Trustable Policy Collaboration Scheme for Multi-Agent Stigmergic Reinforcement Learning

Xing Xu, Rongpeng Li, *Member, IEEE*, Zhifeng Zhao, *Member, IEEE*, and Honggang Zhang, *Senior Member, IEEE*

***Abstract*— In this letter, we propose a trustable policy collaboration scheme in the paradigm of multi-agent independent reinforcement learning (MAIRL). This trustable policy collaboration scheme is realized by directly mixing the policy parameters of homogeneous agents, for which an upper bound of the mixture metric is derived to guarantee the policy improvement. This trustable policy collaboration scheme can update the behavioral policies of agents distributedly and further improve the performance of MAIRL. In addition, we develop a practical implementation of this trustable policy collaboration scheme, and verify its effectiveness in a mixed-autonomy traffic control simulation scenario through the performance comparison with other typical methods.**

***Index Terms*— Multi-agent independent reinforcement learning, fisher information matrix, trustable policy iteration method, stigmergy.**

I. Introduction

INDEPENDENT reinforcement learning (IRL) [1] is an effective paradigm in practical implementations to alleviate the non-stationary learning problem in the field of multi-agent reinforcement learning (MARL) [2]. In IRL, each agent is commonly limited to partially observe the global environment, and undergoes an independent learning process with only self-related sensations. Accordingly, multi-agent collaboration mechanisms should be introduced to reduce the behavioral localities of IRL agents [3]. In the field of reinforcement learning (RL), an agent with parameterized policy can improve its performance by updating the policy parameters along the gradient descent direction based on samples collected through trial-and-error interaction with the corresponding environment. Typically, the trained policy performance is closely related to the amount and variety of obtained samples, since the more fully explored state space leads to more accurate estimation of the cumulative reward signal. However, due to the possible perturbation of the heterogeneity of learning environments or deployment means to the obtained samples, the policy performance of multiple IRL agents may vary significantly, even though these agents are deployed in the same global environment and have exactly the same facilities.

Stigmergy is a swarm collaboration mechanism that exists widely in natural colonies, and has been used in [4] to collaborate IRL agents through the mutual attractors comprised by digital pheromones. Agents are attracted via these attractors and thus receive informative messages helpful to their learning tasks. Here, we borrow the concept of stigmergy by regarding each agent as an attractor whose attractiveness is proportional to the policy performance. Agents can “approach” the member with larger attractiveness in parameter space so as to get the policy improvement. Furthermore, we propose a trustable policy collaboration scheme to implement this attraction process between agents. In this scheme, any agent can learn from other agents with better policy performance by directly mixing their policy parameters, while a proper mixture metric is determined to guarantee the policy improvement. By continuously mixing policy parameters of various IRL agents, the proposed scheme can update their behavioral policies distributedly and reduce the discrepancy in their policy performance. Since the policy improvement is guaranteed, the trustable policy collaboration scheme can also improve the performance of IRL.

The mixture approach of neural network parameters is familiar in parallelly distributed stochastic gradient descent (SGD) methods, which commonly obtain an over-simplistic average model through training distributed samples in parallel. Applying these methods directly to improve individual policy performance seems inefficient in IRL, since deep RL (DRL) is commonly regarded as a general training task in these methods, and the intrinsic relationship between parameter gradient descent and policy improvement has not been fully considered. On the other hand, as a policy gradient method, the conservative policy iteration algorithm [5] applies a mixture update rule directly for policy distributions to find an approximately optimal policy, and provides explicit lower bounds on the improvement of the cumulative reward signal. The mixture metric for this update rule has been investigated to prevent the catastrophic forgetting problem, which is commonly due to the overlarge step size in parameter updating and will largely decrease the policy performance [5]. However, this policy update method is unwieldy and restrictive in practice, as it is unusual to mix policy distributions directly. Therefore, the trust region policy optimization (TRPO) algorithm [6] has been proposed to replace the aforementioned mixture metric with the Kullback–Leibler (K-L) divergence measure between distributions of the current and target policy. In this letter, we further map this K-L

