AIAA SciTech Forum
January 3-7, 2022, San Diego, CA & Virtual
AIAA SCITECH 2022 Forum

10.2514/6.2022-2237

GPS-Denied Three Dimensional Leader-Follower Formation Control Using Deep Reinforcement Learning

Robert Selje* and Amer Al-Radaideh†
New Mexico State University, Las Cruces, New Mexico, 88003, USA

Rajdeep Dutta‡, Senthilnath Jayavelu§, and Xiao-Li Li¶
*Agency for Science, Technology and Research (A*STAR), Singapore 138632*

Liang Sun‖
New Mexico State University, Las Cruces, New Mexico, 88003, USA

In this paper, we consider a formation control problem for leader-follower unmanned aerial vehicles (UAVs) in a GPS-denied environment. The distance and the azimuth and elevation angles, defined in a local spherical coordinate frame, are used to describe the relative motion between two UAVs. A novel deep reinforcement learning (DRL) technique is leveraged to generate the required control policies that maneuver a follower UAV in a desired formation with respect to the leader. The effectiveness of the proposed DRL-based leader-follower formation is demonstrated in a simulated environment.

I. Introduction

Autonomy for robotic systems to accomplish cooperative missions is a challenging and critical field of research with significant contributions in real-world applications, such as platooning and formation control for ground mobile robots [1–3] and unmanned aerial vehicles (UAVs) [4, 5] in planetary exploration [6] as well as in search and rescue operations [7]. A fundamental problem in coordination and control of cooperative robotic systems is the leader-follower formation control, which has been well studied for two-dimensional (2D) and three-dimensional (3D) configurations that rely on shared global coordinate information provided by the Global Positioning System (GPS) [8–12]. However, GPS-based positioning demands a large amount of data transfer subjected to a limited communication bandwidth, interference, and attack. Thus, leader-follower formation control using a relative dynamic model has drawn researchers' attention. In a recent work [13], an adaptive formation controller for a 3D leader-follower quadcopter system was developed based on a nonlinear model representing the formation error dynamics. The model takes into account both the relative position in the horizontal plane and the relative heading angle in the presence of uncertainties, which turn it into a 2D controller.

In earlier works [14–16], the Backstepping technique had been used to develop controllers with event-triggering mechanism by leveraging the Lyapunov theory. A 3D dynamics model provides the relative position and orientation formation information, in terms of the relative distance, azimuth and elevation angles of the two UAVs, to describe their positioning in a local spherical coordinate frame. In other words, this 3D relative dynamics model does not rely on the global position information, such as GPS data. Nevertheless, the derivation of Backstepping-based controllers in [14–16] still requires full-state feedback, matrix derivatives and inversion, which demands considerable amount of effort in derivation, state estimation, and resource allocation for on-board implementations.

The research on multi-agent formation control for Unmanned Aerial Vehicles (UAVs) has gained serious attention over the last decade due to its cooperative way of problem solving capability in diverse applications. The related theoretical and practical challenges arise from their coordination and control based on the relative state information. The formation flight control of multiple UAVs in three-dimensional (3D) environments, have been covered in the existing

*MSc student, Klipsch School of Electrical and Computer Engineering, Thomas & Brown Hall, 1125 Frenger Mall, Las Cruces, NM 88003.

†Ph.D. Candidate, Department of Mechanical and Aerospace Engineering, Jett Hall Rm. 104, 1040 S. Horseshoe Street, Las Cruces, NM 88003.

‡Scientist, Agency for Science, Technology and Research (A*STAR), 1 Fusionopolis Way, #21-01 Connexis (South Tower), Singapore 138632

§Scientist, Agency for Science, Technology and Research (A*STAR), 1 Fusionopolis Way, #21-01 Connexis (South Tower), Singapore 138632

¶Scientist, Agency for Science, Technology and Research (A*STAR), 1 Fusionopolis Way, #21-01 Connexis (South Tower), Singapore 138632

‖Assistant Professor, Department of Mechanical and Aerospace Engineering, Jett Hall Rm. 104, 1040 S. Horseshoe Street, Las Cruces, NM 88003. AIAA Senior Member.

literature [10, 11]. However, there is a lack of research towards establishing 3D leader-follower formation control approaches based on the relative dynamics in GPS denied environments, by exploiting machine learning (ML) techniques. It is highly non-trivial to design a formation controller by utilizing local information rather than global information. Such a controller design demands capturing nonlinear input-output (state-to-action) relations involved in the relative dynamics between UAVs (leader-follower), which becomes more complex in the presence of uncertainties. Towards achieving this complex goal, we call for an ML technique that can adapt to the environmental changes without requiring labelled data, hence reinforcement learning (RL) an unsupervised ML technique. Starting with zero knowledge, an RL agent learns from its mistakes while interacting with the associated environment and finally generates the necessary actions required to optimally complete a task. After learning an appropriate policy, it can then be implemented on a real system to succeed in a formation control task.

In this paper, we aim to develop a *model-free leader-follower controller using deep reinforcement learning* (DRL). DRL is an unsupervised learning method that teaches an agent from the scratch through properly rewarding it while interacting with an environment. Specifically, we employ the deep deterministic policy gradient (DDPG) algorithm [17–19] to tackle the continuous action space involved in the current formation control problem. DDPG is an off-policy learning algorithm that trains the associated function approximator, i.e. deep neural network (DNN), by sampling data batch-wise from a replay memory [17, 19]. Once the agent is trained, then it generates the optimal action for the follower to maintain a formation with the leader.

