Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Deep Reinforcement Learning Framework for Multi-UAV Collision-Free Navigation and Cooperative Control

Jailsingh Bhookya

jailsingh@nitc.ac.in

National Institute of Technology Calicut

Abhishek Mondal

National Institute of Technology Calicut

Suvam Das

National Institute of Technology Calicut

Research Article

Keywords: Unmanned Aerial Vehicle (UAV), Deep Reinforcement Learning(DRL), Proximal Policy Optimization(PPO), 2D/3D-Environment, Model based control

Posted Date: August 20th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-8248621/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Deep Reinforcement Learning Framework for Multi-UAV Collision-Free Navigation and Cooperative Control

Jailsingh Bhookya[1*], Abhishek Mondal[1] and Suvam Das[1]

[1*]Department of Electronics and Communication Engineering, National Institute of Technology Calicut, , Kozhikode, 673601, Kerala, India.

*Corresponding author(s). E-mail(s): jailsingh@nitc.ac.in;
Contributing authors: abhishekmondal@nitc.ac.in;
suvam_m230919ec@nitc.ac.in;

Abstract

Unmanned Aerial Vehicles (UAVs) have emerged as vital components in applications such as urban mobility, surveillance, and logistics. However, achieving safe and efficient navigation in cluttered and dynamic environments remains a significant challenge. This study proposes a Deep Reinforcement Learning (DRL) framework based on the Proximal Policy Optimization (PPO) algorithm to enable multiple UAVs to autonomously navigate complex urban environments while avoiding collisions and reaching designated targets. Unlike traditional model-based control approaches, the proposed PPO-driven system learns optimal navigation strategies through continuous interaction with the environment. It dynamically adapts to both static and moving obstacles—including other UAVs—ensuring responsive and collision-free operation in real time.

Comprehensive simulations conducted in both 2D and 3D urban environments demonstrate that the trained agents achieve robust performance in single- and multi-UAV scenarios. The UAVs exhibit effective cooperation, generalize well to unseen environments, and maintain safe separation during flight. The results confirm that the DRL-based framework provides a scalable and intelligent solution for autonomous UAV navigation, offering a viable pathway toward real-world deployment in smart city and aerial traffic management systems.

Keywords: Unmanned Aerial Vehicle (UAV), Deep Reinforcement Learning(DRL), Proximal Policy Optimization(PPO), 2D/3D-Environment, Model based control

1 Introduction

Unmanned Aerial Vehicles (UAVs), also known as drones, have evolved from military tools into vital assets in civilian sectors, including logistics, agriculture, disaster response, and surveillance [1, 2]. The rapid expansion of UAV applications in urban mobility, surveillance, and logistics has underscored the need for robust and intelligent navigation systems that can operate in dynamic, obstacle-rich environments. My academic background in electronics design and technology, combined with a strong interest in autonomous systems and control algorithms, addresses the challenges of UAV navigation and coordination in complex scenarios. Advances in sensors, embedded systems, and communication technologies have made UAVs more capable and accessible [3]. With the growing deployment of UAV in complex environments, such as urban landscapes or disaster zones, autonomy has become a critical focus in UAV research [4].

Autonomous navigation requires UAVs to plan and follow safe trajectories while avoiding collisions in dynamic, obstacle-rich environments, as illustrated in Fig. 1. Multi-UAV systems offer scalability and fault tolerance, enabling missions to be distributed across multiple agents [5]. However, they also introduce challenges such as real-time coordination, collision avoidance, and communication under uncertainty [6].

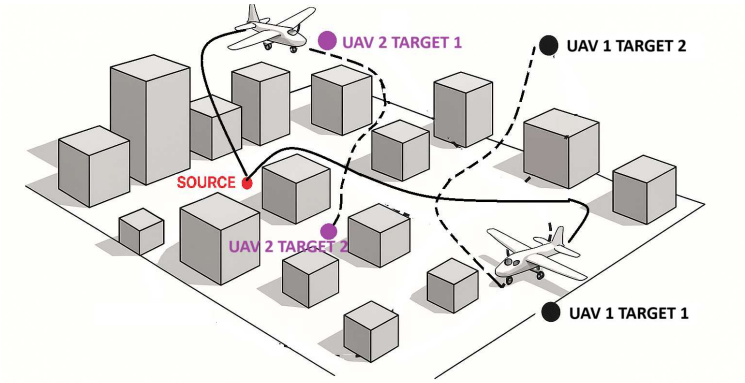


Fig. 1 Illustration of UAVs Navigation in 3D Environment

Traditional control strategies, such as Model Predictive Control (MPC), have shown strong results in structured settings but struggle with adaptability, real-time computation, and scalability in dynamic or multi-agent scenarios [5, 6]. Recent literature has highlighted the potential of Deep Reinforcement Learning (DRL) as a model-free solution that learns directly from environmental feedback [7, 8], as shown in Fig. 2. Algorithms such as Proximal Policy Optimization (PPO) have demonstrated success in enabling robust and adaptive UAV navigation [9, 10]. For example, Qi et al. [9] showed how PPO can improve UAV path planning, and Pan et al. [10] applied PPO for tracking moving targets in cluttered environments.

The evolution of autonomous navigation for Unmanned Aerial Vehicles (UAVs) has been driven by the increasing demand for robust and adaptable systems that can

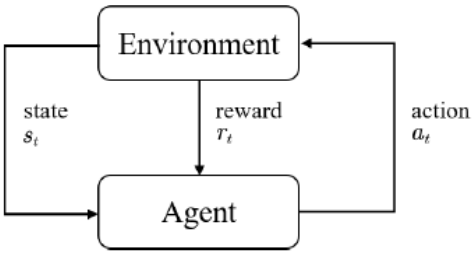


Fig. 2 Reinforcement Learning

operate in complex, dynamic environments. Early research in UAV navigation relied heavily on traditional control methods such as Model Predictive Control (MPC), which excels in structured, static settings due to its ability to handle system constraints and predict future states [1] [5]. However, as highlighted in my own prior work and by researchers such as Saccani et al. [5], MPC-based approaches encounter significant limitations in dynamic, obstacle-rich, or multi-agent scenarios, struggling with real-time adaptability, high computational overhead, and a dependency on accurate environmental models. My personal experience with MPC-based UAV path following confirmed these challenges: while MPC performed well in structured environments and at lower speeds, its effectiveness diminished as UAV speed increased and when obstacles were introduced, particularly when multiple UAVs needed to coordinate in shared airspace. These limitations motivated the exploration of more flexible, learning-based control strategies.

