2018 European Control Conference (ECC)
June 12-15, 2018. Limassol, Cyprus

Distributed Reinforcement Learning Based Optimal Controller For Mobile Robot Formation

Chinmay Shinde[1] and Kaushik Das[1] and Swagat Kumar[1] and Laxmidhar Behera[2]

Abstract— **This paper addresses a problem of attaining desired geometric formation for a group of homogeneous robots using distributed reinforcement learning. The challenges for learning by experience requires huge time and data samples. In multi-agent system (MAS), individual learning becomes more complex as it has to cooperate with its neighboring agent. In this work, a group of homogeneous robots models a single controller while performing a task in a decentralized manner. The framework uses an actor-critic architecture for local learning and its update law is identified using Lyapunov stability analysis. However, a global single controller is achieved by using average consensus protocol. Simulation as well as the experimental results have been given to demonstrate the proposed algorithm.**

Index Terms— **Multi-agent systems, distributed reinforcement learning, actor-critic network, formation control.**

I. INTRODUCTION

Over the last couple of years, MAS has been widely studied in the research community due to its various advantages over a single agent. MAS is used to model complex problems such as urban and air traffic control [1], multi-robot coordination [2], load shedding [3], game-theory based multi-player games [4] and cooperative control [5]. Reinforcement Learning (RL) algorithms are used by evolving systems to perform the task while interacting with the environment. Here, the agents objective is to learn actions that minimize or maximizes the cumulative long-term reward. RL problems can be formulated in Markov Decision Processes (MDP) or as an approximate dynamic programming (ADP). Other continuous-time ADP approaches proposed by Bhasin et. al [6], Vamvoudakis et. al [7], Dierks et. al [8], Wei et. al [9] provide an approximate solution to the Hamilton-Jacobi-Bellman (HJB) equation, which is continuous-time counterpart of the Bellman's optimality principle [10], [11].

In literature, some of the multi-agent learning systems use distributed reward function [12], distributed value function [13] [14] or share the learned Q-Table [15] [16]. Y Xu et. al [12] proposed a distributed reward method using consensus to obtain the average global immediate reward for economic reactive power dispatch. An iterative adaptive dynamic programming (ADP) method developed by Qinglai et al. [14] solves the multi-battery optimal coordination control problem for home energy management system. Soummya et al. [16] presented distributed variant of Q-learning using consensus with innovation term such that each agent sequentially refines its learning parameters by locally processing its instantaneous payoff data and the information received from neighboring agents. Pennesi et. al. [17] proposed a distributed multi-agent actor-critic algorithm using a consensus like algorithm for reward collection problem. This work optimizes Q-table which has discrete state space and actions. But in our approach, we use an actor-critic architecture having continuous state space and action.

In the field of cooperative control of multi-agent systems, formation control is considered as a benchmark problem. The problem objective is to design an algorithm and appropriate protocol such that the robots achieve and preserve a specific geometric shape. Its potential applications are surveillance, search and rescue, exploration and transporting large objects. Traditional formation controllers are classified on the basis of robots sensing capability and interaction topology as position-based, displacement-based and distance-based [18]. Besides the traditional geometry based formation control, researchers also used learning based technique to develop formation control algorithm for MAS. Early RL based formation controllers had primary controllers which used learning either for deciding the formation shape [19] or adaptive weighted behavioral controller [20] suitable for the unstructured environment. For continuous action space, fuzzy based RL formation control algorithms are proposed in [21] [22].

In this work, a formation control problem is addressed in distributed reinforcement learning (DRL) framework. Here ADP is used for solving the above mentioned problem. The key contributions of this work are $1)$ distributed learning technique is used to achieve consensus for designing controller of a single unit. Note that as all the agents are homogeneous, reaching consensus means everybody has same controller $2)$ the algorithm is implemented to achieve formation control.

This paper is organized as follows. Section II gives required mathematical preliminaries. Section III describes the problem statement followed by section IV discussing the proposed algorithm. Section V shows the mobile robot formation result using player-stage simulator and experimental validation results in real time system in section V. The paper is concluded with future outlook in section VII.

II. PRELIMINARIES

This section discusses the mathematical notions used in this work.

[1] Research Scientist, Embedded and Robotics Division, TCS Innovation Lab, Bangalore, India. {chinmay.shinde, kaushik.da, swagat.kumar}@tcs.com
[2] Professor, IIT Kanpur, India. {lbehera@iitk.ac.in}

A. Graph Theory and Average consensus protocol

For a group of networked robots, the communication network is represented as a graph $\mathcal{G}(\nu, \mathcal{E})$ where vertices $\nu = \{v_1, v_2, \ldots, v_N\}$ represents robots and edge $\mathcal{E} = \{(v_i, v_j)\}$ depicts the inter-robot communication link. All vertices v_j connected to vertex v_i by edges $\{(v_i, v_j)\}$ is considered as neighboring robots $\mathcal{N}_i = \{v_j \in \nu : (v_i, v_j) \in \mathcal{E}\}$. Here, the communication graph is considered as undirected graph. The interaction between agents are represented by adjacency matrix $A = \{a_{ij}\}$ where $a_{ii} = 0, a_{ij} = 1$ if $(v_i, v_j) \in \mathcal{E}$ and $a_{ij} = 0$ if $(v_i, v_j) \notin \mathcal{E}$. The degree d_i of vertex v_i is the number of edges incident on it. The max-degree of a graph is given by $d_{\max} = \max_i d_i$. Laplacian matrix is $L = D - A$, where D is a diagonal matrix.

