi_i(a^i_0,s_0)\pi_{-i}(a^{-i}_0,s_0)\prod_{j=0}^{t-1}\sum_{s_{j+1}\in\mathcal{S}}\gamma^t P(s_{j+1};s_j,a_j)\pi_i(a^i_j|s_j)\pi_{-i}(a^{-i}_j|s_j)r_i(z_j,\boldsymbol{a}_j),$$

and

$$B^{\boldsymbol{\pi},g}_{T'}(z) := \sum_{t=0}^{T'} \mu(s_0)\pi_i(a^i_0, s_0)\pi_{-i}(a^{-i}_0, s_0) \prod_{j=0}^{t-1} \sum_{s_{j+1}\in\mathcal{S}} P(s_{j+1}; s_j, a_j) \cdot \pi_i(a^i_j|s_j)\pi_{-i}(a^{-i}_j|s_j)\Theta(z_j, \boldsymbol{a}_j),$$

so that the quantity $v^{\boldsymbol{\pi}}_{i,T'}(s)$ measures the expected cumulative return until the point $T' < \infty$.

Hence, we deduce that

$$\begin{aligned} v^{\boldsymbol{\pi}}_i(z) &\equiv v^{\boldsymbol{\pi}}_{i,\infty}(z) \\ &= v^{\boldsymbol{\pi}}_{i,T'}(z) + \gamma^{T'}\mu(s_0)\pi_i(a^i_0, s_0)\pi_{-i}(a^{-i}_0, s_0) \prod_{j=0}^{T'-1} \sum_{s_{j+1}\in\mathcal{S}} \gamma^t P(s_{j+1}; s_j, a_j)\pi_i(a^i_j|s_j)\pi_{-i}(a^{-i}_j|s_j) v^{\boldsymbol{\pi}}_i(s_{T'}). \end{aligned}$$

Next we observe that:

$$\begin{aligned} c &= \mathbb{E}_{s\sim P}\left[\left(v^{\boldsymbol{\pi},g}_i - v^{\boldsymbol{\pi}',g}_i\right)(z)\right] - \mathbb{E}_{s\sim P}\left[\left(B^{\boldsymbol{\pi},g} - B^{\boldsymbol{\pi}',g}\right)(z)\right] \\ &= \mathbb{E}_{s\sim P}\left[\left(v^{\boldsymbol{\pi},g}_{i,T'} - v^{\boldsymbol{\pi}',g}_{i,T'}\right)(z)\right] - \mathbb{E}_{s\sim P}\left[\left(B^{\boldsymbol{\pi},g}_{T'} - B^{\boldsymbol{\pi}',g}_{T'}\right)(s)\right] \\ &+\gamma^{T'}\mathbb{E}_{s_{T'}\sim P}\Bigg[\mu(s_0)\pi_i(a^i_0, s_0)\pi_{-i}(a^{-i}_0, s_0) \prod_{j=0}^{T'-1} \sum_{s_{j+1}\in\mathcal{S}} P(s_{j+1}; s_j, a_j)\pi_i(a^i_j|s_j)\pi_{-i}(a^{-i}_j|s_j)\left(v^{\boldsymbol{\pi},g}_i(z_{T'}) - B^{\boldsymbol{\pi},g}(z_{T'})\right) \\ &- \mu(s_0)\pi'_i(a'^i_0, s_0)\pi_{-i}(a^{-i}_0, s_0) \prod_{j=0}^{T'-1} \sum_{s_{j+1}\in\mathcal{S}} P(s_{j+1}; s_j, a'_j)\pi'_i(a'^i_j|s_j)\pi_{-i}(a^{-i}_j|s_j)\left(v^{\boldsymbol{\pi}',g}_i(z_{T'}) - B^{\boldsymbol{\pi}',g}(z_{T'})\right)\Bigg]. \end{aligned}$$

Considering the last expectation and its coefficient and denoting the product by κ, using the fact that by the Cauchy-Schwarz inequality we have $\|AX - BY\| \leq \|A\|\|X\| + \|B\|\|Y\|$, moreover whenever A, B are non-expansive we have that $\|AX - BY\| \leq \|X\| + \|Y\|$, hence we observe the following $\kappa \leq \|\kappa\| \leq 2\gamma^{T'}(\|v_i\| + \|B\|)$. Since we can choose T' freely and $\gamma \in]0, 1[$, we can choose T' to be sufficiently large so that $\gamma^{T'}(\|v_i\| + \|B\|) < \frac{1}{4}|c|$. This then implies that

$$\left|\mathbb{E}_{s\sim P}\left[\left(v^{\boldsymbol{\pi},g}_{i,T'} - v^{\boldsymbol{\pi}'}_{i,T'}\right)(z) - \left(B^{\boldsymbol{\pi},g}_{T'} - B^{\boldsymbol{\pi}'}_{T'}\right)(z)\right]\right| > \frac{1}{2}c,$$

which is a contradiction since we have proven that for any finite T' it is the case that

$$\mathbb{E}_{s\sim P}\left[\left(v^{\boldsymbol{\pi},g}_{i,T'} - v^{\boldsymbol{\pi}'}_{i,T'}\right)(z) - \left(B^{\boldsymbol{\pi},g}_{T'} - B^{\boldsymbol{\pi}'}_{T'}\right)(z)\right] = 0,$$

and hence we deduce the result in the infinite horizon case.