Manuscript received January 9, 2022; accepted January 15, 2022. Date of publication January 18, 2022; date of current version April 11, 2022. This work was supported in part by the National Key R&D Program of China No. 2020YFB1804800, the National NSF of China No. 61731002 and 62071425, the Zhejiang Key R&D Plan No. 2019C01002 and 2019C03131, Huawei Cooperation Project, and the Zhejiang Provincial NSF of China No. LY20F010016. The associate editor coordinating the review of this letter and approving it for publication was V.-D. Nguyen. *(Corresponding author: Rongpeng Li.)*

Xing Xu, Rongpeng Li, and Honggang Zhang are with the College of Information Science and Electronic Engineering, Zhejiang University, Hangzhou 310000, China (e-mail: hsuxing@zju.edu.cn; lirongpeng@zju.edu.cn; honggangzhang@zju.edu.cn).

Zhifeng Zhao is with the Zhejiang Laboratory, Hangzhou 311100, China (e-mail: zhaozf@zhejianglab.com).

Digital Object Identifier 10.1109/LCOMM.2022.3144451

divergence measure to the parameter space through Fisher information matrix (FIM), so as to improve the policy performance by directly mixing its parameters and take full advantage of the mixture approach of neural network parameters in parallelly distributed training. In addition, we also introduce the policy advantage applied in [5] into multi-agent systems, which can indicate the discrepancy in policy performance between agents and is helpful to reduce this discrepancy with the trustable policy collaboration scheme. We also develop practical implementations of the trustable policy collaboration scheme, and verify its effectiveness through the performance comparison with other typical methods in a mixed-autonomy traffic control simulation scenario.

II. System Model and Problem Formulation

We first present an illustration of multi-agent IRL in Fig. 1. We assume that there are totally N agents involved in a common global environment. Each agent is designed to perform IRL during the decentralized training phase, and act automatically during the decentralized execution phase. In particular, for agent i, $i \in \{1, 2, 3, \cdots, N\}$, we assume a parameterized policy $\pi^{(i)}(s, a; \theta)$, where $s \in \mathcal{S}$ is sampled from the local state space, $a \in \mathcal{A}$ is selected from the individual action space, $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$ is a stochastic policy, and $\theta \in \mathbb{R}^d$, where d represents the dimension of policy parameters. In addition, we define $\rho_0 : \mathcal{S} \to \mathbb{R}$ as the distribution of the initial state s_0. In the system, we suppose that an individual reward r, which is calculated by a reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to [0, R]$, where R represents the maximal reward value, will be returned to each agent immediately after an individual action is performed. And the task objective is to maximize the sum of all the cumulative individual reward. Besides, we assume the relationship between agents is essentially cooperative in a stigmergic manner. To reduce the behavioral localities of agents and improve their cooperation efficiency in IRL, we allow agents to exchange messages through a device-to-device (D2D) collaboration channel with nearby collaborators. Consider the possible communication bandwidth or delay restriction between agents in real-world facilities, we primarily assume that the messages transmitted by agents are limited to policy parameters. In IRL, the task objective of each agent can be formulated as maximizing the cumulative reward

$$\max \eta(\pi) := \mathbb{E}_{s_0}\left[V_\pi(s_0) := \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, a_t) | \pi, s_0 \Big] \right], \quad (1)$$

where $\eta(\pi)$ represents the cumulative reward under the policy π, $V_\pi(s)$ denotes the state value, and $\gamma \in (0, 1)$ represents the discount factor. We also define $Q_\pi(s, a) := \mathcal{R}(s, a) + \gamma \mathbb{E}_{s' \sim P(s'; s, a)} [V_\pi(s')]$ as the state-action value, where $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ is the transition probability distribution, and gives the next-state (i.e., state s') distribution upon taking action a in state s. Therefore, we can get the state-action advantage value under the policy π

$$A_\pi(s, a) = Q_\pi(s, a) - V_\pi(s). \quad (2)$$

We consider the learning process of each agent as an infinite-horizon discounted Markov decision process (MDP). The visitation probability of a certain state s under the policy π can be summarized as

$$d_\pi(s) := \sum_{t=0}^{\infty} \gamma^t \Pr(s_t = s; \pi), \quad (3)$$

where $\Pr(s_t = s; \pi)$ represents the visitation probability of the state s at time t under the policy π.