Many families of distributed learning algorithm are based on decentralized average consensus (DAC) protocol. An efficient distributed iterative algorithm for computing an average starting from local measurement vectors [23] are

$$\bar{W} = \frac{1}{N}\sum_{i=1}^{N} W_i^{k*}, \quad \lim_{k\to\infty} W_i^k = \bar{W}_i \tag{1}$$

These protocols assign weight w_{ij} with the edge (v_i, v_j) denoting the confidence that vertex v_i assigns to the information coming from vertex v_j. A few consensus strategies [24] for the DAC protocol like Max Degree, Metropolis Hastings, Minimum asymptotic, Heuristics Laplacian strategy having properties $\mathbf{1}^T W = \mathbf{1}^T$, $W\mathbf{1} = \mathbf{1}$, $\rho(W - \mathbf{1}\mathbf{1}^T/n) < 1$, where $\rho(.)$ denotes the spectral radius of a matrix. For our study we considered Max Degree strategy which defines the weight w_{ij} as

$$w_{ij} = \begin{cases} 1/(d+1) & i \neq j \\ 1 - d_i/(d+1) & i = j \\ 0 & i \neq j, \{i,j\} \notin \mathcal{E} \end{cases}$$

where d is network maximum degree and d_i is vertex v_i degree.

B. Mobile robot formation error dynamics

Robot states are $x_i = \left[q_i^T p_i^T\right]$, where $q_i^T \in \mathcal{R}^2$ is position, $p_i^T \in \mathcal{R}^2$ is velocity. Here, agents internal dynamics is $f_i^T(x_i)$, control gain is $g_i^T(x_i)$ and control effort is $u_i^T(x_i)$. The global network dynamics is defined as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u} \tag{2}$$

where $\mathbf{x} = \left[x_1^T, \ldots, x_i^T, \ldots, x_N^T\right]$, $\mathbf{f}(\mathbf{x}) = \left[f_1^T(x_1), \ldots, f_i^T(x_i), \ldots, f_N^T(x_N)\right]$, $\mathbf{g}(\mathbf{x}) = diag(g_i(x_i))$ and the input $\mathbf{u} = \left[u_1^T(x_1), \ldots, u_i^T(x_i), \ldots, u_N^T(x_N)\right]$. All the robots have limited communication range and can share their local information $\mathcal{X}_i$ with their neighbors ($\mathcal{N}_i$) only. Some agents are provided with the information of the reference trajectory position (q_0) and velocity (p_0). They have aprior information of relative displacement (δ_i), $\forall i \in \{1, \ldots, N\}$, with respect to a trajectory along which the formation is transversing. The objective of the robots is to achieve the desired formation such that $q_i - q_j \to (\delta_i - \delta_j)$, $q_i - q_0 \to \delta_i$, $p_i - p_j \to 0$ and $p_i \to p_0$.

To attain the desired formation, i^{th} robot define local neighboring consensus error $e_i = [e_{iqx}, e_{iqy}, e_{ipx}, e_{ipy}]^T$ as

$$e_i = \begin{bmatrix} \sum_{j\in N_i} a_{ij}(q_{ix} - q_{jx} - \delta_{ijx}) + b_i(q_{ix} - q_{0x} - \delta ix) \\ \sum_{j\in N_i} a_{ij}(q_{iy} - q_{jy} - \delta_{ijy}) + b_i(q_{iy} - q_{0y} - \delta iy) \\ \sum_{j\in N_i} a_{ij}(p_{ix} - p_{jx}) + b_i(p_{ix} - p_{0x}) \\ \sum_{j\in N_i} a_{ij}(p_{iy} - p_{jy}) + b_i(p_{iy} - p_{0y}) \end{bmatrix} \tag{3}$$

where $b_i = \{0, 1\}$ indicates the existence of a direct path from the control center to the i^{th} agent in $\mathcal{G}$. The velocity consensus error ensures that the robots attain the reference trajectory orientation. The global error vector for the graph $\mathcal{G}$ is defined as:

$$e = ((L + B) \otimes I_n)(\bar{\mathbf{x}} - \mathbf{x_0}) \tag{4}$$

where L is the Laplacian matrix for the graph $\mathcal{G}$, $e = \left[e_1^T, e_2^T, \ldots, e_N^T\right]^T \in \mathbb{R}^{Nn}$, $\bar{\mathbf{x}} = \left[(q_1 - \delta_1)^T, p_1^T, \ldots, (q_N - \delta_N)^T, p_N^T\right]^T$, $\mathbf{x_0} = (\mathbf{1_N} \otimes \mathbf{I}_n)\left[q_0, p_0\right]$, $B = diag(b_i) \in \mathbb{R}^{N\times N}$.

III. Optimal Formation control formulation

The robots learn by experience using actor-critic reinforcement learning algorithm [11] [25]. The architecture consist of two networks

- Actor network models action $u(x)$ to minimize the cumulative reward V^u for completing the task.
- Critic network evaluates performed action on the basis of immediate rewards $r(t)$ received from the environment.

In MAS, an individual v_i contribution also depends on its neighboring agents $v_j \in \mathcal{N}_i$ performed the action. Thus the immediate and cumulative reward concerned with action $u_i(e_i)$ for v_i agent can be defined as

$$r_i(e_i, u_i, e_j, u_j) = \int_0^\infty \left(e_i^T Q_{ii} e_i + u_i^T R_{ii} u_i + \sum_{j\in\mathcal{N}_i} u_j^T R_{ij} u_j\right) dt \tag{5}$$

$$V_i^u(e_i(t), u_i, u_j) = \int_t^\infty r_i(e_i(s), u_i(s), x_j, u_j) ds \tag{6}$$

where Q_{ii},R_{ii}, R_{ij} are constant matrix.