The contributions of this research are: (i) The strength of DRL is utilized to generate appropriate control commands required in a leader-follower formation control problem, (ii) The effectiveness of the proposed control approach is validated in a simulated environment, where an DRL-enabled follower UAV maintains a predefined formation with a leader UAV. Our result analysis includes an investigation on the basis of converging speed, steady-state tracking error, overshoot, and computational efficiency. The rest of the paper is organized as follows. The system dynamics and equations of motion are explained in Section II. Then, our proposed methodology and the achieved simulation results are presented in Section III and Section IV respectively. Finally, section V concludes the paper with potential future work.

II. System Dynamics

A. Coordinate frames

The key coordinate frames associated with the system dynamics of a UAV, i.e., the inertial frame and the vehicle frame [20], are introduced below.

1. The inertial frame $\mathcal{F}^i$

The inertial coordinate system is an earth-fixed coordinate system with its origin at a pre-defined location. In this paper, this coordinate system is referred to the North-East-Down (NED) reference frame. It is common for North to be referred to as the inertial x direction, East to the y direction, and Down to the z direction.

2. The vehicle frame $\mathcal{F}^v$

The origin of the vehicle frame is at the center of mass of a UAV. However, the axes of $\mathcal{F}^v$ are aligned with the axes of the inertial frame $\mathcal{F}^i$.

B. Relative Dynamics of Leader-Follower UAV System

The present work adopts the relative dynamics of a leader-follower UAV system as developed in [14, 15]. Figure 1 shows a top-down view and a side view of the relationship between the positions and orientations of the leader and follower UAVs, respectively. In this paper, the subscript ℓ refers to the leader UAV, while the f subscript refers to the follower UAV.

Let $\mathbf{p}_\ell \in \mathbb{R}^3$ and $\mathbf{p}_f \in \mathbb{R}^3$ be the positions of the leader and the follower UAVs in the inertial frames, respectively. The Line of Sight (LOS) is defined as the line segment drawn between the center of the leader UAV to the center of the follower, as highlighted in Fig. 1. Let $r \in \mathbb{R}$ be the length of the LOS segment, denoting the relative distance between the leader and follower. Let $\eta_\ell \in [0, 2\pi)$ and $\xi_\ell \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ be the azimuth and elevation angles of the follower with respect to the leader in leader's vehicle-1 frame [20], respectively. The origin of the vehicle-1 frame is identical to that of the $\mathcal{F}^v$, i.e. the center of mass of the aircraft. However, the vehicle-1 frame is rotated in the positive right-handed direction about the Down direction by the heading (or yaw) angle ψ. Similarly, let $\eta_f \in [0, 2\pi)$ and $\xi_f \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ be the azimuth and elevation angles of the leader with respect to the follower in follower's vehicle-1 frame, respectively.

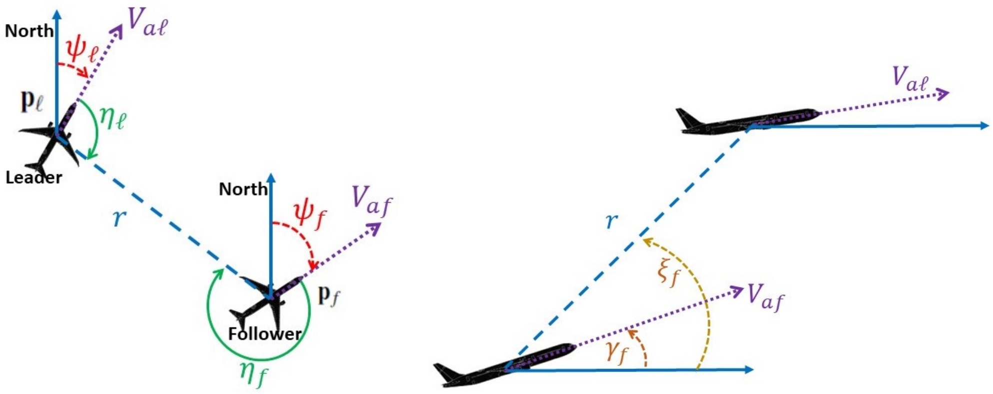


Fig. 1 The overhead view (left) and the side view (right) of the leader and follower UAVs in the three-dimensional spherical coordinate frame [15].

The azimuth and elevation angles are measured between an UAV's airspeed vector and the projection of the LOS segment onto the horizontal and vertical planes, respectively. The positive direction of the azimuth angle is defined as the right-handed rotation about the z-axis of the local-level frame of the UAV. The leader is located to the right of the follower when the azimuth angle η_f lies in $[0, \pi)$, and it is on the left side of the follower when η_f lies in $(\pi, 2\pi)$. To have the follower UAV behind the leader, the azimuth angle η_ℓ should lie in $\left(\frac{\pi}{2}, \frac{3\pi}{2}\right)$; otherwise, the follower appears in front of the leader.

Figure 1 gives a side view of the relationship between the leader and follower with concern to the elevation angle (ξ_f) and the flight path angle (γ_f). The elevation angle reflects whether the leader is below or above the follower. The elevation angle of an UAV is positive if the LOS segment is above the horizontal plane where the UAV is located. The follower is below the leader when the elevation angle ξ_f lies in $\left(0, \frac{\pi}{2}\right]$, and it is above the leader when ξ_f in $\left[-\frac{\pi}{2}, 0\right)$. It is worth mentioning that $\xi_\ell \equiv -\xi_f$.