PROOF OF PART **III**

*To prove Part **III**, we firstly define precisely the notion of a stable point of the MG, $\mathcal{G}$:*

Definition 3 *A policy profile $\boldsymbol{\sigma}^\star = (g^\star, \pi^\star_i, \pi^\star_{-i}) \in \boldsymbol{\Pi}$ is a Markov perfect equilibrium (MPE) in Markov strategies if the following condition holds for any $i \in \mathcal{N} \times \{0\}$:*

$$v^{(g^\star,\pi^\star_i,\pi^\star_{-i})}_i(z) \geq v^{g^\star,(\pi'_i,\pi^\star_{-i})}_i(z),\ \forall z \equiv (s_0, I_0) \in \mathcal{S} \times \{0, 1\},\ \forall \pi'_i \in \Pi_i. \quad (60)$$

$$v^{(g^\star,\pi^\star_i,\pi^\star_{-i})}_c(z) \geq v^{g',(\pi_i,\pi^\star_{-i})}_c(z),\ \forall z \equiv (s_0, I_0) \in \mathcal{S} \times \{0, 1\},\ \forall g'. \quad (61)$$

The condition characterises strategic configurations which are stable points of the MG, $\mathcal{G}$. In particular, an MPE is achieved when at any state no agent can improve their expected cumulative rewards by unilaterally deviating from their current policy. We denote by $NE\{\mathcal{G}\}$ the set of MPE strategies for the MG, $\mathcal{G}$.

Next we prove that the set of maxima of the function B are the MPE of the MG $\mathcal{G}$:

Proposition 4 *The following implication holds:*

$$\boldsymbol{\sigma} \in \underset{g',\boldsymbol{\pi}' \in \boldsymbol{\Pi}}{\arg\sup}\, B^{g',\boldsymbol{\pi}'}(s) \implies \boldsymbol{\sigma} \in NE\{\mathcal{G}\}. \tag{62}$$

where B is the function in Prop. 3.

Prop. 4 indicates that the game has an equivalent representation in which all agents maximise the same function and thus play a team game.

Proof 1 *We do the proof by contradiction. Let $\boldsymbol{\sigma} = (\pi_1, \ldots, \pi_N, g) \in \underset{\boldsymbol{\pi}' \in \boldsymbol{\Pi}, g'}{\arg\sup}\, B^{\boldsymbol{\pi}',g'}(s)$ for any $s \in \mathcal{S}$. Let us now therefore assume that $\boldsymbol{\sigma} \notin NE\{\mathcal{G}\}$, hence there exists some other policy profile $\tilde{\boldsymbol{\sigma}} = (\pi_1, \ldots, \tilde{\pi}_i, \ldots, \pi_N, g)$ which contains at least one profitable deviation by one of the agents $i \in \mathcal{N} \times \{0, \}$. For now let us consider the case in which the profitable deviation is for a agent $i \in \mathcal{N}$ so that $\pi'_i \neq \pi_i$ for $i \in \mathcal{N}$ i.e. $v_i^{(\pi'_i,\pi_{-i}),g}(s) > v_i^{(\pi_i,\pi_{-i}),g}(s)$ (using the preservation of signs of integration). Prop. 3 however implies that $B^{(\pi'_i,\pi_{-i}),g}(s) - B^{(\pi_i,\pi_{-i}),g}(s) > 0$ which is a contradiction since $\boldsymbol{\sigma} = (\pi_i, \pi_{-i}, g)$ is a maximum of B. The proof can be straightforwardly adapted to cover the case in which the deviating agent is Generator after which we deduce the desired result.*

The last result completes the proof of Theorem 1.

Proof of Proposition 2

Proof 7 (Proof of Prop. 2) *The proof is given by establishing a contradiction. Therefore suppose that $\mathcal{M}^{\boldsymbol{\pi},g}\psi(s_{\tau_k}, I(\tau_k)) \leq \psi(s_{\tau_k}, I(\tau_k))$ and suppose that the switching time $\tau'_1 > \tau_1$ is an optimal switching time. Construct the Generator g' and $\tilde{g}$ policy switching times by $(\tau'_0, \tau'_1, \ldots,)$ and g'^2 policy by $(\tau'_0, \tau_1, \ldots)$ respectively. Define by $l = \inf\{t > 0; \mathcal{M}^{\boldsymbol{\pi},g}\psi(s_t, I_0) = \psi(s_t, I_0)\}$ and $m = \sup\{t; t < \tau'_1\}$. By construction we have that*

$$\begin{aligned}
& v_c^{\boldsymbol{\pi},g'}(s, I_0) \\
&= \mathbb{E}\left[R(s_0, \boldsymbol{a}_0) + \mathbb{E}\left[\ldots + \gamma^{l-1}\mathbb{E}\left[R(s_{\tau_1-1}, \boldsymbol{a}_{\tau_1-1}) + \ldots + \gamma^{m-l-1}\mathbb{E}\left[R(s_{\tau'_1-1}, \boldsymbol{a}_{\tau'_1-1}) + \gamma\mathcal{M}^{\pi^1,\pi'^2} v_c^{\boldsymbol{\pi},g'}(s', I(\tau'_1))\right]\right]\right]\right] \\
&< \mathbb{E}\left[R(s_0, \boldsymbol{a}_0) + \mathbb{E}\left[\ldots + \gamma^{l-1}\mathbb{E}\left[R(s_{\tau_1-1}, \boldsymbol{a}_{\tau_1-1}) + \gamma\mathcal{M}^{\boldsymbol{\pi},\tilde{g}} v_c^{\boldsymbol{\pi},g'}(s_{\tau_1}, I(\tau_1))\right]\right]\right]
\end{aligned}$$

We now use the following observation $\mathbb{E}\left[R(s_{\tau_1-1}, \boldsymbol{a}_{\tau_1-1}) + \gamma\mathcal{M}^{\boldsymbol{\pi},\tilde{g}} v_c^{\boldsymbol{\pi},g'}(s_{\tau_1}, I(\tau_1))\right]$

$$\leq \max\left\{\mathcal{M}^{\boldsymbol{\pi},\tilde{g}} v_c^{\boldsymbol{\pi},g'}(s_{\tau_1}, I(\tau_1)), \max_{a_{\tau_1} \in \mathcal{A}} \left[R(s_{\tau_k}, \boldsymbol{a}_{\tau_k}) + \gamma \textstyle\sum_{s' \in \mathcal{S}} P(s'; \boldsymbol{a}_{\tau_1}, s_{\tau_1}) v_c^{\boldsymbol{\pi},g}(s', I(\tau_1))\right]\right\}.$$