For formation control problem the Hamiltonian is defined as

$$H(e_i, u_i, u_j, V_i) = r_i(e_i, u_i, u_j) + V_{ei}((L_i + B_i) \otimes I_n)(\mathbf{x} - \mathbf{x_0}) \tag{7}$$

where $V_{ei} = \frac{\partial V_i}{\partial e_i}$. The optimal value function and action function is defined as

$$V_i(e_i, W_{ic}) = W_{ic}^T \phi(e_i) + \epsilon \tag{8}$$

$$u(e_i, W_{ia}) = \frac{1}{2} R_{ii}^{-1} g^T(x_i) \nabla\phi(e_i) W_{ia} \tag{9}$$

where W_{ia} is actor NN parameter, W_{ic} is critic NN parameter and $\phi(e_i)$ is states feature vector respectively. The optimal value function $V^*(e(t))$ satisfies the HJB equation (7), representing the infinitesimal version of (6) is considered as nonlinear Lyapunov equation with $V^\mu(0) = 0$.

To compute the HJB solution, we approximate the NN parameters by $\hat{W}_{ia}, \hat{W}_{ic}$ respectively. The approximate HJB equation is given by

$$\hat{H}(e_i, \hat{u}_i, u_j, \hat{V}_i) = r_i(e_i, \hat{u}_i, u_j) + \hat{V}_{e_i}\dot{e}_i \tag{10}$$

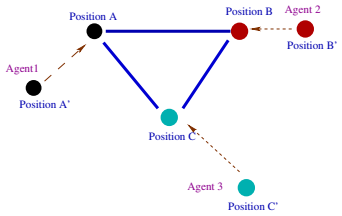


Fig. 1: Formation Control problem

The Bellman error under the principle of optimality is defined as

$$\delta(t) = \hat{H}(e_i, \hat{u}_i, u_j, \hat{V}_i) - H(e_i, u_i, u_j, V_i) \tag{11}$$

$$\delta(t) = \hat{H}(e_i, \hat{u}_i, u_j, \hat{V}_i) - 0 = r_i(e_i, \hat{u}_i, u_j) + \hat{V}_{e_i}\dot{e}_i \tag{12}$$

After differentiating eq. (7), the agent's control action is derived as

$$u_i = -\frac{1}{2}R_{ii}^{-1}g_i^T(x_i)((l_{ii} + b_{ii}) \otimes I_n)^T\hat{V}_{e_i} \tag{13}$$

Fig. 1 shows a group of networked homogeneous robots present at there initial position A', B' , C' from where they would attain the desired formation (A, B, C). Optimal controller (13) for i^{th} robot would be such that $J(u_1^*, \ldots, u_i^*, \ldots, u_N^*) < J(u_1^*, \ldots, u_i, \ldots, u_N^*)$, where J is the formation cost function. It would result in controller parameter $[W_{ia}, W_{ic}] \neq [W_{ja}, W_{jc}]$ for $i \neq j, j \in N$. Here, the goal is to obtain an unique formation controller $[W_{ia}, W_{ic}] = [W_{ja}, W_{jc}]$ for $i \neq j, j \in N$, equivalent to the centralized controller such that

$$(V_e)_{A'} + (V_e)_{B'} + (V_e)_{C'} \leq \frac{(V_e^1)_{A'} + (V_e^2)_{A'} + (V_e^3)_{A'}}{3} + \frac{(V_e^1)_{B'} + (V_e^2)_{B'} + (V_e^3)_{B'}}{3} + \frac{(V_e^1)_{C'} + (V_e^2)_{C'} + (V_e^3)_{C'}}{3} \tag{14}$$

where $(V_e)_{A'}$ is future cumulative reward of proposed controller at position $A^{'}$ in presence of agents at position $B^{'}$ and position $C^{'}$, $(V_e^i)_{A'}$ is the future cumulative reward of multi-agent optimal controller for i^{th} agent at position $A^{'}$.

IV. Proposed Optimal Formation Controller based on Model Consensus

The proposed algorithm aims to build a common model to optimizes the overall system performance over the whole set of local data. It works in a fully distributed fashion and has no requirement of a centralized coordinator during the learning process. The proposed framework consists of two steps 1) local learning step and 2) gossip step. In local step, the robot learns to attain desire formation for minimizing the consensus error using reinforcement learning. In gossip step, the robots will exchange $(e_i, \hat{u}_i, \hat{W}_{ia}, \hat{W}_{ic})$ information and approximate its parameters $(\hat{W}_{ia}, \hat{W}_{ic})$ as local weighted average with those of its neighbors.

The local learning is performed over N iterations which will be referred here as an episode. Each episode would last maximum for Z time steps. Early termination of the episode can occur when the robots collide while learning. Initialize the model parameters and begin the learning with an admissible control law $u_i^{(0,0)}$ generated by linear displacement based formation control with an additional excitation signal. The robot performs action u_i^t and reaches to state e_i^{t+1}. The i^{th} robot communicates with its neighbor $(\mathcal{N}_i)$ and exchange $(e_i, u_i, W_{ia}, W_{ic})$ information. Compute the reward $r_i^t(e_i, u_i, u_j)$ and determine the Bellman's error which would minimize the cost function $V^u(e_i(t), u_i, u_j)$. On the basis of Bellman's error, the agent would update the actor-critic parameters (16)(17) as local step. Thereafter, it would perform gossip step (15) by applying average consensus protocol for the model parameters. Once the critic network is trained such that $V_{e_i}(k) > V_{e_i}(k+1) > \ldots > V_{e_i}(T_f)$. This stable actor network is used to generate optimal control action using (9). For a network of N robots, the local weight update and gossip step are given by Theorem 1.

