# A Deep Reinforcement Learning-Based Adaptive Control Strategy for UAVs in Dynamic and Complex Environments

**Baraa M. Albaker**

Department of Electrical Engineering, Collage of Engineering, Al-Iraqia University, Baghdad, Iraq
Email: baraamalbaker@ymail.com

**Article History**

Received: Sep. 13, 2024

Revised: Dec. 24, 2024

Accepted: Jan. 19, 2025

## Abstract

The increasing usage of Unmanned Aerial Vehicles (UAVs) in diverse applications necessitates the development of effective flight control systems. A major difficulty, nevertheless, is maintaining robust flight control under complex and dynamic environmental conditions, such as obstacle and non-flying zones, wind disturbances, and sensor noise. Traditional control techniques fail to achieve required flight control in these environments. To address this point, an adaptive control strategy is proposed based on a Deep Reinforcement Learning (DRL) model to enhance the flight performance of quadcopters. The learning and adaptation of the DRL-based control strategy are implemented in real-time through continuous interaction with the environment. This is to improve flight control and achieve consistent UAV performance under varying conditions. Proximal policy optimization with a reward function is used to minimize positional errors, ensure collision-free flight paths, and reduce energy consumption. The developed DRL model for quadcopters is trained in a simulated environment and then tested in three complex environmental scenarios, including urban, forest and mountain terrains. Experimental results demonstrate remarkable improvements in UAV flight performance. In the training phase, the reported training reward increased from 10 to 110 and the train loss is dropped from 0.85 to 0.05, which indicated successful model learning. Also, during system verification, rewards increased from 12 to 115 and UAV flight path deviations were decreased from 0.5 to 0.08m. The proposed controller outperforms conventional approaches in urban environments by lowering average trajectory deviations to 0.2m from 0.35m for MPC and 0.6m for PID. Also, the developed DRL-based controller outperformed the PID and MPC controllers, with path deviations of 0.18m in mountains and 0.12m in forests. In addition, fewer collision rates with obstacles are achieved with the model, 3% in forest, 1.8% in urban areas, and 4.5% in mountains. Furthermore, the consumed energy is reduced to 950J as compared to 1200J for PID and 1050J for MPC. The results show the strength of deploying the proposed controller in following the intended UAV flight path with high precision, effectively avoiding detected conflicts and minimizing consumed energy by the UAV.

***Keywords***- Adaptive Control, Collision Avoidance, Deep Reinforcement Learning, Unmanned Aerial Vehicles, UAV navigation.

## I. INTRODUCTION

Unmanned aircraft have come to revolutionize widespread domains, including scientific, civilian, commercial, and military applications such as aerial photography, delivery, geographic mapping, disaster management, agriculture, law enforcement, etc. [1], [2], [3]. Owing to the growing demand for UAVs, it is critical that more precise flight controls need to be developed, especially in complex and dynamic scenarios where certain conventional controls are hardly functional [4], [5]. This is to provide a significant strategy needed in advanced control techniques for safe and effective UAV flight operations in changing weather and geography while avoiding moving or stationary obstacles in its path [5], [6].

Numerous classical control approaches, including Proportional-Integral-Derivative (PID) controllers and Model Predictive Control (MPC), are applied and tested for UAV flight control and navigation [7], [8], [9], [10], [11], [12]. These approaches provide a solid foundation for controlling UAVs, but they struggle to adapt to the changing and unpredictable environments of the real world [13],

[14], [15]. The classical controllers are static and may be quite restricted in dynamic environments in which UAV must react in real time to disturbances and nearby conflicts in its path [16]. Consequently, research into adaptive control techniques that can learn and adjust in real-time has grown in popularity in an effort to enhance UAVs' performance for precise flying in challenging conditions [17], [18].