The objective is to develop and evaluate a deep reinforcement learning framework based on Proximal Policy Optimization (PPO) that enables multiple UAVs to autonomously and efficiently navigate complex 2D and 3D environments, avoid collisions, and coordinate in real-time for safe and scalable urban operations. This work builds on these advances by exploring PPO-based DRL to enable multiple UAVs to learn safe, coordinated navigation in complex 2D and 3D environments.

2 RELATED WORK

Several previous studies have explored user association strategies in UAV-assisted networks, emphasizing the improvement of connectivity and the avoidance of collisions. For example, the authors in [11] proposed a mechanism that allows UAV to dynamically associate with ground user equipment (UEs) and adjust their flight coordinates to maximize downlink (DL) throughput in multi-UAV scenarios. Similarly, [12] addressed a joint optimization problem involving both UE–UAV association and UAV positioning to enhance communication efficiency. However, the methods proposed in these studies do not fully account for the interference and collision avoidance challenges that arise in full-duplex (FD) multi-UAV environments. In such settings, UAVs must coordinate their flight trajectories and communication links to prevent aerial collisions and minimize mutual interference. Therefore, it becomes essential to develop efficient algorithms that jointly optimize decoupled UL/DL associations, UAV trajectory design, and collision avoidance mechanisms in FD multi-UAV networks.

Given the high maneuverability of UAVs, the communication distance between UAVs and UEs can be kept sufficiently short to enable reliable in-band FD communication. Accordingly, both UEs and BSs (including macro base stations and UAVs) are assumed to operate in FD mode for efficient spectrum reuse [13]. For instance, as illustrated in Fig. 3, a UE and its associated UAV can perform simultaneous UL and DL transmissions on distinct frequency channels, while UAV trajectories are dynamically adjusted to maintain safe separation distances and prevent mid-air collisions.

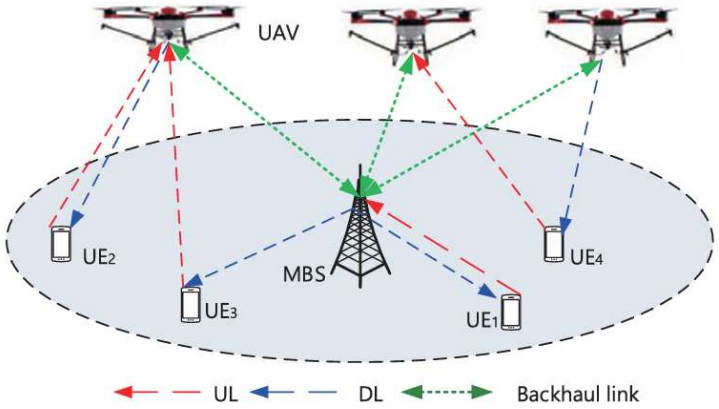


Fig. 3 A graphical illustration for DL-UL decoupling association in an FD wireless network consisting of one MBS and multiple UAVs.

Recent advances in reinforcement learning (RL) and, more specifically, deep reinforcement learning (DRL), have opened new avenues for autonomous UAV navigation [6][7][8]. Unlike MPC, DRL does not require explicit models of the environment; instead, UAVs learn optimal navigation strategies through trial and error, guided by carefully designed reward functions. This model-free approach is especially attractive for multi-UAV systems, where coordination and collision avoidance are paramount. Several studies have demonstrated the effectiveness of DRL in UAV path planning and multi-agent coordination. For instance, Chen and Hossain [6] showed that multi-agent DRL can manage joint user association and trajectory design in UAV networks, enabling real-time decision-making even as environmental conditions change. Peng et al. In [7], the exploration of DRL-based path planning for UAV-assisted edge computing highlighted the method's adaptability and robustness in dynamic settings. Zheng et al. [8] demonstrated that multi-layer reinforcement learning can improve scalability and adaptability compared to traditional methods. Other researchers, such as Gu et al. [2], have also contributed by proposing stepwise DQN algorithms that enhance UAV path planning in complex, crowded environments, further illustrating the versatility of DRL in addressing the challenges of modern UAV operations.

Within the DRL landscape, the Proximal Policy Optimization (PPO) algorithm has emerged as a particularly effective choice for UAV navigation tasks [9][10][14].

PPO is known for its training stability and efficiency, especially in continuous control problems, such as UAV flight. Recent works, such as those by Qi et al. In [9] have further improved PPO for UAV path planning, achieving better convergence and robustness in complex environments. Additionally, Pan and Wang [10] developed a PPO-based maneuvering target tracking method for UAVs in urban environments, demonstrating the algorithm's ability to handle dynamic targets and obstacles. Fernando et al. [14] proposed a scalable, decentralized RL framework using recurrent PPO for UAV target localization, showcasing the potential for real-world deployment in perceptually degraded environments. The advantages of DRL over MPC are numerous: DRL's adaptability, scalability, and ability to support autonomous learning make it well-suited for modern UAV navigation and coordination tasks, especially in scenarios involving multiple UAVs and complex environments [1][4][15].

Despite these advances, DRL-based approaches are not without their challenges. Training can be computationally intensive, and the performance of learned policies is highly sensitive to the design of the reward function [16][17]. Sample inefficiency and the need for careful hyperparameter tuning can also pose practical hurdles [18][19]. However, ongoing research continues to address these issues, improving the robustness and real-world applicability of DRL for autonomous UAV systems. For example, recent work by Zhang et al. [4] proposed an efficient RL-based cooperative navigation algorithm that addresses sparse reward problems and enhances training efficiency through staged-tuning and prioritized experience replay strategies. Similarly, Zeng et al. [3] introduced coverage-aware navigation algorithms leveraging DRL and radio mapping, demonstrating improved learning efficiency and adaptability in urban environments. These innovations highlight the dynamic nature of the field and the ongoing efforts to bridge the gap between theoretical advancements and practical deployment.

The work builds on these developments by focusing on the practical implementation and evaluation of PPO-based DRL for multi-UAV systems in challenging environments. By leveraging PPO, I aim to enable UAVs to autonomously navigate, avoid collisions, and coordinate in shared 3D environments, thereby overcoming the limitations encountered with MPC. The results from simulation experiments demonstrate that, after sufficient training, UAVs can dynamically adapt their trajectories, avoid obstacles, and efficiently reach their targets, even in complex urban-like settings with multiple static and dynamic obstacles [1][4]. These findings are supported by the broader literature, which consistently demonstrates the advantages of DRL over traditional control methods in scenarios requiring real-time adaptability, multi-agent coordination, and robust collision avoidance [6].