Theorem 1. *For a group of networking agent, the actor-critic tuning law is defined as:*

$$\hat{W}_{ic}^k = \sum_{j=1}^{N} w_{ij}\hat{W}_{jc} \qquad \hat{W}_{ia}^k = \sum_{j=1}^{N} w_{ij}\hat{W}_{ja} \tag{15}$$

$$\dot{\hat{W}}_{ic} = -\alpha_{1i}\frac{w_{ii}\sigma_i}{(\sigma_i^T\sigma_i + 1)}(e_i^TQ_{ii}e_i + u_i^TR_{ii}u_i + \sum_{j\in\mathcal{N}_i} u_j^TR_{ij}u_j + \hat{W}_{ic}^k\sigma_i) \tag{16}$$

$$\dot{\hat{W}}_{ia} = -\alpha_{2i}((F_{2i}\hat{W}_{ia}^k - F_{1i}\sigma_i^T\hat{W}_{ic}^k) - \frac{1}{4}\bar{D}_i\hat{W}_{ia}w_{ii}m^T\hat{W}_{ic} - \frac{1}{4}\sum_{j\in\mathcal{N}_i}\bar{D}_{ij}\hat{W}_{jc}w_{ij}m^T\hat{W}_{ia}) \tag{17}$$

where

$$\bar{D}_i = \nabla\phi_i(e_i)g(x_i)R_{ii}^{-1}g(x_i)\phi_i^T(e_i) \tag{18}$$

$$\bar{D}_{ij} = \nabla\phi_j(e_j)g(x_j)R_{jj}^{-T}R_{ij}R_{jj}^{-1}g(x_j)\phi_j^T(e_j) \tag{19}$$

$$m = \frac{\sigma_i}{(\sigma_i^T\sigma_i + 1)^2} \tag{20}$$

$$\sigma_i = \nabla\phi_i(e_i)\dot{e}_i \tag{21}$$

$\alpha_{1i} > 0, \alpha_{2i} > 0$ *are learning rate,* $w_{ij} \geq 0$ *is DAC protocol weights,* $F_{1i} > 0$ *and* $F_{2i} > 0$ *are tuning parameters.*

Proof. The proof will be provided in appendix section.

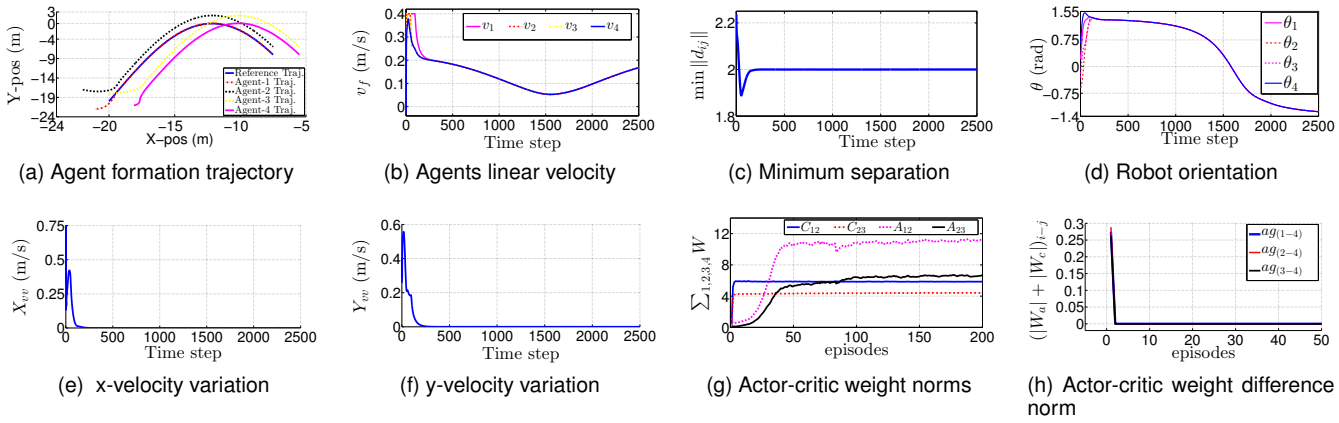
In the next two section, we will validate our work through achieving a formation for the multi-robot system.

V. Simulation Results

Consider 4 Pioneer P3DX robots in Player-Stage simulator with double integrator dynamics $\dot{x}_i = u_i$. The agents are assigned desired displacement with respect to reference trajectory to avoid inter-agent collision. The control commands velocity cv_i, angular velocity ω_i is generated by

$$\begin{aligned} cv_i &= \dot{x}_i \cos\theta_i + \dot{y}_i \sin\theta_i \\ \omega_i &= \frac{1}{l}\left[\dot{x}_i \cos(\theta_i + \frac{\pi}{2}) + \dot{y}_i \sin(\theta_i + \frac{\pi}{2})\right] \end{aligned} \tag{22}$$

where l is wheel axle length and θ_i is robot heading angle. The velocity bounds are $cv_{max} = \pm 0.4$ m/s, $w_{max} = \pm 0.4$ m/s. The agents initial condition $(x_i^0, y_i^0, cv_i^0, \theta_i^0)$ are randomly chosen around (-3,-3,$\pm$0.4,$\pm\frac{\pi}{4}$), (3,-1,$\pm$0.4,$\pm\frac{\pi}{4}$), (0,0,$\pm$0.4,$\pm\frac{\pi}{4}$) and (-1,3,$\pm$0.4,$\pm\frac{\pi}{4}$) respectively with origin at (-20,-20). Agent's communication range (C_s) is 5 meters i.e. if inter-agent distance $(d_{ij}) < 5$ there exist an edge $(v_i, v_j) \in \mathcal{E}$. The reference trajectory which the formation has to transverse is defined as $q_{0x} = 0.05t$, $q_{0y} = 20\sin(0.01t)$. The desired displacement δ_{i0} of v_i agent with respect to reference trajectory is $(-1,-1)$,$(1,-1)$,$(1,1)$,$(-1,1)$ respectively. Only agent v_1 knows about the reference trajectory and its pinning gain



(a) Agent formation trajectory (b) Agents linear velocity (c) Minimum separation (d) Robot orientation

(e) x-velocity variation (f) y-velocity variation (g) Actor-critic weight norms (h) Actor-critic weight difference norm