Let $V_{a\ell}$ and V_{af} be the airspeeds of the leader and follower, γ_ℓ and γ_f be the flight path angles [20] of the leader and follower, and ψ_ℓ and ψ_f be the course angles of the leader and follower, respectively. The dynamics of r, η_f, η_ℓ, and ξ_f are given by [14]

$$\begin{aligned}
\dot{r} &= -V_{a\ell}\left(\mathrm{c}_{\gamma_\ell}\mathrm{c}_{\eta_\ell}\mathrm{c}_{\xi_\ell} + \mathrm{s}_{\gamma_\ell}\mathrm{s}_{\xi_\ell}\right) \\
&\quad -V_{af}\left(\mathrm{c}_{\gamma_f}\mathrm{c}_{\eta_f}\mathrm{c}_{\xi_f} + \mathrm{s}_{\gamma_f}\mathrm{s}_{\xi_f}\right), \qquad (1a) \\
\dot{\eta}_f &= \frac{V_{af}\mathrm{c}_{\gamma_f}\mathrm{s}_{\eta_f} + V_{a\ell}\mathrm{c}_{\gamma_\ell}\mathrm{s}_{\eta_\ell}}{r\mathrm{c}_{\xi_f}} - \psi_f, \qquad (1b) \\
\dot{\eta}_\ell &= \frac{V_{a\ell}\mathrm{c}_{\gamma_\ell}\mathrm{s}_{\eta_\ell} + V_{af}\mathrm{c}_{\gamma_f}\mathrm{s}_{\eta_\ell}}{r\mathrm{c}_{\xi_\ell}} - \psi_\ell, \qquad (1c) \\
\dot{\xi}_f &= \frac{V_{af}}{r}\left(\mathrm{c}_{\gamma_f}\mathrm{c}_{\eta_f}\mathrm{s}_{\xi_f} - \mathrm{s}_{\gamma_f}\mathrm{c}_{\xi_f}\right) \\
&\quad +\frac{V_{a\ell}}{r}\left(\mathrm{s}_{\gamma_\ell}\mathrm{c}_{\xi_\ell} - \mathrm{c}_{\gamma_\ell}\mathrm{c}_{\eta_\ell}\mathrm{s}_{\xi_\ell}\right). \qquad (1d)
\end{aligned}$$

where $\mathrm{s}_* \triangleq \sin *$ and $\mathrm{c}_* \triangleq \cos *$. Letting $\rho \triangleq \frac{1}{r}$, we have $\dot{\rho} = -\rho^2 \dot{r}$ and

$$\begin{aligned}
\dot{\rho} &= \rho^2 V_{a\ell} \left(\mathrm{c}_{\gamma_\ell} \mathrm{c}_{\eta_\ell} \mathrm{c}_{\xi_\ell} + \mathrm{s}_{\gamma_\ell} \mathrm{s}_{\xi_\ell}\right) \\
&\quad + \rho^2 V_{af} \left(\mathrm{c}_{\gamma_f} \mathrm{c}_{\eta_f} \mathrm{c}_{\xi_f} + \mathrm{s}_{\gamma_f} \mathrm{s}_{\xi_f}\right), && \text{(2a)} \\
\dot{\eta}_f &= \frac{\rho}{\mathrm{c}_{\xi_f}} \left(V_{af} \mathrm{c}_{\gamma_f} \mathrm{s}_{\eta_f} + V_{a\ell} \mathrm{c}_{\gamma_\ell} \mathrm{s}_{\eta_\ell}\right) - \dot{\chi}_f, && \text{(2b)} \\
\dot{\eta}_\ell &= \frac{\rho}{\mathrm{c}_{\xi_\ell}} \left(V_{a\ell} \mathrm{c}_{\gamma_\ell} \mathrm{s}_{\eta_\ell} + V_{af} \mathrm{c}_{\gamma_f} \mathrm{s}_{\eta_\ell}\right) - \dot{\chi}_\ell, && \text{(2c)} \\
\dot{\xi}_f &= \rho V_{af} \left(\mathrm{c}_{\gamma_f} \mathrm{c}_{\eta_f} \mathrm{s}_{\xi_f} - \mathrm{s}_{\gamma_f} \mathrm{c}_{\xi_f}\right) \\
&\quad + \rho V_{a\ell} \left(\mathrm{s}_{\gamma_\ell} \mathrm{c}_{\xi_\ell} - \mathrm{c}_{\gamma_\ell} \mathrm{c}_{\eta_\ell} \mathrm{s}_{\xi_\ell}\right). && \text{(2d)}
\end{aligned}$$

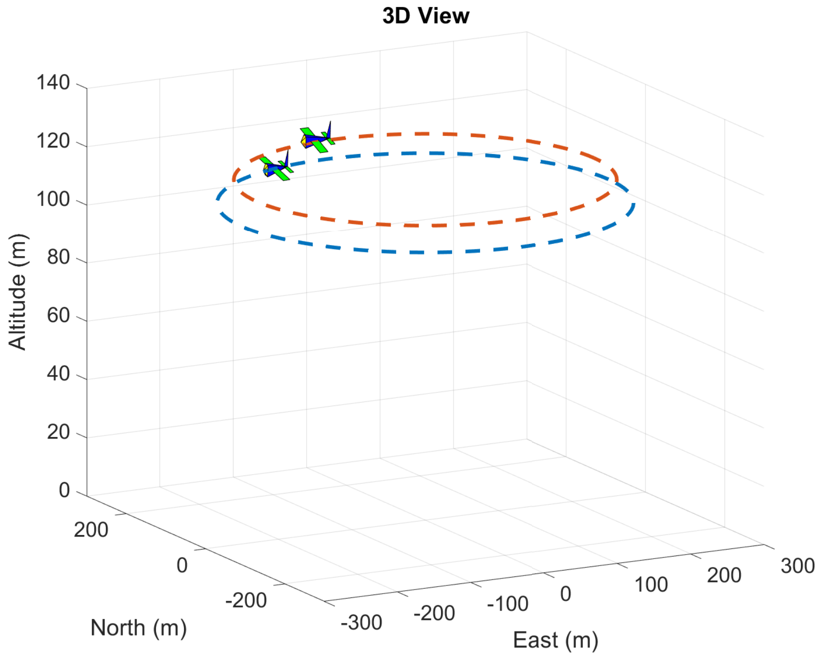


Fig. 2 A follower UAV is regulating to follow a predefined formation with respect to a leader UAV in simulation.