Using this we deduce that

$$\begin{aligned}
& v_2^{\boldsymbol{\pi},g'}(s, I_0) \leq \mathbb{E}\Bigg[R(s_0, \boldsymbol{a}_0) + \mathbb{E}\Bigg[\ldots \\
&+ \gamma^{l-1}\mathbb{E}\left[R(s_{\tau_1-1}, \boldsymbol{a}_{\tau_1-1}) + \gamma \max\left\{\mathcal{M}^{\boldsymbol{\pi},\tilde{g}} v_c^{\boldsymbol{\pi},g'}(s_{\tau_1}, I(\tau_1)), \max_{a_{\tau_1} \in \mathcal{A}} \left[R(s_{\tau_k}, \boldsymbol{a}_{\tau_k}) + \gamma \sum_{s' \in \mathcal{S}} P(s'; \boldsymbol{a}_{\tau_1}, s_{\tau_1}) v_c^{\boldsymbol{\pi},g}(s', I(\tau_1))\right]\right\}\right]\Bigg]\Bigg] \\
&= \mathbb{E}\left[R(s_0, \boldsymbol{a}_0) + \mathbb{E}\left[\ldots + \gamma^{l-1}\mathbb{E}\left[R(s_{\tau_1-1}, \boldsymbol{a}_{\tau_1-1}) + \gamma\left[T v_c^{\boldsymbol{\pi},\tilde{g}}\right](s_{\tau_1}, I(\tau_1))\right]\right]\right] = v_c^{\boldsymbol{\pi},\tilde{g}}(s, I_0))
\end{aligned}$$

where the first inequality is true by assumption on $\mathcal{M}$. This is a contradiction since g' is an optimal policy for Generator. Using analogous reasoning, we deduce the same result for $\tau'_k < \tau_k$ after which deduce the result. Moreover, by invoking the same reasoning, we can conclude that it must be the case that $(\tau_0, \tau_1, \ldots, \tau_{k-1}, \tau_k, \tau_{k+1}, \ldots,)$ are the optimal switching times.

Proof of Theorem 2

Proof 8 (Proof of Theorem 2) *The proof which is done by contradiction follows from the definition of v_c. Denote by $v_i^{\boldsymbol{\pi},g\equiv\mathbf{0}}$ value function an agent $i \in \mathcal{N}$ excluding Generator and its intrinsic-reward* *function. Indeed, let $(\hat{\boldsymbol{\pi}}, \hat{g})$ be the policy profile induced by the Nash equilibrium policy profile and assume that the intrinsic-reward F leads to a decrease in payoff for agent i. Then by construction $v^{\boldsymbol{\pi},g}(s) < v^{\boldsymbol{\pi},g\equiv\mathbf{0}}(s)$ which is a contradiction since $(\hat{\boldsymbol{\pi}}, \hat{g})$ is an MPE profile.*

PROOF OF THEOREM 3

To prove the theorem, we make use of the following result:

Theorem 4 (Theorem 1, pg 4 in (Jaakkola et al., 1994)) *Let $\Xi_t(s)$ be a random process that takes values in $\mathbb{R}^n$ and given by the following:*

$$\Xi_{t+1}(s) = (1 - \alpha_t(s))\,\Xi_t(s)\alpha_t(s)L_t(s), \tag{63}$$

then $\Xi_t(s)$ converges to 0 with probability 1 under the following conditions:

i) $0 \leq \alpha_t \leq 1, \sum_t \alpha_t = \infty$ *and* $\sum_t \alpha_t < \infty$

ii) $\|\mathbb{E}[L_t|\mathcal{F}_t]\| \leq \gamma\|\Xi_t\|$, *with* $\gamma < 1$*;*

iii) $\mathrm{Var}\,[L_t|\mathcal{F}_t] \leq c(1 + \|\Xi_t\|^2)$ *for some* $c > 0$.

Proof 9 *To prove the result, we show (i) - (iii) hold. Condition (i) holds by choice of learning rate. It therefore remains to prove (ii) - (iii). We first prove (ii). For this, we consider our variant of the Q-learning update rule:*

$$Q_{t+1}(s_t, I_t, \boldsymbol{a}_t) = Q_t(s_t, I_t, \boldsymbol{a}_t) + \alpha_t(s_t, I_t, \boldsymbol{a}_t)\left[\max\left\{\mathcal{M}^{\boldsymbol{\pi},g}Q(s_{\tau_k}, I_{\tau_k}, \boldsymbol{a}), \phi(s_{\tau_k}, \boldsymbol{a}) + \gamma\max_{a'\in\mathcal{A}} Q(s', I_{\tau_k}, \boldsymbol{a'})\right\} - Q_t(s_t, I_t, \boldsymbol{a}_t)\right].$$

After subtracting $Q^\star(s_t, I_t, \boldsymbol{a}_t)$ from both sides and some manipulation we obtain that:

$$\begin{aligned}&\Xi_{t+1}(s_t, I_t, \boldsymbol{a}_t)\\ &= (1 - \alpha_t(s_t, I_t, \boldsymbol{a}_t))\Xi_t(s_t, I_t, \boldsymbol{a}_t)\\ &\quad + \alpha_t(s_t, I_t, \boldsymbol{a}_t))\left[\max\left\{\mathcal{M}^{\boldsymbol{\pi},g}Q(s_{\tau_k}, I_{\tau_k}, \boldsymbol{a}), \phi(s_{\tau_k}, \boldsymbol{a}) + \gamma\max_{a'\in\mathcal{A}} Q(s', I_{\tau_k}, \boldsymbol{a'})\right\} - Q^\star(s_t, I_t, \boldsymbol{a}_t)\right],\end{aligned}$$

where $\Xi_t(s_t, I_t, \boldsymbol{a}_t) := Q_t(s_t, I_t, \boldsymbol{a}_t) - Q^\star(s_t, I_t, \boldsymbol{a}_t)$.