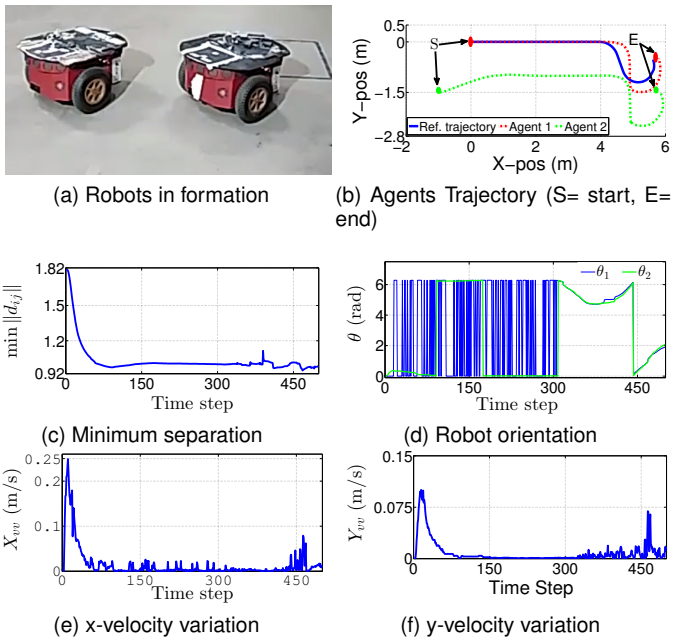
Fig. 2: Four agents training to attain the desired formation at 200 episode

is set as $b_1 = 1$. During learning, the isolated agents with pinning gain $b_i = 0$ has to be provided with reference trajectory information. The parameters for immediate reward function (5) is $Q_i = I$ and $R_{ii} = 1$ and $R_{ij} = 0.3$ $\forall i$ and $\forall j \in N_i$.

The formation control behavior learned by the agents at 200 episode is depicted in Fig. 2. The robots start from random positions and take higher control effort due to which large initial linear velocity (Fig.2b) are observed. Robots attained the desired formation (Fig.2a) and transverse along the reference trajectory with the desired velocity (Fig.2b) and orientation (Fig.2d). Fig. 2e and 2f show velocity variation in X and Y direction. Fig. 2c shows overall minimum inter-agent separation. Fig. 2g and 2h shows the convergence of actor critic network, as the summation of its weights saturates and the inter-agent weight difference of agents 1,2,3 with respect to agent 4 settles to zero. In Fig. 2h the parameter difference norm reaches zero instantly in 1^{st} episode as the algorithm performs consensus step after every local learning. Convergence of the actor network is slow which may be due to a small learning rate.

VI. Experimental Results

The model is experimentally validated with 2 robots formation to maintain a separation of 1m. The experiment is performed with Pioneer P3DX mobile robot from Adept Mobile Robotics. The robot has odometer sensor which provides it coordinates with respect to its initial position. Consider a velocity varying reference trajectory. The desired robot's relative position with respect to the reference trajectory is $(0, 0)$ and $(0, -1)$. Fig. 3 shows both the robots and reference trajectory, minimum separation, $x-$axis velocity and $y-$axis velocity variation for a trajectory of 500 time steps. The initial positions for the two robots are $(0, 0)$ and $(-1, -1.5)$. During experiment, it was observed that initially robot 1 waits for robot 2 to come closer, thereafter it starts following the reference trajectory. In Fig.3b we observe that the robots 1 and 2 successfully maintains a fixed separation between them but is not perfectly tracking the reference trajectory. Fig. 3d shows robots orientation where robot 1 heading angle oscillates around 0° whereas robot 2 has a smooth orientation. From this observation, we induce that since robot 1's relative error with the reference trajectory is small, its action depends largely on the relative error with respect to robot 2. Whereas when the inter-agent distance error reduces, the robot 1 starts experiencing the force to follow the trajectory because of which robot 1 heading angle oscillates. Thus the agent shows both attraction-repulsion with respect to the relative error with the reference trajectory and other robot. Fig. 3c shows minimum separation between



(a) Robots in formation (b) Agents Trajectory (S= start, E= end)

(c) Minimum separation (d) Robot orientation

(e) x-velocity variation (f) y-velocity variation

Fig. 3: Experimental validation using two robots

the agents, we observe as the separation distance reduces to 1 m thereafter the X-axis velocity (Fig. 3e) and Y-axis velocity variation 3f reduce nearly to zero. After 350^{th} instance, the reference trajectory is circular in nature for which it is observed that the minimum separation between the robot has an error of $\pm$ 10cm and y-velocity variation has small jitters.

The robots attain desired formation for constant forward velocity trajectory but for a varying velocity trajectory, the robots try to maintain the separation with small error limit. Video of the experimental results can be seen in the following links.

`https://www.youtube.com/watch?v=OfmtoPgvveE`

VII. CONCLUSIONS

In this paper, we developed a consensus based distributed reinforcement learning algorithm to design an individual controller of a homogeneous multi-agent system. In distributed reinforcement learning, individual learns using approximate dynamic programming with distributed average consensus strategies. Extensive mathematical analysis has been carried out. Simulation results, as well as the experimental result for formation control, are performed using Pioneer P3DX robots. It is observed that they learn to attract towards the desired trajectory and attract-repel with the other robots to attain the desired formation using inter-robot interaction. A Lyapunov based stability analysis is performed. The validation shows the convergence of actor-critic network for the performed task.

Future work extension is to model individual uncertainty as a lower level controller with the existing work acting as a higher level controller as a task behavior model. Further investigation with respect to network delays, segregation of a faulty robot to build a fault tolerant system and addition of new robots at intermediate learning instant would be performed.