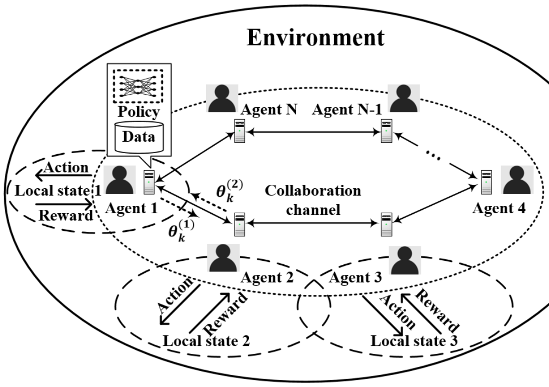

Fig. 1. System model of multi-agent IRL.

Next, for each agent, we will use the subscript label "new" to indicate the expected policy and parameters (i.e., π_{new} and θ_{new}), and use the symbol π and θ to represent the current policy and parameters, respectively. We will also use the symbol $\widetilde{\pi}$ and $\widetilde{\theta}$ to represent the referential policy and parameters that an agent receives from elsewhere. The referential policy and parameters may directly be another agent's current policy and parameters from the D2D collaboration channel, or the current aggregated policy and parameters by averaging over all agents. For each agent, the mixture approach of neural network parameters in parallelly distributed SGD methods can often be summarized as

$$\theta_{\text{new}} = \theta + \alpha(\widetilde{\theta} - \theta), \quad (4)$$

where $\alpha \in [0, 1]$ is the mixture metric. Take the aforementioned average method for example, for agent i, the expected policy parameters are obtained by $\theta_{\text{new}}^{(i)} = \frac{1}{N}\theta^{(i)} + \frac{1}{N}\sum_{j \neq i}^{N} \theta^{(j)}$, and $\alpha = 1 - \frac{1}{N}$ which is influenced by the number of agents involved. Since the neural network parameters represent the policy parameters in DRL, the effect of this mixture approach in (4) on policy improvement should be considered. In DRL, an overlarge step size in parameter updating during policy gradient descent phase may cause the catastrophic forgetting problem, and will largely decrease the policy performance [6]. Besides, the policy performance of multiple IRL agents may vary significantly due to the differences in training samples, which means the mixture metric in (4) between different pairs of collaborative agents may also vary significantly. Therefore, a proper mixture metric in (4) should be determined to inhibit an overlarge step size in parameter updating, and help reduce the discrepancy in policy performance between IRL agents.

III. Trustable Policy Collaboration Scheme

In this section, we propose a trustable policy collaboration scheme, in which the general mixture approach presented in (4) is used to update agents' policy parameters, where an upper bound that the mixture metric complies with is determined by Theorem 1. In this letter, the proposed scheme being trustable means that an improvement in policy performance under this scheme is guaranteed by using two trustable conditions in the derivation of Theorem 1.

Theorem 1: With the referential policy parameters $\widetilde{\theta}$, an agent with current policy parameters θ can improve its cumulative reward through updating θ to θ_{new} according to (4)*, if*

$$\mathbb{A}_\pi(\widetilde{\pi}) > 0 \tag{5}$$

and α in (4) *satisfies*

$$0 < \alpha < \left[2\left(\frac{\mathbb{A}_\pi(\widetilde{\pi})}{C}\right)^{\frac{1}{2}} / \left[(\widetilde{\theta}-\theta)^T G(\theta)(\widetilde{\theta}-\theta)\right]\right]^{\frac{1}{2}}, \tag{6}$$

where $C = \frac{2\varepsilon\gamma}{(1-\gamma)^2}$, $\varepsilon = \max_s \max_a |A_\pi(s,a)|$, $\mathbb{A}_\pi(\widetilde{\pi})$ is the policy advantage of $\widetilde{\pi}$ with respect to π, and $G(\theta)$ is the FIM of policy parameters.