Let us now define by

$$L_t(s_{\tau_k}, I_{\tau_k}, \boldsymbol{a}) := \max\left\{\mathcal{M}^{\boldsymbol{\pi},g}Q(s_{\tau_k}, I_{\tau_k}, \boldsymbol{a}), \phi(s_{\tau_k}, \boldsymbol{a}) + \gamma\max_{a'\in\mathcal{A}} Q(s', I_{\tau_k}, \boldsymbol{a'})\right\} - Q^\star(s_t, I_t, a).$$

Then

$$\Xi_{t+1}(s_t, I_t, \boldsymbol{a}_t) = (1 - \alpha_t(s_t, I_t, \boldsymbol{a}_t))\Xi_t(s_t, I_t, \boldsymbol{a}_t) + \alpha_t(s_t, I_t, \boldsymbol{a}_t))\left[L_t(s_{\tau_k}, a)\right]. \tag{64}$$

We now observe that

$$\begin{aligned}\mathbb{E}\left[L_t(s_{\tau_k}, I_{\tau_k}, \boldsymbol{a})|\mathcal{F}_t\right] &= \sum_{s'\in\mathcal{S}} P(s'; a, s_{\tau_k})\max\left\{\mathcal{M}^{\boldsymbol{\pi},g}Q(s_{\tau_k}, I_{\tau_k}, \boldsymbol{a}), \phi(s_{\tau_k}, \boldsymbol{a}) + \gamma\max_{a'\in\mathcal{A}} Q(s', I_{\tau_k}, \boldsymbol{a'})\right\} - Q^\star(s_{\tau_k}, a)\\ &= T_\phi Q_t(s, I_{\tau_k}, \boldsymbol{a}) - Q^\star(s, I_{\tau_k}, \boldsymbol{a}).\end{aligned} \tag{65}$$

Now, using the fixed point property that implies $Q^\star = T_\phi Q^\star$, we find that

$$\begin{aligned}\mathbb{E}\left[L_t(s_{\tau_k}, I_{\tau_k}, \boldsymbol{a})|\mathcal{F}_t\right] &= T_\phi Q_t(s, I_{\tau_k}, \boldsymbol{a}) - T_\phi Q^\star(s, I_{\tau_k}, \boldsymbol{a})\\ &\leq \|T_\phi Q_t - T_\phi Q^\star\|\\ &\leq \gamma\left\|Q_t - Q^\star\right\|_\infty = \gamma\left\|\Xi_t\right\|_\infty.\end{aligned} \tag{66}$$

using the contraction property of T established in Lemma 3. This proves (ii).

We now prove iii), that is

$$\mathrm{Var}\left[L_t|\mathcal{F}_t\right] \leq c(1 + \|\Xi_t\|^2). \tag{67}$$

Now by (65) we have that

$$\begin{aligned}
\mathrm{Var}\left[L_t|\mathcal{F}_t\right] &= \mathrm{Var}\left[\max\left\{\mathcal{M}^{\boldsymbol{\pi},g}Q(s_{\tau_k},I_{\tau_k},\boldsymbol{a}),\phi(s_{\tau_k},\boldsymbol{a})+\gamma\max_{a'\in\mathcal{A}}Q(s',I_{\tau_k},\boldsymbol{a'})\right\}-Q^\star(s_t,I_t,a)\right]\\
&=\mathbb{E}\Bigg[\Bigg(\max\left\{\mathcal{M}^{\boldsymbol{\pi},g}Q(s_{\tau_k},I_{\tau_k},\boldsymbol{a}),\phi(s_{\tau_k},\boldsymbol{a})+\gamma\max_{a'\in\mathcal{A}}Q(s',I_{\tau_k},\boldsymbol{a'})\right\}\\
&\qquad -Q^\star(s_t,I_t,a)-(T_\Phi Q_t(s,I_{\tau_k},\boldsymbol{a})-Q^\star(s,I_{\tau_k},\boldsymbol{a}))\Bigg)^2\Bigg]\\
&=\mathbb{E}\left[\left(\max\left\{\mathcal{M}^{\boldsymbol{\pi},g}Q(s_{\tau_k},I_{\tau_k},\boldsymbol{a}),\phi(s_{\tau_k},\boldsymbol{a})+\gamma\max_{a'\in\mathcal{A}}Q(s',I_{\tau_k},\boldsymbol{a'})\right\}-T_\Phi Q_t(s,I_{\tau_k},\boldsymbol{a})\right)^2\right]\\
&=\mathrm{Var}\left[\max\left\{\mathcal{M}^{\boldsymbol{\pi},g}Q(s_{\tau_k},I_{\tau_k},\boldsymbol{a}),\phi(s_{\tau_k},\boldsymbol{a})+\gamma\max_{a'\in\mathcal{A}}Q(s',I_{\tau_k},\boldsymbol{a'})\right\}-T_\Phi Q_t(s,I_{\tau_k},\boldsymbol{a}))^2\right]\\
&\leq c(1+\|\Xi_t\|^2),
\end{aligned}$$

for some $c > 0$ where the last line follows due to the boundedness of Q (which follows from Assumptions 2 and 4). This concludes the proof of the Theorem.

Proof of Convergence with Function Approximation

First let us recall the statement of the theorem:

Theorem 3 *LIGS converges to a limit point $r^\star$ which is the unique solution to the equation:*

$$\Pi\mathfrak{F}(\Phi r^\star)=\Phi r^\star, \qquad a.e. \tag{68}$$

where we recall that for any test function $\Lambda\in\mathcal{V}$, the operator $\mathfrak{F}$ is defined by $\mathfrak{F}\Lambda:=\Theta+\gamma P\max\{\mathcal{M}\Lambda,\Lambda\}$.

Moreover, $r^\star$ satisfies the following:

$$\|\Phi r^\star-Q^\star\|\leq c\,\|\Pi Q^\star-Q^\star\|\,. \tag{69}$$

The theorem is proven using a set of results that we now establish. To this end, we first wish to prove the following bound:

Lemma 4 *For any $Q\in\mathcal{V}$ we have that*

$$\|\mathfrak{F}Q-Q'\|\leq\gamma\,\|Q-Q'\|\,, \tag{70}$$

so that the operator $\mathfrak{F}$ is a contraction.