APPENDIX

For i^{th} agent, the approximated cumulative value function ($\hat{V}_i$) and action ($\hat{u}_i$) is obtained by using (15) in (8), (9). The actor-critic weight error is defined as:

$$\tilde{W}_{ic} = W_{ic} - \left(\sum_{j=1}^{N} w_{ij}\hat{W}_{jc}\right) \tag{23}$$

$$\tilde{W}_{ia} = W_{ia} - \left(\sum_{j=1}^{N} w_{ij}\hat{W}_{ja}\right) \tag{24}$$

The critic NN is tuned by minimizing the residual error

$$E_i = \frac{1}{2}\delta_i^T\delta_i \tag{25}$$

$$\dot{\hat{W}}_{ic} = -\alpha_1 \frac{dE_i}{d\hat{W}_{ic}} \tag{26}$$

Assumption 1. *The input gain matrix g(x) is known and bounded i.e.* $0 < \|g(x)\| < \bar{g}$*, where* $\bar{g}$ *is a known positive constant*

Assumption 2. *The NN approx error and its gradient are bounded on a compact set* Ω *so that* $\|\epsilon\| < b_\epsilon$ *and* $\|\nabla\epsilon\| < b_{\epsilon_x}$*. The NN activation functions and their gradients are bounded so that* $\|\phi_i(e_i)\| < b_\phi$ *and* $\|\nabla\phi_i(e_i)\| < b_{\phi_e}$

The assumption 1 and 2 applies boundedness to the input control gain, approximation error and input feature vector such that the system bounded is assured. For N agents (i=1,2,...,N) performing a cooperative task will have a coupled error dynamics, for which we consider the following Lyapunov function and its derivative

$$L(t) = \sum_{i=1}^{N}(V_i(e_i) + \frac{1}{2}\operatorname{tr}(\tilde{W}_{ic}^T\alpha_{1i}^{-1}\tilde{W}_{ic}) + \frac{1}{2}\operatorname{tr}(\tilde{W}_{ia}\alpha_{2i}^{-1}\tilde{W}_{ia})) \tag{27}$$

$$\dot{L}(t) = \sum_{i=1}^{N}(\dot{V}_i(e_i) + \operatorname{tr}(\tilde{W}_{ic}^T\alpha_{1i}^{-1}\dot{\tilde{W}}_{ic}) + \frac{1}{2}\operatorname{tr}(\tilde{W}_{ia}\alpha_{2i}^{-1}\dot{\tilde{W}}_{ia})) \tag{28}$$

The term $\dot{V}_i(x)$ is derivative of (8), is added and subtracted with $\frac{1}{2}W_{ic}^T D_i(e_i)W_{ia}$ and substitute $W_{ic}^T\sigma_i$ from HJB equation (7) then

$$\dot{V}_i(e_i) = -e_i^T Q_{ii} e_i - \tfrac{1}{4}W_{ia}^T D_i(e_i)W_{ia} - \textstyle\sum_{j\in N_i}\tfrac{1}{4}W_{ja}^T D_{ij}(e_j)W_{ja} + \varepsilon_{HJBi}(e_i)$$
$$+ \frac{1}{2}W_{ic}D_i(e_i)\tilde{W}_{ia} + \nabla\varepsilon_i^T(e_i)(f_i(x) - \frac{1}{2}D_i(e_i)\nabla\phi_i^T(e_i)\left(\sum_{j=1}^{N} w_{ij}\hat{W}_{ja}\right) \tag{29}$$

For the second term of (28), substitute critic update term (16) and add HJB equation (7). As per Nash equilibrium, the i^{th} agent chooses the best action possible taking into account others action with consideration that the other robots action would remain unchanged. Thus they consider that other robots action are optimal.

$$\tilde{W}_{ic}^T\alpha_{ic}^{-1}\dot{\tilde{W}}_{ic} = \frac{\tilde{W}_{ic}^T w_{ii}^2\sigma_i}{(\sigma_i^T\sigma_i+1)^2}(-\sigma_i^T\tilde{W}_{ic} + \frac{1}{4}\tilde{W}_{ia}^T D_i(e_i)\tilde{W}_{ia}^T + \varepsilon_{HJBi}(e_i)) \tag{30}$$

Substitute (29),(30) in (28)

$$\dot{L} = \textstyle\sum_{i=1}^{N} -e_i^T Q_{ii}e_i - \tfrac{1}{4}W_{ia}^T D_i(e_i)W_{ia} - \sum_{j\in N_i}\tfrac{1}{4}W_{ja}^T D_{ij}(e_j)W_{ja} + \varepsilon_{HJBi}(x)$$
$$+ \frac{1}{2}W_{ic}D_i(e_i)\tilde{W}_{ia} + \nabla\varepsilon_i^T(x)(f_i(x) - \frac{1}{2}D_i(e_i)\nabla\phi_i^T(e_i)\left(\sum_{j=1}^{N} w_{ij}\hat{W}_{ja}\right)$$
$$+ \tfrac{\tilde{W}_{ic}^T w_{ii}^2\sigma_i}{(\sigma_i^T\sigma_i+1)^2}(-\sigma_i^T\tilde{W}_{ic} + \tfrac{1}{4}\tilde{W}_{ia}^T D_i(e_i)\tilde{W}_{ia}^T + \varepsilon_{HJBi}(e_i)) + \tilde{W}_{ia}^T\alpha_{2i}^{-1}\dot{\tilde{W}}_{ia} \tag{31}$$

and we define the actor tuning law as

$$\dot{\hat{W}}_{ia} = -\alpha_2\{(F_{2i}(\sum_{j=1}^{N} w_{ij}\hat{W}_{ja}) - F_{1i}\bar{\sigma}_i^T(\sum_{j=1}^{N} w_{ij}\hat{W}_{jc}))$$
$$- \frac{1}{4}D_i(e_i)\hat{W}_{ja}\frac{w_{ii}\bar{\sigma}_i^T}{m_s}\hat{W}_{jc} - \frac{1}{4}\sum_{j\in\mathcal{N}_i}\bar{D}_{ij}\hat{W}_{jc}w_{ij}m^T\hat{W}_{ia}\} \tag{32}$$