III. Methodology

A. Simulated Environment and RL Architecture

In our proposed method, we apply reinforcement learning to generate appropriate control commands for the follower UAV such that it can fly in a formation with the leader UAV, as shown in Figure 2. Reinforcement learning teaches an agent to adapt and take necessary decisions (actions) to achieve a target, by rewarding its good moves and penalizing its bad moves with respect to a goal-oriented task. In the current RL framework, we have five inputs describing the relative distance (ρ), the azimuth and elevation angles of the two UAVs (η_l and η_f), and the airspeeds of the two UAVs (V_{al} and V_{af}). We assume that the relative distance between the leader and follower UAVs is determined using onboard sensors, such as a stereo camera or a LiDAR. The output of RL are the control commands to the follower UAV, i.e. airspeed and roll angle.

Figure 3 shows the proposed block diagram of the leader follower formation control. In our case, the environment includes two UAV as shown in Figure 2, where both the movement and position for the leader and the follower UAVs are

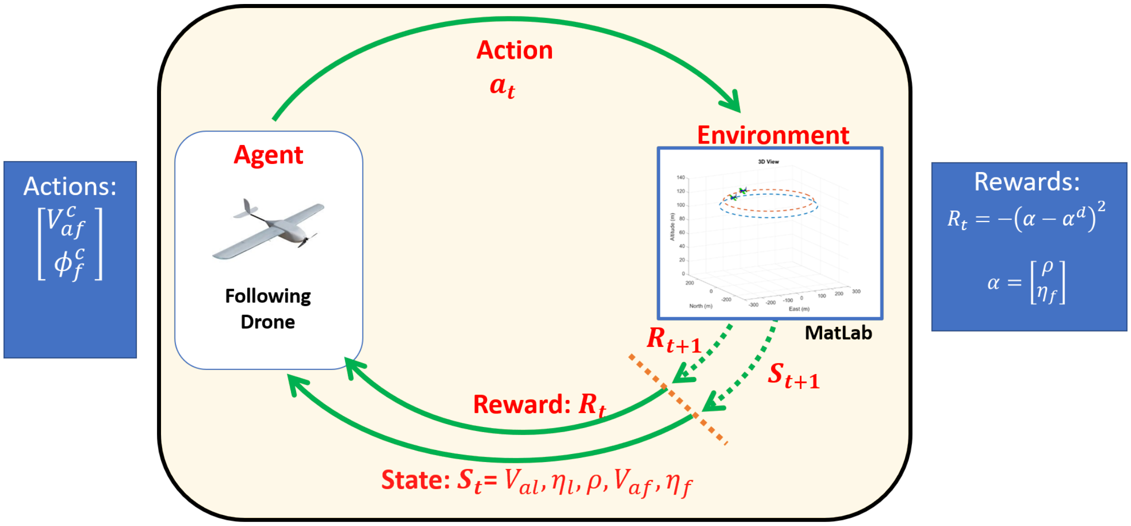

Actions:
$\begin{bmatrix} V_{af}^{c} \\ \phi_{f}^{c} \end{bmatrix}$
Action
a_t
Agent
Following Drone
Environment
3D View
MatLab
R_{t+1}
S_{t+1}
Reward: R_t
State: $S_t = V_{al}, \eta_l, \rho, V_{af}, \eta_f$
Rewards:
$R_t = -(\alpha - \alpha^d)^2$
$\alpha = \begin{bmatrix} \rho \\ \eta_f \end{bmatrix}$

Fig. 3 Block diagram of the proposed reinforcement learning based method for the leader-follower formation control problem.

defined in the polar coordinate frame. The observation made by the follower is the relative distance, the azimuth and elevation angles of the two UAVs. The follower, or agent, performs an action in the environment, and based on the action, a reward is given.

B. Deep Deterministic Policy Gradient (DDPG) Method

Consider a Markov decision process (MDP) with continuous state and action spaces, $\mathcal{S}$ and $\mathcal{A}$, respectively. The states are sampled from a density function $p(s)$ and the actions are sampled from a policy distribution $\pi_\theta(a|s)$, where the policy π is parameterized by a deep neural network (*Actor*) parameter θ. At time instant t, an RL agent in the current state $s_t \sim p(s)$ takes an action $a_t \sim \pi_\theta(a|s_t)$ to reach the future state $s_{t+1} \sim p(.|s_t, a_t)$, and the reward obtained for this state transition is $r_t(s_t, a_t)$. A trajectory $(s_1, a_1, r_1, s_2, ...)$ comprising many state transitions, gives the cumulative reward defined as return $R_t = \sum_{t=1}^{\infty} \gamma^{t-1} r_t$. Here, $\gamma \in (0, 1)$ is the discount factor that assigns different weights to the rewards obtained at different time steps [17, 18]. The expected return by following a policy π is given by

$$Q^{\pi}(s, a) = \mathbb{E}_{\pi(a_t|s_t), p(s_{t+1}|s_t, a_t)}[\sum_{t=1}^{\infty} \gamma^{t-1} r_t | s_1 = s, a_1 = a] \tag{3}$$