Proof 10 *Recall, for any test function ψ , a projection operator Π acting Λ is defined by the following*

$$\Pi\Lambda:=\underset{\bar{\Lambda}\in\{\Phi r|r\in\mathbb{R}^p\}}{\arg\min}\left\|\bar{\Lambda}-\Lambda\right\|.$$

Now, we first note that in the proof of Lemma 3, we deduced that for any $\Lambda\in L_2$ we have that

$$\left\|\mathcal{M}\Lambda-\left[\psi(\cdot,a)+\gamma\max_{\boldsymbol{a}\in\boldsymbol{\mathcal{A}}}\mathcal{P}^{\boldsymbol{a}}\Lambda'\right]\right\|\leq\gamma\,\|\Lambda-\Lambda'\|\,,$$

(c.f. Lemma 3).

Setting $\Lambda=Q$ and $\psi=\Theta$, it can be straightforwardly deduced that for any $Q,\hat{Q}\in L_2$: $\left\|\mathcal{M}Q-\hat{Q}\right\|\leq\gamma\left\|Q-\hat{Q}\right\|$. Hence, using the contraction property of $\mathcal{M}$, we readily deduce the following bound:

$$\max\left\{\left\|\mathcal{M}Q-\hat{Q}\right\|,\left\|\mathcal{M}Q-\mathcal{M}\hat{Q}\right\|\right\}\leq\gamma\left\|Q-\hat{Q}\right\|, \tag{71}$$

We now observe that $\mathfrak{F}$ is a contraction. Indeed, since for any $Q, Q' \in L_2$ we have that:

$$\begin{aligned}
\|\mathfrak{F}Q - \mathfrak{F}Q'\| &= \|\Theta + \gamma P \max\{\mathcal{M}Q, Q\} - (\Theta + \gamma P \max\{\mathcal{M}Q', Q'\})\| \\
&= \gamma \|P \max\{\mathcal{M}Q, Q\} - P \max\{\mathcal{M}Q', Q'\}\| \\
&\leq \gamma \|\max\{\mathcal{M}Q, Q\} - \max\{\mathcal{M}Q', Q'\}\| \\
&\leq \gamma \|\max\{\mathcal{M}Q - \mathcal{M}Q', Q - \mathcal{M}Q', \mathcal{M}Q - Q', Q - Q'\}\| \\
&\leq \gamma \max\{\|\mathcal{M}Q - \mathcal{M}Q'\|, \|Q - \mathcal{M}Q'\|, \|\mathcal{M}Q - Q'\|, \|Q - Q'\|\} \\
&= \gamma \|Q - Q'\|,
\end{aligned}$$

using (71) and again using the non-expansiveness of P.

We next show that the following two bounds hold:

Lemma 5 *For any $Q \in \mathcal{V}$ we have that*

i) $\quad \|\Pi\mathfrak{F}Q - \Pi\mathfrak{F}\bar{Q}\| \leq \gamma \|Q - \bar{Q}\|,$

ii) $\quad \|\Phi r^\star - Q^\star\| \leq \frac{1}{\sqrt{1-\gamma^2}} \|\Pi Q^\star - Q^\star\|.$

Proof 11 *The first result is straightforward since as Π is a projection it is non-expansive and hence:*

$$\|\Pi\mathfrak{F}Q - \Pi\mathfrak{F}\bar{Q}\| \leq \|\mathfrak{F}Q - \mathfrak{F}\bar{Q}\| \leq \gamma \|Q - \bar{Q}\|,$$

using the contraction property of $\mathfrak{F}$. This proves i). For ii), we note that by the orthogonality property of projections we have that $\langle \Phi r^\star - \Pi Q^\star, \Phi r^\star - \Pi Q^\star \rangle$, hence we observe that:

$$\begin{aligned}
\|\Phi r^\star - Q^\star\|^2 &= \|\Phi r^\star - \Pi Q^\star\|^2 + \|\Phi r^\star - \Pi Q^\star\|^2 \\
&= \|\Pi\mathfrak{F}\Phi r^\star - \Pi Q^\star\|^2 + \|\Phi r^\star - \Pi Q^\star\|^2 \\
&\leq \|\mathfrak{F}\Phi r^\star - Q^\star\|^2 + \|\Phi r^\star - \Pi Q^\star\|^2 \\
&= \|\mathfrak{F}\Phi r^\star - \mathfrak{F}Q^\star\|^2 + \|\Phi r^\star - \Pi Q^\star\|^2 \\
&\leq \gamma^2 \|\Phi r^\star - Q^\star\|^2 + \|\Phi r^\star - \Pi Q^\star\|^2,
\end{aligned}$$

after which we readily deduce the desired result.

Lemma 6 *Define the operator H by the following: $HQ(z) = \begin{cases} \mathcal{M}Q(z), & \text{if } \mathcal{M}Q(z) > \Phi r^\star, \\ Q(z), & \text{otherwise}, \end{cases}$*

and $\tilde{\mathfrak{F}}$ by: $\tilde{\mathfrak{F}}Q := \Theta + \gamma PHQ$.

For any $Q, \bar{Q} \in L_2$ we have that

$$\left\|\tilde{\mathfrak{F}}Q - \tilde{\mathfrak{F}}\bar{Q}\right\| \leq \gamma \|Q - \bar{Q}\| \tag{72}$$

and hence $\tilde{\mathfrak{F}}$ is a contraction mapping.