Proof: We get the expected policy through the following mixture update rule, which is used in [5] as a more conservative policy iteration method and $\widetilde{\pi}$ is expected to choose a better action at every state

$$\pi_{\text{new}}(s,a) = (1-\beta)\pi(s,a) + \beta\widetilde{\pi}(s,a), \tag{7}$$

where $\beta \in [0,1]$ denotes the step size. Furthermore, we have the following well-known equation about the cumulative reward of π_{new} and π

$$\eta(\pi_{\text{new}}) - \eta(\pi) = \sum_s d_{\pi_{\text{new}}}(s) \sum_a \pi_{\text{new}}(s,a) A_\pi(s,a). \tag{8}$$

A more recent proof of (8) can be found in [6]. However, the γ-discounted future state distribution under the expected policy (i.e., $d_{\pi_{\text{new}}}(s)$) is hard to obtain. Instead, we can get the following inequality by defining the policy advantage

$$\mathbb{A}_\pi(\widetilde{\pi}) := \mathbb{E}_{s\sim d_\pi(s)}\Big[\mathbb{E}_{a\sim\widetilde{\pi}}\left[A_\pi(s,a)\right]\Big], \tag{9}$$

$$\eta(\pi_{\text{new}}) - \eta(\pi) \geq \beta\mathbb{A}_\pi(\widetilde{\pi}) - C\beta^2, \tag{10}$$

where $C = \frac{2\varepsilon\gamma}{(1-\gamma)^2}$, and $\varepsilon = \max_s \left|\sum_a \widetilde{\pi}(s,a)A_\pi(s,a)\right|$. Detailed proofs of inequality (10) can be found in [5], except that an unnormalized $d_\pi(s)$ is considered here, and $\beta \ll 1$ is typically the case in the conservative policy iteration algorithm. According to (10), we can guarantee the cumulative reward under the expected policy doesn't decrease by keeping the right-side of (10) non-negative. Furthermore, according to [6], let $\varepsilon = \max_s \max_a |A_\pi(s,a)|$ and C be the same, K-L divergence can be introduced to replace β with the conclusion in (10) unchanged. Let $\beta = \sqrt{D_{KL}^{\max}(\pi, \pi_{\text{new}})}$, in order to keep the right-side of (10) positive, we get the following two trustable conditions to guarantee an increase of the cumulative reward

1) $\mathbb{A}_\pi(\widetilde{\pi}) > 0$;
2) $0 < D_{KL}^{\max}(\pi, \pi_{\text{new}}) < \sqrt{\frac{\mathbb{A}_\pi(\widetilde{\pi})}{C}}$,

where $D_{KL}^{\max}(\pi, \pi_{\text{new}}) := \max_s D_{KL}\big(\pi(\cdot|s) \| \pi_{\text{new}}(\cdot|s)\big)$. K-L divergence, or relative entropy, is defined by $D_{KL}(p\|q) = \sum_{i=1}^n p(x_i)\log\frac{p(x_i)}{q(x_i)}$ for the two distributions p and q of discrete random variable x.

On the other hand, the change in policy parameters under the mixture approach in (4) can be represented as $\Delta\theta = \alpha(\widetilde{\theta} - \theta)$. However, the distance measure in parameter space, such as Euclidean measure, cannot be used directly to measure the distance between probability distributions. Therefore, a map (or manifold [7]) should be established to revise the effect of certain changes in policy parameters on the probability distributions. In this letter, the distance between probability distributions is measured by K-L divergence. Furthermore, for any state s in DRL, we can get the K-L divergence resulting from the change in policy parameters under the mixture approach by