3 Motivation

The increasing use of UAVs in urban mobility, surveillance, and logistics demands intelligent navigation systems capable of real-time collision avoidance and coordination in dynamic, obstacle-rich environments. Traditional approaches, such as Model Predictive Control (MPC), struggle with adaptability and scalability in such settings [1, 4].

To address these challenges, this work employs Deep Reinforcement Learning (DRL) with Proximal Policy Optimization (PPO), which is chosen for its stability and effectiveness in continuous multi-agent scenarios. Recent studies have demonstrated PPO's superior performance in UAV path planning, target tracking, and energy-efficient navigation compared to conventional methods [2–4].

To overcome these limitations, I turned to Deep Reinforcement Learning (DRL), which offers a model-free, learning-based framework that enables UAVs to autonomously learn optimal navigation strategies through continuous interaction with their environment. Among the DRL algorithms, Proximal Policy Optimization (PPO) was selected for its training stability, efficient handling of continuous action spaces, and practical balance between performance and complexity. Recent research by Qi et al. demonstrated the effectiveness of PPO-based algorithms for UAV path planning in complex environments, with improved convergence and robustness compared to traditional methods[2]. Similarly, Pan and Wang showed that PPO-based approaches can achieve high success rates in maneuvering, target tracking, and obstacle avoidance in urban environments[3]. Chikhaoui et al. further highlighted the advantages of PPO-based DRL in energy-efficient navigation and collision avoidance in 3D urban spaces[4].

The primary contribution of this project is the development and evaluation of a multi-agent DRL framework based on PPO, specifically tailored for trajectory optimization in obstacle-rich, dynamic 3D environments. The framework enables real-time collision avoidance, coordinated multi-UAV navigation, and energy-efficient path planning, addressing the limitations of centralized MPC approaches. By integrating advanced reward shaping, network architectures, and experience replay strategies, the proposed system achieves robust generalization and scalability. This work bridges the gap between traditional control methods and modern DRL, demonstrating the practical effectiveness of PPO-based DRL for multi-UAV systems. The novelty lies in the seamless integration of continuous control, multi-agent coordination, and dynamic obstacle handling within a unified learning framework, paving the way for safer, more intelligent, and scalable autonomous aerial operations in urban landscapes. This work bridges the gap between traditional control and modern DRL, showcasing the practical effectiveness of PPO-based approaches for safe, scalable UAV operations in complex urban scenarios.

4 Methodology

The work implements a methodology adopted to address the problem of UAV trajectory optimization, with an emphasis on collision avoidance and real-time adaptability. The proposed solution leverages a *Deep Reinforcement Learning (DRL)* framework using the *Proximal Policy Optimization (PPO)* algorithm. PPO is selected for its balance between training stability and policy improvement efficiency.

4.1 Proximal Policy Optimization (PPO)

Proximal Policy Optimization (PPO) is a reinforcement learning algorithm that trains an agent to make decisions by updating its policy. It is a policy gradient method that

improves stability by limiting the size of policy updates using a "clipped" objective function, which prevents the new policy from deviating too drastically from the old one. This makes the learning process more reliable, simpler to implement, and more sample-efficient than standard policy gradient methods; the architecture of the PPO is presented in Fig. 4.

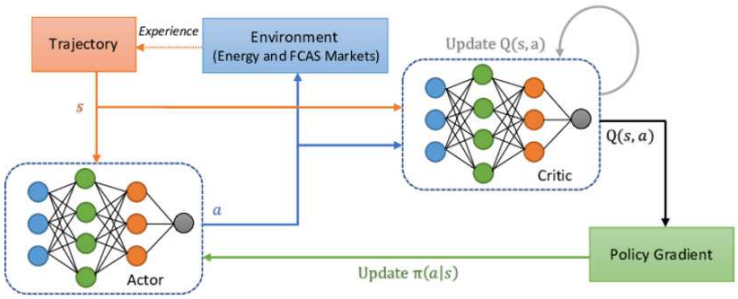


Fig. 4 Proximal Policy Optimization (PPO) architecture

The core algorithm utilized in this work is PPO, a popular and efficient DRL algorithm. The PPO addresses the instability issues associated with earlier policy gradient methods by incorporating a clipping mechanism that restricts the magnitude of policy updates. This approach prevents sudden and drastic changes in the learned policy, ensuring smoother and more stable convergence toward an optimal solution.

DRL, especially Proximal Policy Optimization (PPO), allows agents to learn optimal policies directly from their interactions with the environment, making it well-suited for navigation and collision avoidance [8].

PPO is particularly suitable for tasks involving navigation and obstacle avoidance, as it allows agents (in this case, UAVs) to learn directly through interaction with the environment. By continually exploring and exploiting the state-action space, the agent gradually improves its decision-making strategy. The learning framework is based on the actor-critic method, where the actor network is responsible for determining the actions to take, while the critic network evaluates the outcomes of these actions by estimating a value function. The continuous interplay between these networks refines the agent's policy iteratively.

The UAV is modeled as an autonomous agent navigating a two-dimensional environment filled with obstacles. The goal is to reach a specified target location from a predefined source while minimizing collision risk. The reinforcement learning environment is defined by three key components: the state space, the action space, and the reward function.

4.1.1 State Space

The state of the UAV is defined by its normalized current position and its relative position with respect to the target. Specifically, the state vector includes the difference between the UAV's coordinates and the target's coordinates, as well as the UAV's own

normalized position. The normalization process scales the values to a range between -1 and 1, which stabilizes the learning process by ensuring that the neural network input remains within a manageable range. This normalized state vector is expressed as:

$$s = \left[\frac{p_x - p_{\text{target},x}}{50}, \frac{p_y - p_{\text{target},y}}{50}, \frac{p_x}{50}, \frac{p_y}{50} \right]$$

where:
p_x, p_y: Current position of the UAV.
$p_{\text{target},x}, p_{\text{target},y}$: Target position coordinates.
The normalization ensures that all inputs are bounded between -1 and 1. [1][20]

4.1.2 Action Space

The actions taken by the UAV consist of displacements along the x-axis and y-axis. These displacements are bounded to ensure feasible and safe movements within the environment. The actor network outputs the parameters of a Gaussian distribution (mean and standard deviation), from which the actual movement vector is sampled. This probabilistic action selection enables the UAV to explore various strategies during training, which is crucial for learning robust navigation behaviors [4, 21].