Proof 12 *Using (71), we now observe that*

$$\begin{aligned}
\left\|\tilde{\mathfrak{F}}Q - \tilde{\mathfrak{F}}\bar{Q}\right\| &= \|\Theta + \gamma PHQ - (\Theta + \gamma PH\bar{Q})\| \\
&\leq \gamma \|HQ - H\bar{Q}\| \\
&\leq \gamma \|\max\{\mathcal{M}Q - \mathcal{M}\bar{Q}, Q - \bar{Q}, \mathcal{M}Q - \bar{Q}, \mathcal{M}\bar{Q} - Q\}\| \\
&\leq \gamma \max\{\|\mathcal{M}Q - \mathcal{M}\bar{Q}\|, \|Q - \bar{Q}\|, \|\mathcal{M}Q - \bar{Q}\|, \|\mathcal{M}\bar{Q} - Q\|\} \\
&\leq \gamma \max\{\gamma \|Q - \bar{Q}\|, \|Q - \bar{Q}\|, \|\mathcal{M}Q - \bar{Q}\|, \|\mathcal{M}\bar{Q} - Q\|\} \\
&= \gamma \|Q - \bar{Q}\|,
\end{aligned}$$

again using the non-expansive property of P.

Lemma 7 *Define by $\tilde{Q} := \Theta + \gamma P v^{\tilde{\boldsymbol{\pi}}}$ where*

$$v^{\tilde{\boldsymbol{\pi}}}(z) := \Theta(s_{\tau_k}, a) + \gamma \max_{a \in \mathcal{A}} \sum_{s' \in \mathcal{S}} P(s'; a, s_{\tau_k}) \Phi r^\star(s', I(\tau_k)), \tag{73}$$

then $\tilde{Q}$ is a fixed point of $\tilde{\mathfrak{F}}\tilde{Q}$, that is $\tilde{\mathfrak{F}}\tilde{Q} = \tilde{Q}$.

Proof 13 *We begin by observing that*

$$\begin{aligned} H\tilde{Q}(z) &= H\left(\Theta(z) + \gamma P v^{\tilde{\boldsymbol{\pi}}}\right) \\ &= \begin{cases} \mathcal{M}Q(z), & \text{if } \mathcal{M}Q(z) > \Phi r^\star, \\ Q(z), & \text{otherwise}, \end{cases} \\ &= \begin{cases} \mathcal{M}Q(z), & \text{if } \mathcal{M}Q(z) > \Phi r^\star, \\ \Theta(z) + \gamma P v^{\tilde{\boldsymbol{\pi}}}, & \text{otherwise}, \end{cases} \\ &= v^{\tilde{\boldsymbol{\pi}}}(z). \end{aligned}$$

Hence,

$$\tilde{\mathfrak{F}}\tilde{Q} = \Theta + \gamma P H \tilde{Q} = \Theta + \gamma P v^{\tilde{\boldsymbol{\pi}}} = \tilde{Q}. \tag{74}$$

which proves the result.

Lemma 8 *The following bound holds:*

$$\mathbb{E}\left[v^{\hat{\boldsymbol{\pi}}}(z_0)\right] - \mathbb{E}\left[v^{\tilde{\boldsymbol{\pi}}}(z_0)\right] \leq 2\left[(1-\gamma)\sqrt{(1-\gamma^2)}\right]^{-1} \|\Pi Q^\star - Q^\star\|. \tag{75}$$

Proof 14 *By definitions of $v^{\hat{\boldsymbol{\pi}}}$ and $v^{\tilde{\boldsymbol{\pi}}}$ (c.f (73)) and using Jensen's inequality and the stationarity property we have that,*

$$\begin{aligned} \mathbb{E}\left[v^{\hat{\boldsymbol{\pi}}}(z_0)\right] - \mathbb{E}\left[v^{\tilde{\boldsymbol{\pi}}}(z_0)\right] &= \mathbb{E}\left[P v^{\hat{\boldsymbol{\pi}}}(z_0)\right] - \mathbb{E}\left[P v^{\tilde{\boldsymbol{\pi}}}(z_0)\right] \\ &\leq \left|\mathbb{E}\left[P v^{\hat{\boldsymbol{\pi}}}(z_0)\right] - \mathbb{E}\left[P v^{\tilde{\boldsymbol{\pi}}}(z_0)\right]\right| \\ &\leq \left\|P v^{\hat{\boldsymbol{\pi}}} - P v^{\tilde{\boldsymbol{\pi}}}\right\|. \end{aligned} \tag{76}$$

Now recall that $\tilde{Q} := \Theta + \gamma P v^{\tilde{\boldsymbol{\pi}}}$ and $Q^\star := \Theta + \gamma P v^{\boldsymbol{\pi}^\star}$, using these expressions in (76) we find that

$$\mathbb{E}\left[v^{\hat{\boldsymbol{\pi}}}(z_0)\right] - \mathbb{E}\left[v^{\tilde{\boldsymbol{\pi}}}(z_0)\right] \leq \frac{1}{\gamma}\left\|\tilde{Q} - Q^\star\right\|.$$

Moreover, by the triangle inequality and using the fact that $\mathfrak{F}(\Phi r^\star) = \tilde{\mathfrak{F}}(\Phi r^\star)$ and that $\mathfrak{F}Q^\star = Q^\star$ and $\mathfrak{F}\tilde{Q} = \tilde{Q}$ (c.f. (75)) we have that

$$\begin{aligned} \left\|\tilde{Q} - Q^\star\right\| &\leq \left\|\tilde{Q} - \mathfrak{F}(\Phi r^\star)\right\| + \left\|Q^\star - \tilde{\mathfrak{F}}(\Phi r^\star)\right\| \\ &\leq \gamma\left\|\tilde{Q} - \Phi r^\star\right\| + \gamma\left\|Q^\star - \Phi r^\star\right\| \\ &\leq 2\gamma\left\|\tilde{Q} - \Phi r^\star\right\| + \gamma\left\|Q^\star - \tilde{Q}\right\|, \end{aligned}$$

which gives the following bound:

$$\left\|\tilde{Q} - Q^\star\right\| \leq 2(1-\gamma)^{-1}\left\|\tilde{Q} - \Phi r^\star\right\|,$$

from which, using Lemma 5, we deduce that $\left\|\tilde{Q} - Q^\star\right\| \leq 2\left[(1-\gamma)\sqrt{(1-\gamma^2)}\right]^{-1}\left\|\tilde{Q} - \Phi r^\star\right\|$, after which by (77), we finally obtain

$$\mathbb{E}\left[v^{\hat{\boldsymbol{\pi}}}(z_0)\right] - \mathbb{E}\left[v^{\tilde{\boldsymbol{\pi}}}(z_0)\right] \leq 2\left[(1-\gamma)\sqrt{(1-\gamma^2)}\right]^{-1}\left\|\tilde{Q} - \Phi r^\star\right\|,$$

as required.