$$\begin{aligned} &D_{KL}\big(\pi(s,a;\theta) \| \pi(s,a;\theta+\Delta\theta)\big) \\ &\quad = \int \pi(s,a;\theta)\log\frac{\pi(s,a;\theta)}{\pi(s,a;\theta+\Delta\theta)}dx \qquad (11) \\ &\quad \approx \frac{1}{2}\Delta\theta^T\left[\int \pi(s,a;\theta)\Gamma(s,a;\theta)\Gamma(s,a;\theta)^T dx\right]\Delta\theta \qquad (12) \end{aligned}$$

where $\Gamma(s,a;\theta) = \frac{\partial \log \pi(s,a;\theta)}{\partial\theta}$, and the approximation proofs of equation (12) can be found in [7]. Similarly, we define $G(\theta) := \int \pi(s,a;\theta)\Gamma(s,a;\theta)\Gamma(s,a;\theta)^T dx$ as FIM, which plays the role of the revise map between policy parameters and probability distributions.

Now consider the aforementioned trustable conditions, we can directly get

$$\frac{1}{2}\Delta\theta^T G(\theta)\Delta\theta < \sqrt{\frac{\mathbb{A}_\pi(\widetilde{\pi})}{C}}. \tag{13}$$

Consider the change in policy parameters under the mixture approach in (4) (i.e., $\Delta\theta = \alpha(\widetilde{\theta} - \theta)$), we have the following trustable upper bound for the mixture metric

$$\alpha < \left[2\left(\frac{\mathbb{A}_\pi(\widetilde{\pi})}{C}\right)^{\frac{1}{2}} / \left[(\widetilde{\theta}-\theta)^T G(\theta)(\widetilde{\theta}-\theta)\right]\right]^{\frac{1}{2}}. \tag{14}$$

□

Remark 1: Note that $G(\theta)$ is a positive definite matrix, and thus the sign of $\mathbb{A}_\pi(\widetilde{\pi})$ determines whether the upper bound is positive or negative. If the policy advantage $\mathbb{A}_\pi(\widetilde{\pi})$ is positive, the agent with π can benefit by mixing its policy parameters with those of the agent with $\widetilde{\pi}$, and its mixture metric α will also enlarge with the increase of the policy advantage value. Besides, the deviation between the policy parameters of two agents, which is revised by FIM, can also affect the mixture metric. For example, the mixture metric may remain large when the policy parameters are insensitive to change at some "point", even though the deviation is huge.

Remark 2: To get the upper bound of α, some practical implementations should be developed to evaluate $\mathbb{A}_\pi(\widetilde{\pi})$ and $G(\theta)$. We use Monte-Carlo simulation and the importance sampling estimator to estimate the policy advantage utilizing the off-policy data, specifically

$$\begin{aligned} \mathbb{A}_\pi(\widetilde{\pi}) &= \sum_s d_\pi(s)\sum_a \widetilde{\pi}(s,a)A_\pi(s,a) \\ &= \sum_s d_\pi(s)\sum_a \Big(\widetilde{\pi}(s,a) - \pi(s,a)\Big)A_\pi(s,a) \qquad (15) \\ &\approx \mathbb{E}_{s,a\sim\pi^*}\left[\frac{\widetilde{\pi}(s,a)}{\pi^*(s,a)} - \frac{\pi(s,a)}{\pi^*(s,a)}\right]A_{\pi^*}(s,a), \qquad (16) \end{aligned}$$

where (15) is due to $\sum_s d_\pi(s) \sum_a \pi(s,a) A_\pi(s,a) = 0$. $\pi^*(s,a)$ represents the behavioral policy sampling action a at state s, which commonly denotes the corresponding policy for a sample in the replay buffer. Besides, we can also estimate FIM by Monte-Carlo simulation

$$G(\theta) = \mathbb{E}_{s,a\sim d_\pi(s),\pi(s,a)} \times \left[\left(\frac{\partial \log \pi(s,a)}{\partial \theta} \right) \left(\frac{\partial \log \pi(s,a)}{\partial \theta} \right)^T \right]. \quad (17)$$

However, some software packages applying the K-FAC method to approximate FIM, such as [8], can be more effective when the neural network contains a huge amount of parameters.