4.1.3 Reward Function

The reward function is designed to incentivize progress toward the target while penalizing collisions and aimless wandering. The primary component of the reward is the negative Euclidean distance between the UAV and the target, which encourages the agent to reduce this distance. A bonus reward is provided when the UAV reaches close proximity to the target (within a defined threshold), and a large penalty is imposed for any collision with obstacles. The reward function is mathematically formulated as:

$$R = -d + 30 \cdot \mathbb{I}(d < 1.5) - 200 \cdot \mathbb{I}(\text{collision})$$

where d is the Euclidean distance between the UAV and the target, and $\mathbb{I}$ denotes an indicator function that returns 1 if the condition is true and 0 otherwise [4, 22].

4.1.4 Distance Calculation

The distance to the target is calculated using the Euclidean norm, given by:

$$d = \|\mathbf{p}_{\text{current}} - \mathbf{p}_{\text{target}}\|$$

where:
$\mathbf{p}_{\text{current}}$: UAV's current position.
$\mathbf{p}_{\text{target}}$: Target position.
$\|\cdot\|$: Euclidean norm.

4.1.5 Algorithmic workflow

Algorithm 1: PPO-Based Multi-UAV Navigation with 3D Obstacles

Input: Number of episodes N_{episode}, maximum steps N_{step}, source position s_{src}, target positions $\{\mathcal{T}_i\}$ for each UAV, environment boundaries and building definitions

Output: Trained actor and critic networks, reward history, UAV trajectories, and return paths

```
Initialize actor and critic network parameters;
Initialize PPO agent and memory buffer;
for each episode do
    for each UAV do
        Set initial position and initialize trajectory;
        for each target do
            for each step do
                Construct state vector;
                Sample action and update position;
                Handle collisions and target reaching;
                Compute and store reward;
                Append position to trajectory;
                if target reached then
                    break
        Return to source and store trajectory;
    Update PPO agent using collected experiences;
    Store episode reward and print progress periodically;
```

4.2 Actor-Critic Framework

The actor-critic architecture is central to the PPO algorithm. The actor network outputs the parameters of a probability distribution (typically Gaussian) over the action space. From this distribution, the UAV samples its next move. This probabilistic action selection mechanism promotes sufficient exploration during the early training phases[20].

The critic network, on the other hand, estimates the value function, which predicts the expected return from a given state. By comparing the actual returns to these estimates, the algorithm determines whether the actions taken were better or worse than expected [1, 4].

4.2.1 Policy Output (Actor Network)

The policy (actor network) must output actions for the agent to explore the environment. Instead of deterministic actions, we use a probabilistic approach to allow for exploration during training. This helps the agent learn better in dynamic and uncertain environments.

The actor network outputs a Gaussian policy for sampling actions:

$$a \sim \mathcal{N}(\mu(s), \sigma(s))$$

where:

$\mu(s)$: Mean action predicted by the actor network.

$\sigma(s)$: Standard deviation that controls exploration by the spread of possible actions predicted by the actor network.

a: The sampled action from the policy.

4.2.2 Critic Value Estimation

The critic network estimates the value of a state s , representing its expected return, given by:

$$V(s) = f_{\text{critic}}(s)$$

where f_{critic} is the function implemented by the critic neural network.

4.3 Discounted Return (G)

To account for future rewards, the algorithm computes a discounted return at each time step. This is given by the recursive formula:

$$G_t = r_t + \gamma G_{t+1}$$

Where G_t is the return at the time step t, r_t is the reward received, and γ is the discount factor, typically between 0.9 and 0.99, which controls the importance of future rewards [1, 4].

4.4 Training Process

4.4.1 Environment Simulation

The simulated environment includes a 2D/3D space with randomized obstacles and targets. Each episode begins with the UAV at a fixed source location, and it must reach the target location while avoiding these obstacles. The dynamic nature of obstacle placement encourages generalization in the learned policy.

4.4.2 Action Sampling

At each time step, the actor network outputs the parameters of a Gaussian distribution. The UAV samples its next action from this distribution, introducing stochasticity and enabling exploration.

$$\mathbf{p}_{\text{new}} = \mathbf{p}_{\text{current}} + \mathbf{a}$$

4.4.3 Advantage Estimation

The advantage function measures how much better an action is performed compared to the expected baseline. It is computed as:

$$A_t = G_t - V(s_t)$$

A positive advantage indicates that the action was beneficial and should be reinforced, whereas a negative value suggests that the action was suboptimal, as shown in Fig. 5.

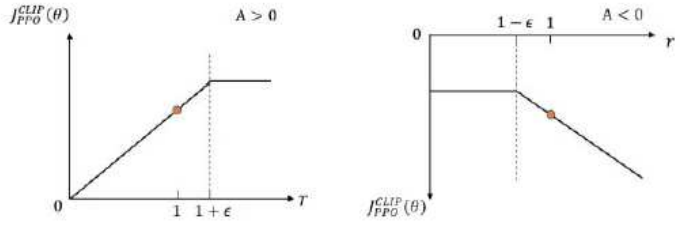


Fig. 5 restricted range of objective function

where:
A_t: Advantage at time step t.
G_t: Discounted return/reward at time step t.
$V(s_t)$: Value of the state s_t estimated by the critic.
If A > 0, the action was better than expected, and the policy should be reinforced.
If A < 0, the action was worse than expected, and the policy should be suppressed.

4.4.4 Gradient Clipping

To prevent large updates during backpropagation, gradient clipping is employed. When the computed gradient exceeds a defined threshold, it is clipped to remain within bounds[9, 23]:

$$\text{grad}_{\text{clipped}} = \text{clip}(\text{grad}, -\text{max_norm}, \text{max_norm})$$

This technique stabilizes training, especially in environments with sparse or erratic rewards.

4.4.5 Loss Functions

The actor loss is formulated to encourage advantageous actions while penalizing abrupt policy updates. PPO uses a clipped objective function:

$$L^{\text{actor}} = -\mathbb{E}\left[\min\left(r_t A_t, \text{clip}(r_t, 1-\epsilon, 1+\epsilon)A_t\right)\right]$$

where:
$r_t = \frac{\pi(a_t|s_t)}{\pi_{\text{old}}(a_t|s_t)}$ is the probability ratio of the new policy to the old policy, and ϵ is a small constant (typically 0.2) that defines the clipping range. Additionally, A_t is the advantage at time step t.