Let us rewrite the update in the following way:

$$r_{t+1} = r_t + \gamma_t \Xi(w_t, r_t),$$

where the function $\Xi : \mathbb{R}^{2d} \times \mathbb{R}^p \to \mathbb{R}^p$ is given by:

$$\Xi(w, r) := \phi(z) \left(\Theta(z) + \gamma \max\left\{(\Phi r)(z'), \mathcal{M}(\Phi r)(z')\right\} - (\Phi r)(z)\right),$$

for any $w \equiv (z, z') \in (\mathbb{N} \times \mathcal{S})^2$ where $z = (t, s) \in \mathbb{N} \times \mathcal{S}$ and $z' = (t, s') \in \mathbb{N} \times \mathcal{S}$ and for any $r \in \mathbb{R}^p$. Let us also define the function $\boldsymbol{\Xi} : \mathbb{R}^p \to \mathbb{R}^p$ by the following:

$$\boldsymbol{\Xi}(r) := \mathbb{E}_{w_0 \sim (\mathbb{P}, \mathbb{P})} \left[\Xi(w_0, r)\right]; w_0 := (z_0, z_1).$$

Lemma 9 *The following statements hold for all $z \in \{0, 1\} \times \mathcal{S}$:*

i) $(r - r^\star)\boldsymbol{\Xi}_k(r) < 0, \qquad \forall r \neq r^\star,$

ii) $\boldsymbol{\Xi}_k(r^\star) = 0.$

Proof 15 *To prove the statement, we first note that each component of $\boldsymbol{\Xi}_k(r)$ admits a representation as an inner product, indeed:*

$$\begin{aligned}
\boldsymbol{\Xi}_k(r) &= \mathbb{E}\left[\phi_k(z_0)(\Theta(z_0) + \gamma \max\left\{\Phi r(z_1), \mathcal{M}\Phi(z_1)\right\} - (\Phi r)(z_0)\right] \\
&= \mathbb{E}\left[\phi_k(z_0)(\Theta(z_0) + \gamma \mathbb{E}\left[\max\left\{\Phi r(z_1), \mathcal{M}\Phi(z_1)\right\} |z_0\right] - (\Phi r)(z_0)\right] \\
&= \mathbb{E}\left[\phi_k(z_0)(\Theta(z_0) + \gamma P \max\left\{(\Phi r, \mathcal{M}\Phi)\right\}(z_0) - (\Phi r)(z_0)\right] \\
&= \langle \phi_k, \mathfrak{F}\Phi r - \Phi r \rangle,
\end{aligned}$$

using the iterated law of expectations and the definitions of P and $\mathfrak{F}$.

We now are in position to prove i). Indeed, we now observe the following:

$$\begin{aligned}
(r - r^\star)\, \boldsymbol{\Xi}_k(r) &= \sum_{l=1} (r(l) - r^\star(l)) \langle \phi_l, \mathfrak{F}\Phi r - \Phi r \rangle \\
&= \langle \Phi r - \Phi r^\star, \mathfrak{F}\Phi r - \Phi r \rangle \\
&= \langle \Phi r - \Phi r^\star, (\mathbf{1} - \Pi)\mathfrak{F}\Phi r + \Pi\mathfrak{F}\Phi r - \Phi r \rangle \\
&= \langle \Phi r - \Phi r^\star, \Pi\mathfrak{F}\Phi r - \Phi r \rangle,
\end{aligned}$$

where in the last step we used the orthogonality of $(\mathbf{1} - \Pi)$. We now recall that $\Pi\mathfrak{F}\Phi r^\star = \Phi r^\star$ since $\Phi r^\star$ is a fixed point of $\Pi\mathfrak{F}$. Additionally, using Lemma 5 we observe that $\|\Pi\mathfrak{F}\Phi r - \Phi r^\star\| \leq \gamma\|\Phi r - \Phi r^\star\|$. With this we now find that

$$\begin{aligned}
&\langle \Phi r - \Phi r^\star, \Pi\mathfrak{F}\Phi r - \Phi r \rangle \\
&= \langle \Phi r - \Phi r^\star, (\Pi\mathfrak{F}\Phi r - \Phi r^\star) + \Phi r^\star - \Phi r \rangle \\
&\leq \|\Phi r - \Phi r^\star\| \, \|\Pi\mathfrak{F}\Phi r - \Phi r^\star\| - \|\Phi r^\star - \Phi r\|^2 \\
&\leq (\gamma - 1) \|\Phi r^\star - \Phi r\|^2,
\end{aligned}$$

which is negative since $\gamma < 1$ which completes the proof of part i).

The proof of part ii) is straightforward since we readily observe that

$$\boldsymbol{\Xi}_k(r^\star) = \langle \phi_l, \mathfrak{F}\Phi r^\star - \Phi r \rangle = \langle \phi_l, \Pi\mathfrak{F}\Phi r^\star - \Phi r \rangle = 0,$$

as required and from which we deduce the result.