Reinforcement learning aims to determine an optimal policy that maximizes the expected return. Towards this, an Actor network parameterizes π with a parameter θ, such that the expected return $\mathbb{E}_{p(s), \pi_\theta(a|s)} Q^{\pi_\theta}(s, a)$ gets maximized. However, the vanilla estimator [18] of the action-value function suffers from high variance leading to a slow convergence. To mitigate this issue, the action-value function is estimated by a *Critic* network denoted by $\hat{Q}(s, a)$, whose parameters are learned to maintain $\hat{Q}(s, a) \approx Q^{\pi_\theta}(s, a)$. The optimization problem becomes

$$\theta^* = \arg\max_{\theta} \ \mathbb{E}_{p(s), \pi_\theta(a|s)} \hat{Q}(s, a) \ . \tag{4}$$

Here, the *Critic* network parameters ϕ are estimated by minimizing a mean-square Bellman loss function, as follows.

$$L(\phi) = \mathbb{E}_{(s_t, a_t, r_t, s_{t+1}) \sim \mathcal{D}} [\{\hat{Q}_\phi(s_t, a_t) - (r_t + \gamma \max_{a_{t+1}} \hat{Q}_\phi(s_{t+1}, a_{t+1}))\}^2] \ , \tag{5}$$

where $\mathcal{D}$ denotes the replay buffer that stores state transition tuple information to pass past experiences during training data sampling. The class of *Actor-Critic* (AC) methods solve the optimization problem (4) by using the gradient of the expected return, as follows.

$$\theta \longleftarrow \theta + \alpha \mathbb{E}_{p(s), \pi_\theta(a|s)} [\nabla_\theta \log \pi_\theta(a|s) \hat{Q}(s, a)] \ . \tag{6}$$

The DDPG algorithm belongs to the class of AC methods, which utilizes the first order information of the *Critic* while training the *Actor*. The associated deterministic policies are updated along the gradient ascent, given as:

$$\theta \leftarrow \theta + \alpha \mathbb{E}_{p(s)} \left[\nabla_\theta \pi_\theta(s) \nabla_a \hat{Q}(s,a)|_{a=\pi_\theta(s)} \right] . \tag{7}$$

Note that DDPG trains a deterministic policy in an off-policy manner so that sufficient state transition information can be passed for proper network training [18, 19].

Reward Shaping: The reward function design is crucial in training/teaching an RL agent so that it can act and adapt according to the desire. The reward is defined as the difference between the control commands to the follower UAV, given by

$$r_t = -(\rho - \rho^d)^2 - (\eta_l - \eta_l^d)^2 . \tag{8}$$

According to the definition (8), the reward values become more negative when the follower UAV is far away from the formation with respect to the leader, and the same tend to zero as the follower approaches the desired formation. We intend to penalize the RL agent with a lower score when its outputs are deviated from the desired parameters, i.e., ρ^d and η_l^d. The learning objective is to maximize the reward over episodes.

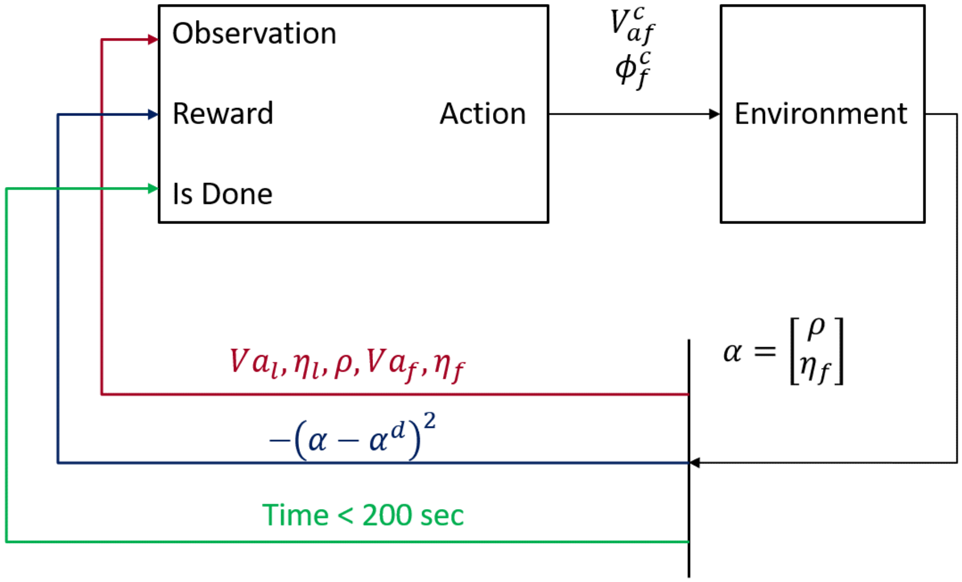


Fig. 4 Reinforcement learning architecture in MATLAB/Simulink for the leader-follower formation control problem. Note: $(\alpha - \alpha_d)^2 \triangleq (\alpha - \alpha_d)^T (\alpha - \alpha_d)$

Network Architecture: The actor-network is composed of twelve total layers. The first layer takes in the observation data with an input size of five. The input layer is followed by five fully connected layers, with a Relu activation in between each layer. The first fully connected layer contains 256 nodes, while the other layers contain 400 nodes. At the end of the fully connected layers, a sigmoid activation is used to bound the input within an interval of (0, 1). The actor-network is concluded with a scaling layer that shifts the result to the range of the specified actions. We set the V_{af}^c signal between 2 m/s to 30 m/s, and the ϕ_f^c control signal range between -45 degrees and 45 degrees. A set of actions will be obtained from the network and be used by the follower UAV as control commands.