The critical loss is defined as the mean squared error between the predicted and actual returns. The critic network estimates the value of each state. The critic's objective is to minimize the error between the predicted value and the actual discounted reward [24, 25].

$$L^{\text{critic}} = \frac{1}{N} \sum_{t=1}^{N} (V(s_t) - G_t)^2$$

This loss drives the critic to improve its predictions, thereby enhancing the accuracy of advantage estimation.

where

N: Number of time steps in the episode.

$V(s_t)$: Value of state s_t.

G_t: Discounted return at time t.

4.4.6 Gradient Clipping

To ensure stability during training, gradients are clipped to prevent excessively large updates:

$$\nabla_\theta \leftarrow \text{clip}(\nabla_\theta, -\delta, \delta) \tag{1}$$

4.4.7 Policy Updates

Here, both the actor and critic networks are updated using their respective loss functions via the Adam optimizer. This optimization process is carried out iteratively over multiple episodes.

4.4.8 Performance Evaluation

The effectiveness of the trained policy is evaluated over multiple episodes by analyzing key metrics such as reward convergence, the efficiency of trajectories, and the rate of successful obstacle avoidance. Visualization of the UAV's path is used to assess whether the UAV is able to consistently navigate around obstacles and reach the target location.

4.5 UAV Dynamics Integration

In the earlier version, UAVs were controlled using direct positional changes in the form of $[\Delta x, \Delta y, \Delta z]$. However, although this approach was effective for basic path planning and collision avoidance, it did not accurately reflect how real UAVs move in physical space. To bridge this gap and make the simulation more realistic, we extended our model to include basic UAV flight dynamics using four essential control inputs: thrust, pitch, roll, and yaw rate. The UAV's motion is influenced by both its linear and angular control inputs. In the dynamic model, it can be considered as follows:

- **Thrust (T)**: Controls vertical lift or descent.
- **Pitch (θ)**: Tilts the UAV forward or backward, affecting motion along the x-axis.
- **Roll (ϕ)**: Tilts the UAV sideways, influencing motion along the y-axis.
- **Yaw rate ($\dot{\psi}$)**: Changes the heading angle, affecting the direction.

- **Control Parameters:** Thrust (T), pitch (θ), roll (ϕ), and yaw rate ($\dot{\psi}$).
- **Kinematic Model:**

$$a_x = T\sin\theta, \quad a_y = T\sin\phi$$
$$a_z = T\cos\theta\cos\phi - g$$
$$\mathbf{v} \leftarrow \mathbf{v} + \mathbf{a}\Delta t$$
$$\mathbf{p} \leftarrow \mathbf{p} + \mathbf{v}\Delta t$$

  where g is gravitational acceleration.

The state of the UAV is represented as:

$$s = [x, y, z, v_x, v_y, v_z, \psi]$$

where (x, y, z) is the current position, (v_x, v_y, v_z) are the velocities along each axis, and ψ is the yaw (heading) angle.

Given an action $a = [T, \theta, \phi, \dot{\psi}]$, the UAV's position and velocity are updated using the following simplified equations:

$$\begin{aligned}
\psi_{t+1} &= \psi_t + \dot{\psi} \cdot \Delta t \\
a_x &= T \cdot \sin(\theta) \\
a_y &= T \cdot \sin(\phi) \\
a_z &= T \cdot \cos(\theta) \cdot \cos(\phi) - g \\
v_x &= v_x + a_x \cdot \Delta t \\
v_y &= v_y + a_y \cdot \Delta t \\
v_z &= v_z + a_z \cdot \Delta t \\
x &= x + v_x \cdot \Delta t \\
y &= y + v_y \cdot \Delta t \\
z &= z + v_z \cdot \Delta t
\end{aligned}$$

Here, g is the gravitational constant (9.81 m/s^2) and Δt is the simulation time step.

5 Results and Discussions

5.1 UAV Collision Avoidance in 2-D Environment

A single UAV was tasked with navigating from the source [0, 0] to the target [50, 50] in a 2D environment with ten rectangular obstacles, as listed in Table I. Using PPO, the UAV successfully avoided all obstacles and reached the target, which is also listed in Table II, as shown in Fig. 6. The reward convergence plot in Fig. 7 demonstrates effective learning.

5.2 Multi-UAV Collision Avoidance in 3-D Environment

The following diagram illustrates a 3D environment containing seven buildings with varying positions, heights, and widths, along with a base station and four target points. Two UAVs are required to navigate from the source to their assigned targets. Each

Table 1 Buildings positions and width

[(x,y positions) , (width)]
[(2,42) , (4,5)]
[(5,13) , (3,3)]
[(7,14) , (3,3)]
[(17,45) , (4,4)]
[(22,38) , (2,2)]
[(20,12), (3,3)]
[(25,27), (2,2)]
[(29,7), (2,2)]
[(32,0), (5,5)]
[(40,10), (2,2)]

Table 2 Target Positions

	Source [x,y,z]	Target [x,y,z]
UAV	[0,0]	[50,50]

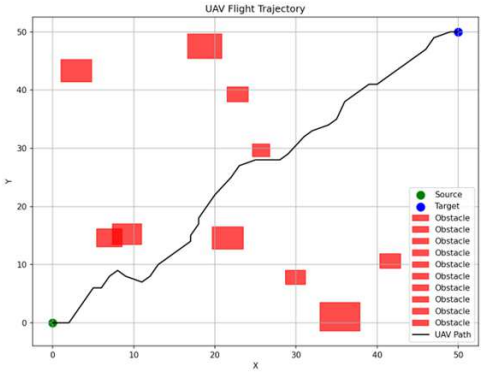


Fig. 6 UAV Trajectory of 2D Environment

UAV is assigned two targets and will travel through the city, first reaching its first target and then proceeding to the second, while avoiding the buildings. This navigation is achieved using the Proximal Policy Optimization (PPO) algorithm, a deep reinforcement learning technique.

In this, the stars represent the first targets, and the cross-signs represent the second targets for both UAVs. The red-colored symbols indicate the first target (star) and the second target (cross) for UAV1, while the purple-colored symbols indicate the first target (star) and the second target (cross) for UAV2.