To prove the theorem, we make use of a special case of the following result:

Theorem 5 (Th. 17, p. 239 in (Benveniste et al., 2012)) *Consider a stochastic process $r_t : \mathbb{R} \times \{\infty\} \times \Omega \to \mathbb{R}^k$ which takes an initial value r_0 and evolves according to the following:*

$$r_{t+1} = r_t + \alpha \Xi(s_t, r_t), \tag{77}$$

for some function $s : \mathbb{R}^{2d} \times \mathbb{R}^k \to \mathbb{R}^k$ and where the following statements hold:

1. $\{s_t | t = 0, 1, \ldots\}$ *is a stationary, ergodic Markov process taking values in* $\mathbb{R}^{2d}$
2. *For any positive scalar* q*, there exists a scalar* μ_q *such that* $\mathbb{E}\left[1 + \|s_t\|^q | s \equiv s_0\right] \leq \mu_q \left(1 + \|s\|^q\right)$
3. *The step size sequence satisfies the Robbins-Monro conditions, that is* $\sum_{t=0}^{\infty} \alpha_t = \infty$ *and* $\sum_{t=0}^{\infty} \alpha_t^2 < \infty$
4. *There exists scalars* c *and* q *such that* $\|\Xi(w, r)\| \leq c\left(1 + \|w\|^q\right)\left(1 + \|r\|\right)$
5. *There exists scalars* c *and* q *such that* $\sum_{t=0}^{\infty} \|\mathbb{E}\left[\Xi(w_t, r) | z_0 \equiv z\right] - \mathbb{E}\left[\Xi(w_0, r)\right]\| \leq c\left(1 + \|w\|^q\right)\left(1 + \|r\|\right)$
6. *There exists a scalar* $c > 0$ *such that* $\|\mathbb{E}[\Xi(w_0, r)] - \mathbb{E}[\Xi(w_0, \bar{r})]\| \leq c\|r - \bar{r}\|$
7. *There exists scalars* $c > 0$ *and* $q > 0$ *such that* $\sum_{t=0}^{\infty} \|\mathbb{E}\left[\Xi(w_t, r) | w_0 \equiv w\right] - \mathbb{E}\left[\Xi(w_0, \bar{r})\right]\| \leq c\|r - \bar{r}\|\left(1 + \|w\|^q\right)$
8. *There exists some* $r^\star \in \mathbb{R}^k$ *such that* $\boldsymbol{\Xi}(r)(r - r^\star) < 0$ *for all* $r \neq r^\star$ *and* $\bar{s}(r^\star) = 0$*.*

Then r_t *converges to* $r^\star$ *almost surely.*

In order to apply the Theorem 5, we show that conditions 1 - 7 are satisfied.

Proof 16 *Conditions 1-2 are true by assumption while condition 3 can be made true by choice of the learning rates. Therefore it remains to verify conditions 4-7 are met.*

To prove 4, we observe that

$$\begin{aligned}
\|\Xi(w, r)\| &= \|\phi(z)\left(\Theta(z) + \gamma \max\left\{(\Phi r)(z'), \mathcal{M}\Phi(z')\right\} - (\Phi r)(z)\right)\| \\
&\leq \|\phi(z)\| \|\Theta(z) + \gamma\left(\|\phi(z')\| \|r\| + \mathcal{M}\Phi(z')\right)\| + \|\phi(z)\| \|r\| \\
&\leq \|\phi(z)\| \left(\|\Theta(z)\| + \gamma\|\mathcal{M}\Phi(z')\|\right) + \|\phi(z)\| \left(\gamma \|\phi(z')\| + \|\phi(z)\|\right) \|r\|.
\end{aligned}$$

Now using the definition of $\mathcal{M}$*, we readily observe that* $\|\mathcal{M}\Phi(z')\| \leq \|\Theta\| + \gamma\|\mathcal{P}^{\pi}_{s' s_t}\Phi\| \leq \|\Theta\| + \gamma\|\Phi\|$ *using the non-expansiveness of* P*.*

Hence, we lastly deduce that

$$\begin{aligned}
\|\Xi(w, r)\| &\leq \|\phi(z)\| \left(\|\Theta(z)\| + \gamma\|\mathcal{M}\Phi(z')\|\right) + \|\phi(z)\| \left(\gamma \|\phi(z')\| + \|\phi(z)\|\right) \|r\| \\
&\leq \|\phi(z)\| \left(\|\Theta(z)\| + \gamma\|\Theta\| + \gamma\|\psi\|\right) + \|\phi(z)\| \left(\gamma \|\phi(z')\| + \|\phi(z)\|\right) \|r\|,
\end{aligned}$$

we then easily deduce the result using the boundedness of ϕ, Θ *and* ψ*.*

Now we observe the following Lipschitz condition on Ξ*:*

$$\begin{aligned}
&\|\Xi(w, r) - \Xi(w, \bar{r})\| \\
&= \|\phi(z)\left(\gamma \max\left\{(\Phi r)(z'), \mathcal{M}\Phi(z')\right\} - \gamma \max\left\{(\Phi \bar{r})(z'), \mathcal{M}\Phi(z')\right\}\right) - \left((\Phi r)(z) - \Phi\bar{r}(z)\right)\| \\
&\leq \gamma \|\phi(z)\| \|\max\left\{\phi'(z')r, \mathcal{M}\Phi'(z')\right\} - \max\left\{(\phi'(z')\bar{r}), \mathcal{M}\Phi'(z')\right\}\| + \|\phi(z)\| \|\phi'(z)r - \phi(z)\bar{r}\| \\
&\leq \gamma \|\phi(z)\| \|\phi'(z')r - \phi'(z')\bar{r}\| + \|\phi(z)\| \|\phi'(z)r - \phi'(z)\bar{r}\| \\
&\leq \|\phi(z)\| \left(\|\phi(z)\| + \gamma \|\phi(z)\| \|\phi'(z') - \phi'(z')\|\right) \|r - \bar{r}\| \\
&\leq c\|r - \bar{r}\|,
\end{aligned}$$

using Cauchy-Schwarz inequality and that for any scalars a, b, c *we have that* $|\max\{a, b\} - \max\{b, c\}| \leq |a - c|$*.*

Using Assumptions 3 and 4, we therefore deduce that

$$\sum_{t=0}^{\infty} \|\mathbb{E}\left[\Xi(w, r) - \Xi(w, \bar{r}) | w_0 = w\right] - \mathbb{E}\left[\Xi(w_0, r) - \Xi(w_0, \bar{r})\|\right] \leq c\|r - \bar{r}\|\left(1 + \|w\|^l\right). \tag{78}$$

Part 2 is assured by Lemma 5 while Part 4 is assured by Lemma 8 and lastly Part 8 is assured by Lemma 9.