An DDPG-based RL framework employs a critic-network and an actor-network to estimate the value function and the policy distribution, respectively. As shown in Figs. 4 and 5, the critic network uses the observation and action to evaluate how well the actor network performs. The actor side of the network contains a single fully connected layer of 400 nodes, while the observation side has two fully connected layers with a Relu activation in between. The first and second fully connected layers of the observation side contain 256 and 400 nodes, respectively. The outputs of the two layers are combined before feeding into a final Relu activation.

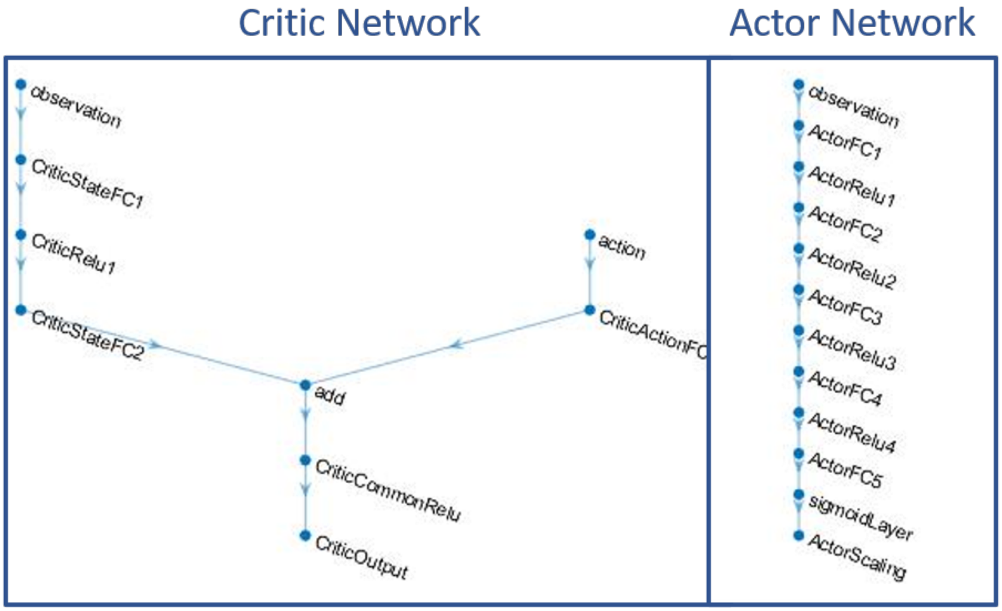


Fig. 5 Actor architecture showing each of the layers.

IV. Simulation Results

We trained the RL agent on a Dell desktop computer with an octa-core Intel i7 processor operating at 3.40 GHz per processor and 16-GB dual-channel (RAM). The RL training process undergoes 400 episodes with each one comprising 1000 iteration. We carried out several simulations with 25 different combinations of the learning rates, ranging from 1×10^{-6} to 1×10^{-10}, of the actor and critic networks. The best overall performing pair of the learning rates are 1×10^{-7} for the actor and 1×10^{-5} for the critic. At each time step of the training, the follower UAV receives a set of observations from the environment that contains the velocities (V_{al} and V_{af}), roll angles (η_l and η_f) of the two UAVs, along with the inverse of their relative distance (ρ). Based on the observation, the RL agent chooses an action for the follower UAV to take, which is the velocity (V^c_{af}) and roll angle (ϕ^c_f) commands. After training for 400 episodes, the RL agents received a reward of -133.7053, as shown in Fig. 6.

The learned policy is then used to generate the control commands to the follower UAV to perform the desired behavior. Fig. 7 shows that the follower UAV succeeds in maintaining the desired formation with respect to the leader. The proposed control approach is validated on a simulated environment where the follower UAV is commanded to follow a predefined formation with respect to a leader UAV moving along a planar circular orbit with constant airspeed and orbit radius.

To further test the proposed technique, noisy observations were generated by using the ground-truth values of $(\rho, \eta_l, \eta_f, V_{al}, V_{af})$ added with the Gaussian noise of $\mathcal{N}(0, \sigma_*^2)$, $* = \{\rho, \eta_l, \eta_f, V_{al}, V_{af}\}$, where $\sigma_\rho = 0.01$, $\sigma_{\eta_l} = \sigma_{\eta_f} = 0.01$, and $\sigma_{V_{al}} = \sigma_{V_{af}} = 0.1$. Figure 8 shows the roll angle action generated using the ground-truth and noisy observations of the environment. The resulting trajectories of the follower UAV in two cases are very close to the one shown in Fig 7. It can be seen that the learned RL policy is able to drive the follower UAV to fly towards the desired formation with noisy observations.

V. Conclusion

In this paper, we leverage a reinforcement learning technique that generates appropriate control commands, airspeed and roll angle, to a follower UAV for maintaining a desired formation with a leader. The proposed RL-based approach does not require any global information and only utilizes local vehicle-frame information. The impact of this non-conventional approach is demonstrated using simulations with 6-dof UAV dynamics, where the follower UAV succeeds in flying in a formation with the leader moving along a planar circle. Further, the leader-follower formation is achieved even in the presence of observation noise, which justifies the robustness of the proposed RL-based controller.