This adds to $\dot{L}$ the terms

$$w_{ii}\tilde{W}_{ia}^T F_{2i}(\sum_{j=1}^{N} w_{ij}\hat{W}_{ja}) - w_{ii}\tilde{W}_{ia}^T F_{1i}\bar{\sigma}_i^T(\sum_{j=1}^{N} w_{ij}\hat{W}_{jc})$$
$$= w_{ii}\tilde{W}_{ia}^T F_{2i}(W_{ia} - \tilde{W}_{ia}) - w_{ii}\tilde{W}_{ia}^T F_{1i}\bar{\sigma}_i^T(W_{ic} - \tilde{W}_{ic})$$
$$= w_{ii}\tilde{W}_{ia}^T F_{2i}W_{ia} - w_{ii}\tilde{W}_{ia}^T F_{2i}\tilde{W}_{ia} - w_{ii}\tilde{W}_{ia}^T F_{1i}\bar{\sigma}_i^T W_{ic} + w_{ii}\tilde{W}_{ia}^T F_{1i}\bar{\sigma}_i^T\tilde{W}_{ic} \tag{33}$$

Overall

$$\dot{L} = \sum_{i=1}^{N} -e_i^T Q_{ii}e_i - \frac{1}{4}W_{ia}^T D_i(e_i)W_{ia} - \frac{1}{4}\sum_{j\in\mathcal{N}_i}\bar{D}_{ij}\hat{W}_{jc}w_{ij}m^T\hat{W}_{ia}$$
$$+ \varepsilon_{HJBi}(e_i) + \tilde{W}_{ic}w_{ii}^2\bar{\sigma}_2(-\bar{\sigma}_2^T\tilde{W}_{ic} + \frac{\varepsilon_{HJBi}(e_i)}{m}) + \varepsilon_{1i}(e_i)$$
$$+ \frac{1}{2}\tilde{W}_{ia}^T D_i(e_i)W_{ic} + \frac{1}{4}\tilde{W}_{ia}^T D_i(x)W_{ic}\frac{w_{ii}^2\bar{\sigma}_2^T}{m_s}\tilde{W}_{ic}$$
$$- \frac{1}{4}\tilde{W}_{ia}D_i(x)W_{ic}\frac{w_{ii}^2\bar{\sigma}_2^T}{m_s}W_{ic} + \frac{1}{4}\tilde{W}_{ia}^T D_i(x)W_{ic}\frac{w_{ii}^2\bar{\sigma}_2^T}{m_s}\tilde{W}_{ia}$$
$$+ w_{ii}\tilde{W}_{ia}^T F_2 W_{ia} - w_{ii}\tilde{W}_{ia}^T F_2\tilde{W}_{ia} - w_{ii}\tilde{W}_{ia}^T F_1\bar{\sigma}_2^T W_{ic}$$
$$+ w_{ii}\tilde{W}_{ia}^T F_1\bar{\sigma}_2^T\tilde{W}_{ic} \tag{34}$$

Now it is desired to introduce norm bounds. It is easy to show that under the assumptions

$$\|\varepsilon_{1i}(x)\| < b_{\varepsilon_x} b_g^2 b_{\phi_x} \sigma_{\min}(R)(\|W_{ic}\| + \|\tilde{W}_{ia}\|) \tag{35}$$

The ε_{HJBi} converges to zeros uniformly as N increases is shown in [26]. Select $\varepsilon > 0$ and $N_0(\varepsilon)$ such that $\sup_{x \in \|\varepsilon_{HJB}\|} < \varepsilon$. Then, assuming $N > N_0$ and writing in terms of $\tilde{Z}_i = \begin{bmatrix} x \\ \bar{\sigma}_i^T \tilde{W}_{ic} \\ \tilde{W}_{ia} \end{bmatrix}$ becomes

$$\dot{L} < \sum_{i=1}^{N} \frac{1}{4}\|W_{ia}\|^2 \|D_i(x)\| + \sum_{j \in N_i} \frac{1}{4}\|W_{ja}\|^2 \|D_{ij}(x)\| + \varepsilon + \frac{1}{2}\|W_{ic}\| b_{\varepsilon_x} b_{\phi_x} b_g^2 \sigma_{\min}(R) - \tilde{Z}_i^T M \tilde{Z}_i + \tilde{Z}_i^T d \tag{36}$$

Define

$$M = \begin{bmatrix} Q_{ii} & 0 & 0 \\ 0 & w_{ii}^2 I & (-\frac{w_{ii}}{2}F_1 - \frac{w_{ii}^2}{8m_s} D_i(x) W_{ic})^T \\ 0 & (-\frac{w_{ii}}{2}F_1 - \frac{w_{ii}^2}{8m_s} D_i(x) W_{ic})^T & w_{ii}F_2 - \frac{w_{ii}^2}{4}(D_i(x) W_{ic} m^T + m W_{ic}^T D_i(x)) \end{bmatrix}$$

$$d = \begin{bmatrix} b_{\varepsilon_x} b_f \\ w_{ii}^2 \frac{\varepsilon}{m_s} \\ (\frac{1}{2}D_i - w_{ii}F_1\bar{\sigma}_2 - \frac{w_{ii}}{4} D_i W_{ic} m^T) W_{ic} + w_{ii} F_2 W_{ia} + \frac{1}{2} b_{\varepsilon_x} b_g^2 b_{\phi_x} \sigma_{\min}(R) \end{bmatrix}$$

$$c = \frac{1}{4}\|W_{ia}\|^2 |D_i(x)| + \sum_{j \in N_i} \frac{1}{4}\|W_{ja}\|^2 \|D_{ij}(x)\| + \varepsilon + \frac{1}{2}\|W_{ic}\| b_{\varepsilon_x} b_g^2 b_{\phi_x} \sigma_{\min}(R) \tag{37}$$