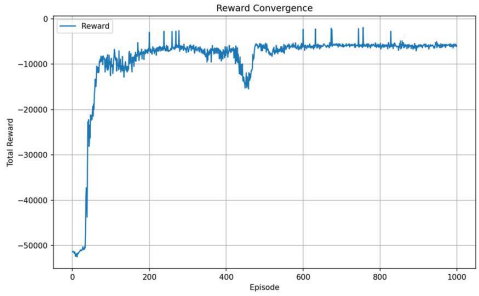


Fig. 7 Reward Convergence Plot of 2D Environment

5.2.1 Simulation Environment

Two UAVs began their flight from the source position [25, 25, 5], avoiding the buildings considered obstacles. In the trajectory diagram shown in Fig. 8 & Fig. 9:

1. The blue-colored trajectory represents the flight path of UAV1, which first reached its initial target at [15, 10, 25] and proceeded to the next target at [45, 42, 20].
2. The yellow-colored trajectory represents the flight path of UAV2, which first reached its initial target at [45, 5, 15] and then traveled to[5, 45, 20].

- The positions, widths, and heights of the buildings are listed in Table III.

Table 3 Building positions, width, and height

[(x,y positions) , (width) , height]
[(5,15) , (4,3) , 22]
[(10,40) , (3,3) , 20)]
[(15,30) , (3,3) , 30]
[(20,5) , (4,4) , 25]
[(25,35) , (2,2) , 20]
[(30,15), (2,2) , 20]
[(35,40), (3,3) , 35]

- Target positions are listed in Table IV.

Table 4 Target Positions

	Target 1 [x,y,z]	Target 2 [x,y,z]
UAV 1	[15, 10, 25]	[45, 42, 20]
UAV 2	[45, 5, 15]	[5, 45, 20]

The Multi-UAV Reward Convergence Plot for the 3D Environment 1 (Env1) in Fig. 10 illustrates the learning stability and performance improvement of the reinforcement learning (RL)-based collision avoidance and trajectory optimization algorithm during

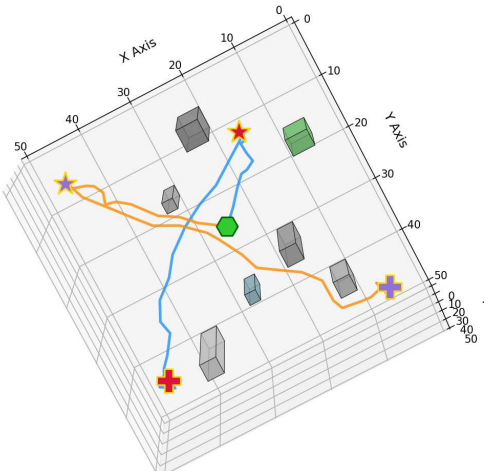


Fig. 8 Top view of Multi-UAV Trajectory of 3D Env1

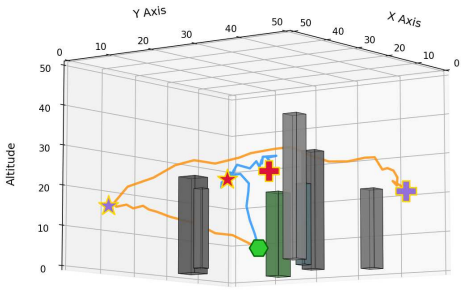


Fig. 9 Side View of Multi-UAV Trajectory of 3D Environment

the training process. The reward curve represents the cumulative reward obtained by multiple UAV agents over successive training episodes within the three-dimensional environment. As training progresses, the average reward gradually increases and the variance decreases, demonstrating that the agents are converging toward optimal cooperative behaviors. This convergence implies that the UAVs have effectively learned to balance multiple objectives, including collision avoidance, maintaining communication links, and planning energy-efficient paths.

5.3 Multi-UAV Multi-Target Tracking and Collision Avoidance

In this scenario, two UAVs are deployed in a realistic 3D urban environment to track and intercept two separate moving targets. The environment comprises seven buildings situated at various locations with differing heights and widths, adding complexity and creating several potential obstacles for the UAVs to navigate. A base station is also present, and each UAV is assigned its own target to follow. Both UAVs start from the

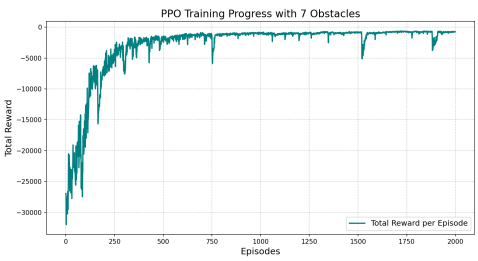


Fig. 10 Multi-UAV Reward Convergence Plot of 3D Env1

same fixed source location, while the two targets begin their movement from different positions. Each target moves along its own unique and non-linear path through the city-like environment, reflecting the challenges of real-world target tracking, where movement can be unpredictable and dynamic.

To handle this, we use the Proximal Policy Optimization (PPO) algorithm—a powerful deep reinforcement learning method. PPO enables each UAV to learn how to make informed decisions based on its observations, such as its own location, the location of its assigned target, and the surrounding buildings. Over many training episodes, the UAVs learn to navigate efficiently, track their targets, and avoid obstacles, including each other.

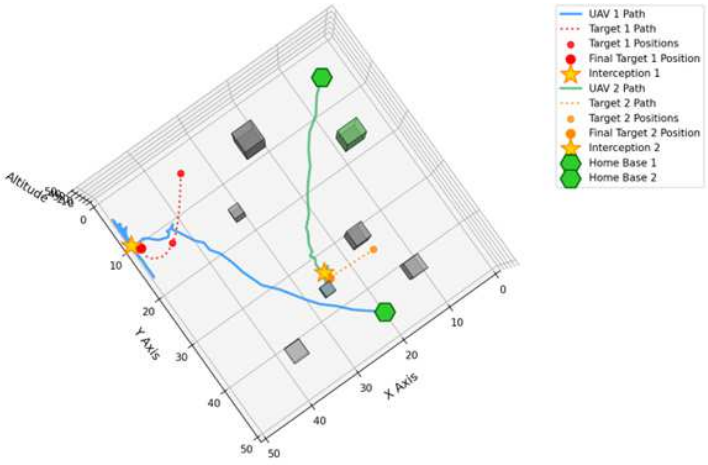


Fig. 11 UAV Trajectory of 3D Environment

The simulation results are shown in Fig. 11, Fig. 12, & Fig. 13 through trajectory plots:

1. The blue and green lines represent the flight paths of the two UAVs as they follow their assigned targets while avoiding collisions.
2. The red and orange lines show the paths taken by the two mobile targets as they move through the environment.