From an ML perspective, an agent learns the desired behavior by maximizing rewards over episodes; therefore,

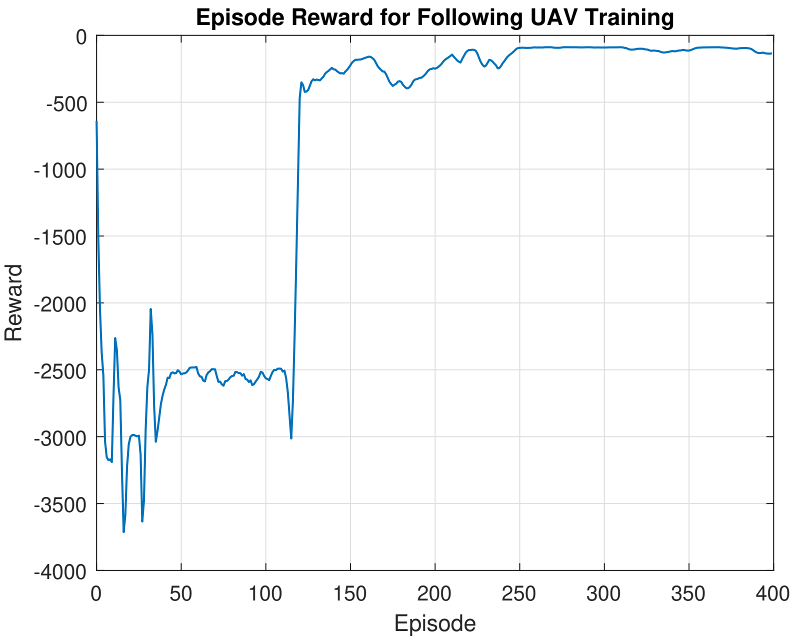


Fig. 6 Episode reward evolution for the follower UAV during a training process.

reward shaping plays an important role in RL. In future, we plan to improve the reward definition by normalizing it so that the leader-follower formation control can be achieved under various leader trajectories (not only a 2D circle). To this end, the present research highlights the use of reinforcement learning for leader-follower formation control in GPS-denied environments, which has potential to open doors to exciting research on multi-agent cooperative control.

References

[1] Farag, A., Hussein, A., Shehata, O. M., García, F., Tadjine, H. H., and Matthes, E., "Dynamics Platooning Model and Protocols for Self-Driving Vehicles," *2019 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2019, pp. 1974–1980.

[2] Ramaswamy, S. P., and Balakrishnan, S., "Formation control of car-like mobile robots: A Lyapunov function based approach," *2008 American Control Conference*, IEEE, 2008, pp. 657–662.

[3] Xu, Z., Schröter, M., Necsulescu, D., Ma, L., and Schilling, K., "Formation control of car-like autonomous vehicles under communication delay," *Proceedings of the 31st Chinese Control Conference*, IEEE, 2012, pp. 6376–6383.

[4] Zhang, M., Liu, Z., Li, H., Huang, H., and Wang, X., "Leader-Follower Formation Control of Unmanned Aerial Vehicles Based on Active Disturbances Rejection Control," *Proceedings of the 2019 4th International Conference on Automation, Control and Robotics Engineering*, 2019, pp. 1–6.

[5] Wang, X., Yu, Y., and Li, Z., "Distributed sliding mode control for leader-follower formation flight of fixed-wing unmanned aerial vehicles subject to velocity constraints," *International Journal of Robust and Nonlinear Control*, 2020.

[6] Schenker, P. S., Huntsberger, T. L., Pirjanian, P., Trebi-Ollennu, A., Das, H., Joshi, S. S., Aghazarian, H., Ganino, A., Kennedy, B. A., and Garrett, M. S., "Robot work crews for planetary outposts: close cooperation and coordination of multiple mobile robots," *Sensor Fusion and Decentralized Control in Robotic Systems III*, Vol. 4196, International Society for Optics and Photonics, 2000, pp. 210–220.

[7] Cardona, G. A., and Calderon, J. M., "Robot swarm navigation and victim detection using rendezvous consensus in search and rescue operations," *Applied Sciences*, Vol. 9, No. 8, 2019, p. 1702.

[8] Semsar, E., and Khorasani, K., "Adaptive formation control of UAVs in the presence of unknown vortex forces and leader commands," *2006 American Control Conference*, 2006, pp. 3563–3568. doi:10.1109/ACC.2006.1657270.

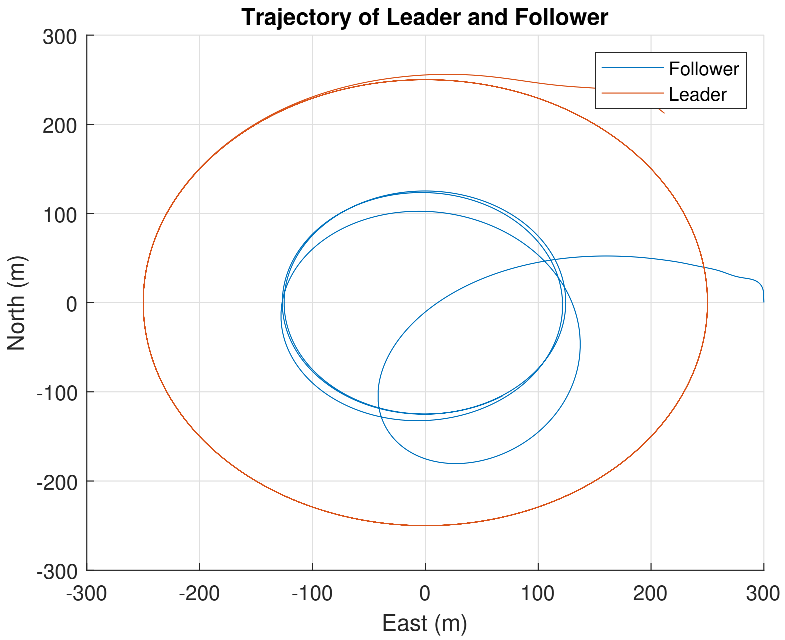


Fig. 7 The predefined leader trajectory (in red) and the resulting trajectory of the follower UAV (in blue).