Let the parameters be chosen such that $M > 0$ then

$$\dot{L} < \sum_{i=1}^{N} -|\tilde{Z}|^2 \sigma_{\min}(M) + \|d\| \|\tilde{Z}\| + c + \varepsilon \tag{38}$$

Computing the squares and with finite uniformly ultimately bound weights $\hat{W}_{ic}$, $\hat{W}_{ia}$, the Lyapunov derivative is negative if

$$\|\tilde{Z}\| > \frac{\|d\|}{2\sigma_{\min}(M)} + \left(\frac{d^2}{4\sigma_{\min}^2(M)} + \frac{c + \varepsilon}{\sigma_{\min}(M)}\right)^2 \tag{39}$$

REFERENCES

[1] P. Mannion, J. Duggan, and E. Howley. An experimental review of reinforcement learning algorithms for adaptive traffic signal control. In *Autonomic Road Transport Support Systems*, pages 47–66. Springer, 2016.

[2] H. Modares, F. L. Lewis, and A. Davoudi. Optimal output synchronization of nonlinear multi-agent systems using approximate dynamic programming. In *Neural Networks (IJCNN), 2016 International Joint Conference on*, pages 4227–4232. IEEE, 2016.

[3] Q. Ba and K. Savla. A dynamic programming approach to optimal load shedding control of cascading failure in dc power networks. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*, pages 3648–3653. IEEE, 2016.

[4] K. G. Vamvoudakis, H. Modares, B. Kiumarsi, and F. L. Lewis. Game theory-based control system algorithms with real-time reinforcement learning: How to solve multiplayer games online. *IEEE Control Systems*, 37(1):33–52, 2017.

[5] D. L. Cruz and W. Yu. Path planning of multi-agent systems in unknown environment with neural kernel smoothing and reinforcement learning. *Neurocomputing*, 233:34–42, 2017.

[6] S. Bhasin, R Kamalapurkar, M. Johnson, K. G. Vamvoudakis, F. L. Lewis, and W. E. Dixon. A novel actor–critic–identifier architecture for approximate optimal control of uncertain nonlinear systems. *Automatica*, 49(1):82–92, 2013.

[7] K. G Vamvoudakis and F. L. Lewis. Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem. *Automatica*, 46(5):878–888, 2010.

[8] T. Dierks and S. Jagannathan. Optimal control of affine nonlinear continuous-time systems. In *American Control Conference (ACC), 2010*, pages 1568–1573. IEEE, 2010.

[9] Q. Wei, D. Liu, F. L. Lewis, Y. Liu, and J. Zhang. Mixed iterative adaptive dynamic programming for optimal battery energy control in smart residential microgrids. *IEEE Transactions on Industrial Electronics*, 64(5):4110–4120, 2017.

[10] R. S. Sutton and A. G. Barto. Reinforcement learning: An introduction. 2011.

[11] K. Doya. Reinforcement learning in continuous time and space. *Neural computation*, 12(1):219–245, 2000.

[12] Y. Xu, W. Zhang, and W. Liu. Distributed dynamic programming-based approach for economic dispatch in smart grids. *IEEE Transactions on Industrial Informatics*, 11(1):166–175, 2015.

[13] J. Schneider, W. Wong, A. Moore, and M. Riedmiller. Distributed value functions. *Robotics Institute*, page 264, 1999.

[14] Q. Wei, D. Liu, G. Shi, and Y. Liu. Multibattery optimal coordination control for home energy management systems via distributed iterative adaptive dynamic programming. *IEEE Transactions on Industrial Electronics*, 62(7):4203–4214, 2015.

[15] C. Guestrin, M. Lagoudakis, and R. Parr. Coordinated reinforcement learning. In *ICML*, volume 2, pages 227–234, 2002.

[16] S. Kar, J. M. Moura, and H. V. Poor. Qd-learning: A collaborative distributed strategy for multi-agent reinforcement learning through consensus + innovations. *IEEE Transactions on Signal Processing*, 61(7):1848–1862, 2013.

[17] P. Pennesi and I. C. Paschalidis. A distributed actor-critic algorithm and applications to mobile sensor network coordination problems. *IEEE Transactions on Automatic Control*, 55(2):492–497, 2010.

[18] K. Oh, M. Park, and H. Ahn. A survey of multi-agent formation control. *Automatica*, 53:424–440, 2015.

[19] G. Zuo, J. Han, and G. Han. Multi-robot formation control using reinforcement learning method. *Advances in Swarm Intelligence*, pages 667–674, 2010.

[20] J. Lin, K. Hwang, and Y. Wang. A simple scheme for formation control based on weighted behavior learning. *IEEE Transactions on Neural Networks and Learning Systems*, 25(6):1033–1044, 2014.

[21] H. Zhang, J. Zhang, G. Yang, and Y. Luo. Leader-based optimal coordination control for the consensus problem of multiagent differential games via fuzzy adaptive dynamic programming. *IEEE Transactions on Fuzzy Systems*, 23(1):152–163, 2015.

[22] V. Derhami and Y. Momeni. Applying reinforcement learning in formation control of agents. In *Intelligent Distributed Computing IX*, pages 297–307. Springer, 2016.

[23] R. Fierimonte, S. Scardapane, M. Panella, and A. Uncini. A comparison of consensus strategies for distributed learning of random vector functional-link networks. In *Advances in Neural Networks*, pages 143–152. Springer, 2016.

[24] L. Xiao and S. Boyd. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1):65–78, 2004.

[25] I. Grondman, L. Busoniu, G. A. Lopes, and R. Babuska. A survey of actor-critic reinforcement learning: Standard and natural policy gradients. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6):1291–1307, 2012.

[26] M. Abu-Khalaf and F. L. Lewis. Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network hjb approach. *Automatica*, 41(5):779–791, 2005.