Algorithm 1 Trustable Policy Collaboration Scheme

```
Input: $\pi_\theta^{(i)}$ for $i = 1, 2, 3, \dots, N$;
Output: $\pi_{\theta_{\text{new}}}^{(i)}$ for $i = 1, 2, 3, \dots, N$;
Initialize nearby collaborators $\Omega_i$ for $i = 1, 2, 3, \dots, N$, number of iterations $T$, number of samples to estimate policy advantage $M$, number of samples to evaluate FIM $K$;
for iteration $t = 1, 2, 3, \dots, T$ do
  for agent $i = 1, 2, 3, \dots, N$ do
    for each agent $j$ in $\Omega_i$ do
      Receive $\theta^{(j)}$ by the D2D collaboration channel;
      Random select $M$ samples from the replay buffer of agent $i$ under respective policies $\pi^*$ to estimate $\mathbb{A}_{\pi^{(i)}}(\pi^{(j)})$ according to (16);
      if $\mathbb{A}_{\pi^{(i)}}(\pi^{(j)}) > 0$ then
        Random select $K$ samples from the replay buffer of agent $i$ under the current policy $\pi^{(i)}$ to evaluate $G(\theta^{(i)})$ according to (17);
        Get the upper bound of $\alpha_{i,j}$ in Theorem 1;
        Make the mixture metric $\alpha_{i,j}$ slightly less than the calculated upper bound and update $\theta^{(i)}$ by
            $\theta_{\text{new}}^{(i)} = \theta^{(i)} + \alpha_{i,j}\left(\theta^{(j)} - \theta^{(i)}\right)$
      end if
    end for
  end for
end for
Return $\pi_{\theta_{\text{new}}}^{(i)}$ for $i = 1, 2, 3, \dots, N$;
```

An illustration of this trustable policy collaboration scheme is shown in Algorithm 1, which is performed distributedly after agents finishing each IRL process. By determining a proper mixture metric, Algorithm 1 considers the effect of parameter gradient descent on policy improvement, which is normally ignored in parallelly distributed SGD methods, such as federated RL [9]. In Algorithm 1, the referential policy parameters for an agent are obtained from its nearby collaborators (i.e., $\widetilde{\pi} = \pi^{(j)}$), and both the policy advantage and FIM are estimated based on the accumulated samples in its replay buffer, which may be generated under different behavioral polices (i.e., π^*) during the policy gradient phase. In addition, the collaboration between any pair of agents happens only when the policy advantage is positive, and we set the mixture metric slightly less than its calculated upper bound to guarantee not only the policy improvement but also the convergence speed. In the proposed scheme, each agent is attracted by the agent with a positive policy advantage and can improve its cumulative reward through "approaching" this agent in parameter space. Since the policy improvement is guaranteed in this stigmergic collaboration process, the discrepancy in agents' policy performance is reduced and the effectiveness of IRL is further improved.

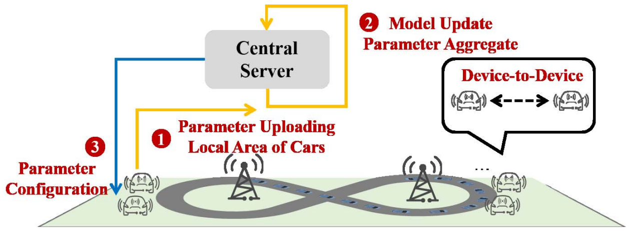

Fig. 2. A mixed-autonomy traffic control simulation scenario.