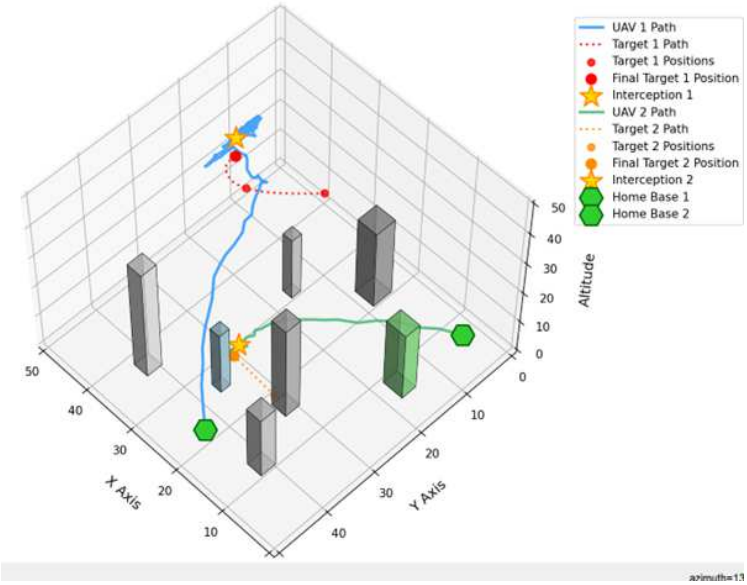


Fig. 12 UAV Trajectory of 3D Environment

3. The star markers indicate the interception points where each UAV successfully reaches its target. Once intercepted, both the UAV and its target stop moving, indicating the end of the mission.

This experiment demonstrates how multiple UAVs can operate simultaneously in a shared space, adapt to changing conditions, and effectively carry out their tasks using reinforcement learning. It highlights the potential of intelligent algorithms to handle complex navigation and coordination in real-world urban settings.

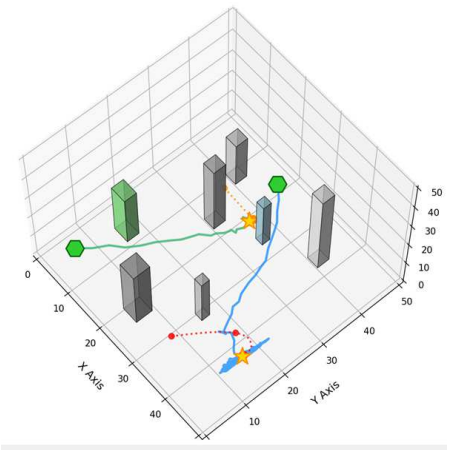


Fig. 13 UAV Trajectory of 3D Environment

In the later stages of training, the reward curve stabilizes around a high value, confirming the successful convergence of the learning algorithm shown in Fig. 14 and consistent coordination among the UAVs. The smoothness of the convergence trend also reflects the robustness of the reward design and the efficiency of the underlying policy update mechanism.

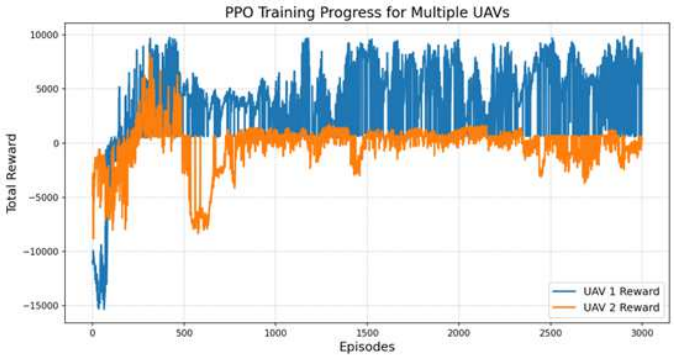


Fig. 14 UAV Trajectory convergence of 3D Environment

5.4 Obstacle Avoidance Performance

The Deep Reinforcement Learning (DRL) framework developed in this work enables UAVs to autonomously navigate complex urban environments, efficiently reaching targets while avoiding both static and dynamic obstacles. Although early training stages may involve occasional failures as the policy learns to generalize, convergence leads to consistently efficient, collision-free trajectories. The Proximal Policy Optimization (PPO) algorithm facilitates robust exploration and real-time adaptation, allowing UAVs to dynamically adjust their paths and maintain safe distances from obstacles and other UAVs. Randomized placement of obstacles and targets during training further enhances the policy's ability to generalize, resulting in high success rates for collision avoidance and energy-efficient, coordinated multi-UAV navigation.

6 Conclusion

This work demonstrates the effectiveness of deep reinforcement learning, specifically using the Proximal Policy Optimization (PPO) algorithm, for the autonomous navigation of multiple UAVs in complex urban environments. By moving beyond traditional control methods, such as Model Predictive Control, which are limited by computational intensity and poor adaptability in dynamic multi-agent scenarios, the proposed PPO-based framework enables UAVs to achieve robust collision avoidance, efficient path planning, and coordinated multi-agent navigation even in obstacle-rich and unpredictable settings. The results show that UAVs can autonomously adapt their trajectories, avoid both static and dynamic obstacles, and efficiently reach their targets,

validating the practical advantages of learning-based approaches over conventional methods. Recent research by Qi et al., Pan and Wang, and Chikhaoui et al. further supports these findings, highlighting PPO's superior convergence, robustness, and success rates for UAV navigation in urban environments.

Looking ahead, several promising directions for future development are evident. Hardware-in-the-loop validation will be essential to test the learned policies on real UAVs under physical constraints and real-world noise. Extending the framework to handle moving obstacles and scaling up to larger UAV swarms will further improve robustness and applicability. Integrating more sophisticated energy and time models, as well as exploring transfer learning and explainable decision-making techniques, will enhance the system's efficiency and transparency. With advancements in communication technologies, such as 5G and 6G, and the potential for incorporating human feedback, the future of intelligent UAV systems is poised for significant growth and real-world impact.

