Nonetheless, since MARL environments consist of multiple learners, IL suffer from convergence issues (Yang and Wang 2020). Additionally, since IL methods cannot explicitly factor in the behaviour of other agents, achieving coordination among IL agents can be a difficult to achieve, resulting in suboptimal performance in some tasks.

To tackle these challenges, in this paper we propose a new MARL training formalism which we call *semi-centralised* training. In this framework, the agents use knowledge of other agents' policies to compute accurate VFs estimates while using a critic (and policy) that has only the agent's own action and state as an input. Specifically, we introduce *policy embedding*, a technique that enables the agents' VF estimates to capture the present and future behaviour of other agents. As with parameter sharing, a technique used in many MARL methods (e.g., MAPPO (Yu et al. 2021a), QMIX (Rashid et al. 2018)), this procedure requires the agents to communicate their individual policies during training.

Our framework, policy embedding reinforcement learning algorithm (PERLA), embeds the agents' policies within the critic estimation. This is done by sampling the actions of other agents and using them to marginalise the influence of other agents from the critic. The method is scalable and the resulting critic retains a functional form that requires only the agent's own action (and state) as input to the action-value function. Consequently, the resulting critic exhibits the efficient scaling benefits of a fully decentralised critic while **preserving the convergence guarantees** and coordination abilities of a centralised critic.

Many leading MARL methods such as MAPPO (Yu et al. 2021a), IPPO (de Witt et al. 2020a), MADDPG (Lowe et al. 2017) use the actor-critic formalism. For concreteness, we instantiate our methodology within actor-critic architectures which involve both critic and policy updates which is a natural candidate to instantiate our framework. We propose Algorithm 1, which is a PERLA version of MAPPO/IPPO and show it improves the performance of the underlying algorithm. Applications of PERLA could naturally be extended to other degenerate architectures (e.g. value-based methods).

We summarise the advantages of PERLA below.

1) PERLA enables (networked) MARL agents to scale efficiently by factoring the joint policy into a new functional representation of the critic. Crucially, each agent's critic does not use a joint action input. This enables efficient scaling with the number of agents without restrictive VF constraints (validated empirically in Sec. 5.2).

2) Elimination of non-stationarity of IL by incorporating updated policies into the agents' critics through the policy embedding procedure. This avoids the non-stationarity problem in MARL when critics use localised observations.

3) Plug & play enhancement because PERLA can seamlessly incorporate different Actor-Critic algorithms, PERLA significantly boosts performance over the base MARL learners PPO (Schulman et al. 2017), and MAPPO (Yu et al. 2021a) (validated empirically Sec. 5.1).

4) PERLA is theoretically sound and we prove that *i)* PERLA induces a vast reduction of variance of VF estimates (Theorem 1), *ii)* PERLA preserves policy gradient estimators (Theorem 3), and *iii)* Actor-Critic style algorithm based on PERLA converges almost surely to a locally optimal policy profile (Theorem 6).

2 Related Work

Centralised Learning CL is assured to generate policies that are consistent with the desired system goal whenever the IGM principle (Son et al. 2019) is satisfied.[1] In order to realise the IGM principle in the CL framework, QMIX and VDN propose two sufficient conditions of IGM to factorise the joint action-value function. Such decompositions are limited by the joint action-value function class they can represent and can perform badly is systems that do not adhere to these conditions (Wang et al. 2020). QPLEX (Wang et al. 2020) uses a dueling network architecture to factor the joint action-value function. It however has been shown to fail in simple tasks with non-monotonic VFs (Rashid et al. 2020). QTRAN (Son et al. 2019) formulates the MARL problem as a constrained optimisation problem but has been shown to scale poorly in complex MARL tasks such as the StarCraft Multi-Agent Challenge (SMAC) (Peng et al. 2020). WQMIX (Rashid et al. 2020) considers a weighted projection towards better performing joint actions but does not guarantee IGM consistency. Actor-critic methods such as COMA (Foerster et al. 2018b) and MADDPG (Lowe et al. 2017) are popular methods within MARL. Although these methods use CL without restrictive assumptions, they are significantly outperformed by value-based methods such as QMIX and MAPPO (Yu et al. 2021a) on standard MARL benchmarks like SMAC (Peng et al. 2020). Consequently, achieving full expressiveness of the IGM function class with scalability remains an open challenge for MARL.

Parameter Sharing (PS) (Gupta, Egorov, and Kochenderfer 2017) aims to speed up learning by sharing the policy parameters of all the agents during training. In this setup, all agents share the same policy network forcing the agents to share the same policy which is only appropriate in systems with rich symmetries between agents. Leading MARL methods such as MAPPO use PS with proven performance benefits in benchmark domains. Our goal differs from PS since the benefit of PERLA is to enable the agents to embed knowledge of the other agents' behaviours into the agent's own critic estimation and policy. As in PS, PERLA requires communication between agents during training, but in PERLA each agent can maintain its own set of parameters, which is more suitable for environments where agents significantly differ from each other.

Opponent Modelling allows each agent to infer the other agents' policies from their observed behavior. This is widely used in MARL with imperfect information. LOLA (Foerster et al. 2018a) and its recent variant of COLA (Willi et al. 2022) seeks to model the impact of an agent's policy on the opponent's future policy updates. Through behavioral cloning, the agent infers the opponent's parameters from the corresponding state-action trajectories. Based on unsupervised representation learning, several methods predict the other agents' actions by using each agents' own policy ar-

[1]The IGM principle imposes an equivalence between the joint greedy action and the collection of individual greedy actions.