IV. Numerical Simulation Results

Next, we manifest the effectiveness of the trustable policy collaboration scheme in orchestrating multi-agent systems. As depicted in Fig. 2, a mixed-autonomy traffic control scenario is selected to verify the improvement of task objective (i.e., maximizing the sum of all the cumulative reward) brought by the proposed scheme. This simulation scenario was released by [10] as a new benchmark for the problem of mixed-autonomy traffic control, and also employed in [9] to verify the combination of federated learning with IRL. In particular, there are totally 14 vehicles running circularly along a one-way lane that resembles a shape "8" and an intersection is located at the lane. And each vehicle must adjust its acceleration to pass through this intersection in order to increase the average speed, but slamming on the brakes will be forced on vehicles that are about to crash. During training, each epoch lasts 150 seconds at most. In addition, to achieve the IRL paradigm, the scenario is slightly modified in this article by assigning the related local state of the global environment to each vehicle, including the position and speed of its own, and the vehicle ahead and behind. To create the discrepancy between the samples obtained by different agents, we randomly set the duration of an agent to collect a sample between 0.1 and 10 seconds, and all agents are required to optimize their acceleration policies through the proximal policy optimization (PPO) method [11] every 25 seconds. Note that the proposed scheme is implemented after agents finishing each policy gradient process, and we set $T = 1$ in Algorithm 1.

In Fig. 2, these 14 vehicles are categorized into two classes, namely, 7 cars underlying simulation of urban mobility (SUMO) and 7 DRL-empowered cars. All DRL-empowered cars simultaneously maintain dedicated links to update their parameters either through the central server maintained in the network or the D2D collaboration channel. Besides, considering that SUMO is an open source, human being-level, highly portable and widely used traffic simulation package [10], we treat all 14 cars controlled by the underlying SUMO controllers as the baseline. In Fig. 3, the normalized average speed is an indicator of the task objective. We can verify the effectiveness of multi-agent collaboration by comparing the performance with aforementioned average method (i.e., DRL-Ave) and IRL (i.e., DRL). Note that DRL-Ave can represent the naive federated RL, which utilizes an average method

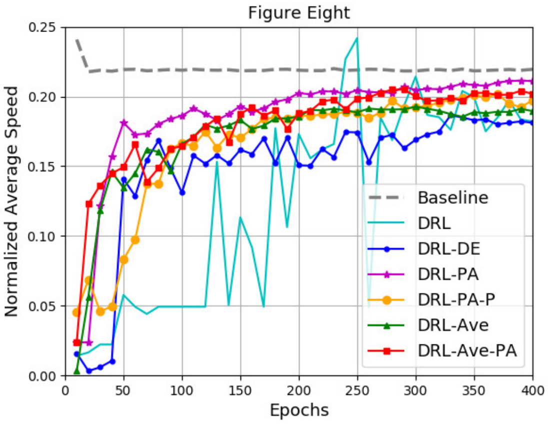

Fig. 3. Normalized average speed under different methods.

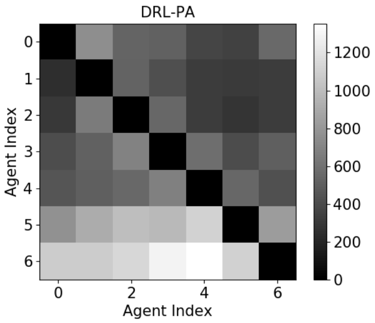

Fig. 4. Number of mixture processes applied in DRL-PA.

to obtain the aggregated parameters, as studied in [9]. Besides, in DRL-DE, each agent mixes its parameters with every other agent in its nearby collaborators holding a fixed mixture metric $\alpha = 0.5$, while the value of α is determined by Algorithm 1 in DRL-PA. Here, the D2D collaboration channel is modeled by D2D communication and nearby collaborators of each agent include all the other agents. Considering the possible conflicts and interference in real communication, in DRL-PA-P, we assume that the link between any two agents is established with a probability $\Pr = 0.5$, in which the mixture metric α is determined by Algorithm 1, otherwise the agents perform IRL. In DRL-Ave-PA, we additionally introduce a mixture metric between each agent's parameters and the aggregated (or averaged) parameters over all agents, which is fixedly equal to 1.0 in DRL-Ave. Note that the mixture metric in DRL-Ave-PA is also determined by Algorithm 1, and each agent only communicates with the central server which maintains the aggregated parameters. Through the performance comparison between the above-mentioned methods, we can observe that methods get performance improvement by deploying the trustable policy collaboration scheme, and a proper mixture metric in the mixture approach is beneficial to multi-agent collaboration, even under poor communication conditions.