Declarations

Competing Interests The authors declare no competing interests.
Data Availability Statement The data that support the findings of this study are available from the corresponding author upon reasonable request. Requests for access may be considered on a case-by-case basis, provided the necessary conditions are met.
Code availability The code used in the experimental studies is available upon request to either authors.
Ethics approval Not Applicable.
Consent to participate Not Applicable.
Consent to publish Not Applicable.
Authors' contributions Jailsingh Bhookya provided continuous guidance/supervision throughout the research work concept and manuscript preparation. Abhishek Mondal is a co-supervisor in this work, and suggested complex environment of UAV. Suvam Das is a postgraduate student under the supervision of Jailsingh Bhookya and Abhishek Mondal. Both the supervisors are equally contributed to idea generation. All the authors read and approved the final manuscript.
Acknowledgements The authors acknowledge the department of Electronics and Communication Engineering at NIT Calicut for providing the lab infrastructure for carrying out the necessary experimental work presented in this paper.
Funding The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

References

[1] R. Bishop, "A survey of intelligent vehicle applications worldwide", IEEE intelligent Vehicles Symposium 2000

[2] Chen Da and Ekram Hossain , "Multi-Agent Deep Reinforcement Learning for Joint Decoupled User Association and Trajectory Design in Full-Duplex Multi-UAV Networks", IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL.

22, NO. 10, OCTOBER 2023

[3] Omid Esrafilian, David Gesbert "3D City Map Reconstruction from UAV-based Radio Measurements", Communication Systems Department, EURECOM, Sophia Antipolis, France

[4] Danilo Saccani, Leonardo Cecchin and Lorenzo Fagiano, "Multitrajectory Model Predictive Control for Safe UAV Navigation in an Unknown Environment," IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, VOL. 31, NO. 5, SEPTEMBER 2023

[5] ZHENGYANG CUI, YONG WANG "UAV Path Planning Based on Multi-Layer Reinforcement Learning Technique", IEEE Access 26, april 2021

[6] Gunasekaran Raja,Yashvandh Baskar,Priyanka Dhanasekaran, Raheel Nawaz,Keping Yu, "An Efficient Formation Control mechanism for Multi-UAV Navigation in Remote Surveillance",2021 IEEE Globecom Workshops (GC Wkshps)

[7] Yingsheng Peng, Yong Liu, and Han Zhang,"Deep Reinforcement Learning based Path Planning for UAV-assisted Edge Computing Networks",IEEE Wireless Communications And Networking Conference (WCNC) 2021

[8] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, Oleg Klimov, "Proximal Policy Optimization Algorithms", Machine Learning, arxiv, Cornell University

[9] Chenyang Qi, Chengfu Wu, Lei Lei, Xiaolu Li, Peiyan Cong ,"UAV path planning based on the improved PPO algorithm",2022 Asia Conference on Advanced Robotics, Automation, and Control Engineering (ARACE)

[10] Shuiping Pan,Qingling Wang,"Maneuvering target tracking of UAV in urban environment with PPO-MT" 2023 China Automation Congress (CAC)

[11] El Hammouti, H., Benjillali, M., Shihada, B., & Alouini, M. S. (2019). Learn-as-you-fly: A distributed algorithm for joint 3D placement and user association in multi-UAVs networks. IEEE Transactions on Wireless Communications, 18(12), 5831-5844.

[12] Xi, X., Cao, X., Yang, P., Chen, J., Quek, T., & Wu, D. (2019). Joint user association and UAV location optimization for UAV-aided communications. IEEE Wireless Communications Letters, 8(6), 1688-1691.

[13] Dai, C., Zhu, K., & Hossain, E. (2022). Multi-agent deep reinforcement learning for joint decoupled user association and trajectory design in full-duplex multi-UAV networks. IEEE transactions on mobile computing, 22(10), 6056-6070.

[14] Leon Fernando, Billy Lau Pik Lik, Chau Yuen, U-Xuan Tan,"A Scalable Decentralized Reinforcement Learning Framework for UAV Target Localization Using Recurrent PPO"TENCON 2024 - 2024 IEEE Region 10 Conference (TENCON)

[15] Xu Yue,Xu Yue,Yan Qiuzhen,Zhang Ying,"A method for high-precision direction finding of a satellite navigation interference source using the UAV" 2024 14th International Symposium on Antennas, Propagation and EM Theory (ISAPE)

[16] Stefani Kecman,Dinko Osmankovic,"Perspective Projection Approach to a Faster UAV Navigation and Obstacle Avoidance Learning"2023 XXIX International Conference on Information, Communication and Automation Technologies (ICAT)

[17] Fei Meng, Mingqiu Zhou, Jing Zhang, Shan Li,"Research on Autonomous Flight Technology of Battlefield UAV Based on Inertial/Vision Integrated Navigation"IEEE ITAIC

[18] Wenyuan Yang,"Research on Environment Perception and Path Planning Modeling of UAV Navigation Guided by Deep Learning"2024 IEEE 7th International Conference on Information Systems and Computer Aided Education (ICISCAE)

[19] Didula Dissanayaka ,Thumeera R. Wanasinghe , Oscar De Silva , Awantha Jayasiri, George K. I. Mann,"Review of Navigation Methods for UAV-Based Parcel Delivery"IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING

[20] Qijia Gu,Zhen An,Lanmin Chen,Kunfu Wang,"Path Planning of UAV Using Stepwise Deep Q-learning Algorithm"2022 8th Annual International Conference on Network and Information Systems for Computers (ICNISC)

[21] Parthiv Aneesh,Ravi Pratap Singh,Ratnakar Dash"Autonomous UAV Navigation in Indoor Corridor Environments by a Depth Estimator based on a Deep Neural Network"2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)

[22] Lijuan Zhang,Weiguo Yi,Hang Lin, Jiabin Peng,Pan Gao,"An Efficient Reinforcement Learning-Based Cooperative Navigation Algorithm for Multiple UAVs in Complex Environments"IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS

[23] Yun Chen, Nuria Gonz´alez-Prelcic and Robert W. Heath Jr."COLLISION-FREE UAV NAVIGATION WITH A MONOCULAR CAMERA USING DEEP REINFORCEMENT LEARNING"2020 IEEE INTERNATIONAL WORKSHOP ON MACHINE LEARNING FOR SIGNAL PROCESSING,

[24] Mohsen Eskandari, Andrey V. Savkin, "AI-based Navigation and Communication Control for a Team of UAVs with Reconfigurable Intelligent Surfaces Supporting

Mobile Internet of Vehicles" 2023 IEEE Conference on Control Technology and Applications (CCTA)

[25] Peng Sha,Qingling Wang "Autonomous Navigation of UAVs in Resource Limited Environment Using Deep Reinforcement Learning"2022 37th Youth Academic Annual Conference of Chinese Association of Automation (YAC)