[9] Dutta, R., Sun, L., Kothari, M., Sharma, R., and Pack, D., "A cooperative formation control strategy maintaining connectivity of a multi-agent system," *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, 2014, pp. 1189–1194. doi:10.1109/IROS.2014.6942708.

[10] Dutta, R., Sun, L., and Pack, D., "Multi-agent Formation Control with Maintaining and Controlling Network Connectivity," *American Control Conference*, Boston, MA, USA,, 2016. doi:10.1109/IROS.2014.6942708.

[11] Farmani, N., Sun, L., and Pack, D., "Tracking multiple mobile targets using cooperative Unmanned Aerial Vehicles," *Unmanned Aircraft Systems (ICUAS), 2015 International Conference on*, 2015, pp. 395–400. doi:10.1109/ICUAS.2015.7152315.

[12] Wang, X., Yadav, V., and Balakrishnan, S. N., "Cooperative UAV Formation Flying With Obstacle/Collision Avoidance," *IEEE Transactions on Control Systems Technology*, Vol. 15, No. 4, 2007, pp. 672–679. doi:10.1109/TCST.2007.899191.

[13] Xuan-Mung, N., and Hong, S. K., "Robust adaptive formation control of quadcopters based on a leader–follower approach," *International Journal of Advanced Robotic Systems*, Vol. 16, No. 4, 2019, p. 1729881419862733.

[14] Sun, L., and Hu, B., "Event-triggering in three-dimensional leader-follower formation control for unmanned aerial vehicles," *ASME 2016 Dynamic Systems and Control Conference*, American Society of Mechanical Engineers Digital Collection, 2016.

[15] Al-Radaideh, A., Selje II, R., and Sun, L., "Relative Dynamics Modeling and Three-Dimensional Formation Control for Leader-Follower UAVs in the Presence of Wind," *AIAA Scitech 2020 Forum*, 2020, p. 0878.

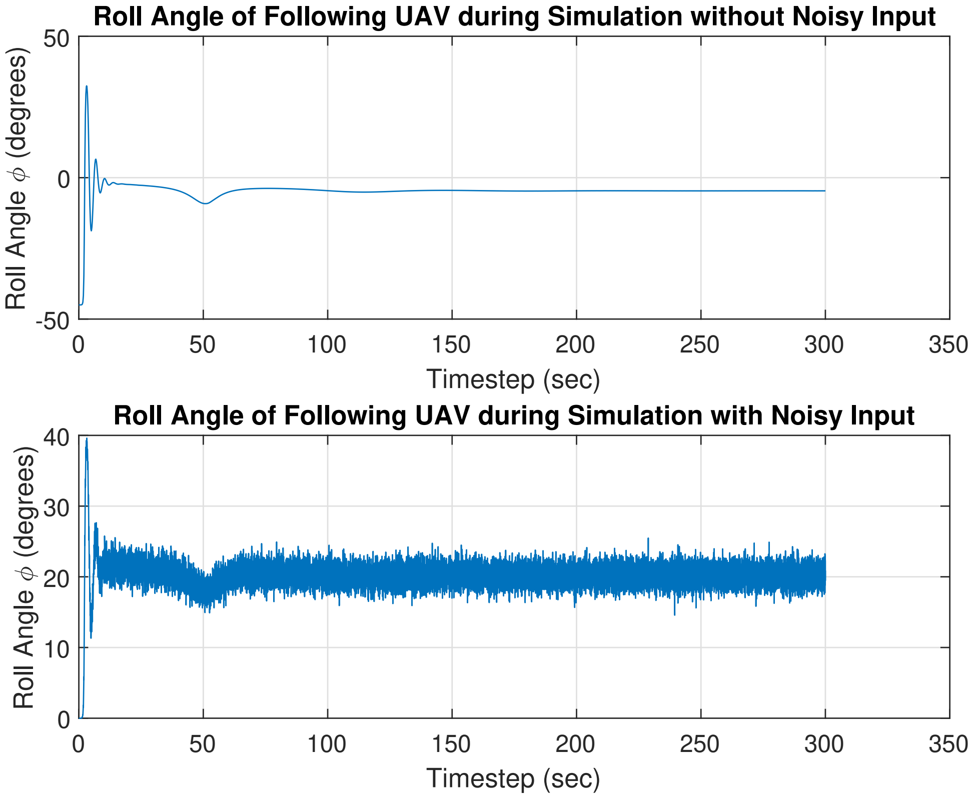
[16] Sun, L., Hu, B., and Zhao, S., "An event-triggering-based approach for three-dimensional local-level frame formation control of leader-follower UAVs," *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, IEEE, 2017, pp. 472–479.

[17] Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., and Riedmiller, M., "Deterministic policy gradient algorithms," *In International conference on machine learning*, PMLR, 2014, pp. 387–395.

[18] Tangkaratt, V., Abdolmaleki, A., and Sugiyama, M., "Guide actor-critic for continuous control," *arXiv preprint*, , No. arXiv:1705.07606, 2017.

[19] Li, S., Wu, Y., Cui, X., Dong, H., Fang, F., and Russell, S., "Robust multi-agent reinforcement learning via minimax deep deterministic policy gradient," *In Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, pp. 4213–4220.

[20] Beard, R. W., and McLain, T. W., *Small Unmanned Aircraft: Theory and Practice*, Princeton University Press, 2011.


Roll Angle of Following UAV during Simulation without Noisy Input

Roll Angle of Following UAV during Simulation with Noisy Input

Fig. 8 The velocity and roll angle of follower UAV generated by the learned RL policy.