We also count the number of mixture processes happened between 7 DRL-empowered cars during the entire training phase in DRL-PA. As indicated in Fig. 4, agent index indicates the No. of an agent in descending order of the duration to collect a sample. The value associated with the square in row i and column j represents the number of times the j_{th} agent has been referred by the i_{th} agent (i.e., $\mathbb{A}_{\pi^{(i)}}(\pi^{(j)}) > 0$ in Algorithm 1). We can observe from Fig. 4 that two different agents have distinct degrees of reference to each other, and the agent with more training samples, as illustrated by row 5 and 6 in Fig. 4, can better benefit from the mixture process, due to a more accurate estimation of the state-action advantage value in (16).

V. Conclusion and Future Works

In this letter, to reduce the possible discrepancy in policy performance between IRL agents, we propose a trustable policy collaboration scheme to update agents' policy parameters through a mixture approach, in which a proper mixture metric is determined to guarantee the policy improvement. Our scheme can be implemented distributedly between an agent and another homogeneous agent or the central server with the aggregated (e.g. averaged) parameters, and only the neural network parameters need to be transmitted between agents which maintains ease of implementation. We also verify the effectiveness of the proposed scheme in a mixed-autonomy traffic control simulation scenario, and validate its superiority. However, the determination of an upper bound of the mixture metric requires a large amount of computing resources, and certain memory is also required to store the past training samples, which remain to be optimized in the future.

References

[1] M. Tan, "Multi-agent reinforcement learning: Independent vs. cooperative agents," in *Proc. 10th Int. Conf. Mach. Learn.*, Amherst, MA, USA, 1993, pp. 330–337.

[2] L. Busoniu, R. Babuska, and B. De Schutter, "A comprehensive survey of multiagent reinforcement learning," *IEEE Trans. Syst., Man, Cybern., C (Appl. Rev.)*, vol. 38, no. 2, pp. 156–172, Mar. 2008.

[3] L. Matignon, G. J. Laurent, and N. Le Fort-Piat, "Independent reinforcement learners in cooperative Markov games: A survey regarding coordination problems," *Knowl. Eng. Rev.*, vol. 27, no. 1, pp. 1–31, Feb. 2012.

[4] X. Xu, R. Li, Z. Zhao, and H. Zhang, "Stigmergic independent reinforcement learning for multiagent collaboration," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Feb. 15, 2021, doi: 10.1109/TNNLS.2021.3056418.

[5] S. Kakade and J. Langford, "Approximately optimal approximate reinforcement learning," in *Proc. 19th Int. Conf. Mach. Learn.*, Sydney, NSW, Australia, 2002, pp. 267–274.

[6] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proc. 32nd Int. Conf. Mach. Learn.*, Lille, France, 2015, pp. 1889–1897.

[7] S.-I. Amari, "Natural gradient works efficiently in learning," *Neural Comput.*, vol. 10, no. 2, pp. 251–276, Feb. 1998.

[8] T. George. (2021). *NNGeometry: Easy and Fast Fisher Information Matrices and Neural Tangent Kernels in PyTorch*. [Online]. Available: https://doi.org/10.5281/zenodo.4532597

[9] X. Xu, R. Li, Z. Zhao, and H. Zhang, "The gradient convergence bound of federated multi-agent reinforcement learning with efficient communication," 2021, *arXiv:2103.13026*.

[10] E. Vinitsky *et al.*, "Benchmarks for reinforcement learning in mixed-autonomy traffic," in *Proc. 2nd Conf. Robot Learn.*, Zurich, Switzerland, 2018, pp. 399–409.

[11] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, *arXiv:1707.06347*.