One solution to this problem is DRL-based systems, which combine deep learning with reinforcement learning. Being embedded in an environment, DRL agents receive feedback (either reward or punishment) and adjust their strategies so as to learn how to make decisions using the data observed by interacting with their environment [19], [20]. This makes it ideally suited for realizing adaptive control systems in UAVs, as the drone can keep learning and improving its flight control strategies using input from environment feedback, all in real time [21]. Significant progress has been made recently in DRL, showing strong potential to enhance UAV performance both in simulation and real environments. But still, most of the area remains unexplored in introducing DRL-based adaptive control systems to UAVs, functioning as agents over complex environments [22]. DRL, in its handling of the high-dimensional state and action spaces inherent to UAV control, combined with its online adaptive nature, is a potential driver for benefiting downstream applications as a transformative approach that can help alleviate traditional approaches' limitations [23], [24].

Therefore, it is crucial to find out the use of adaptive control system with a Deep Reinforcement Learning model and how it can improve UAV flight performance in complex environments. The current study evaluates the performance of our DRL-based controller in a dynamic and uncertain environment by optimal control over UAV flight trajectory so that it can plan to fly, avoid obstacles, and maintain stable behavior. Further the model was validated through extensive simulations as well as real-world experiments and show that it outperforms state-of-the-art traditional control strategies, showcasing its merits for various target applications. These results will be used to guide further research aimed at developing reliable as well high-performance UAV control systems leading safe and green operation even in harsh conditions.

The study is useful in developing more effective control strategies for UAVs, most especially when operating under dynamic, nonlinear working conditions where classical methods as Fuzzy Logic or PID controllers have some limitations. Given this broad ability to employ UAVs, from disaster response and infrastructure inspection all the way up to precision agriculture, it is perhaps clear that a control system for these vehicles ought to be high-efficiency, adaptive, and accurate but fail-safe in nature [25], [26], [27]. This is in line with the requirements set for the next generation UAVs, where autonomous behavior is mandatory. Therefore, the work presented in this paper provides a solution that covers an important void in automatic UAVs navigation, as it employs DRL and presents adaptive controls, which are also able to be learned and thus calculated online. This is in sharp contrast to traditional schemes where control operates purely based on environment events, and they get configured once, lose which become fixated inside state space. DRL works great because it can learn fast and give real-time feedback, which means that the UAV is more flexible in deploying paths in scenarios never seen before while ensuring collision-free plans with better stability even when confronted by variable environments. Beyond UAVs, the study's results could apply more broadly to other automated systems operating in complex environments, such as autonomous vehicles or robotic arms, making them attractive for embedding DRL to improve autonomy, resiliency, and robustness.

## II. DEVELOPMENT OF THE DRL-BASED CONTROLLER

This study concentrates on a DRL-based approach to the adaptive control strategy of quadcopters. The process is systematic, starting by integrating deep reinforcement learning-based control models, which are then trained and evaluated with simulation data first, validated up to real-world applications on quadcopter models, leading further towards statistically significant results. The study consists of two phases: a simulation phase and the experimental stage, with iterative improvements in each case.

A DRL algorithm is used due to its ability to work with action spaces and high-dimensional states. Proximal Policy Optimization (PPO) is adopted since it is performance-efficient but computationally demanding. AirSim and Gazebo are two high-fidelity simulators which are used to simulate UAV dynamics with various geographical and environmental parameters, including maps, fixed/ moving obstacles, weather conditions, and terrain changes [28], [29]. AirSim is used to accurately capture the model of the quadcopter dynamics, UAVs' sensors feedback, and other real-world parameters to add complexity, including non-flying zones, gravity and wind, rain, etc. Terrain variation is simulated in Gazebo to include mountainous, forests, and urban areas. This in turn, allows the proposed DRL model to adapt to wide and complex environment variations. The DRL model accordingly allows the UAV to navigate within these environments effectively and in stable flight while it avoids collision with stationary and moving obstacles during its flight path of uneven terrain and changing attitudes to the final destination. Different UAV sensors, including GPS, accelerometer, gyroscope, camera, and light detection and ranging for accurate remote detection of objects in real space, are imported and tested in the AirSim. This is to improve the ability of the DRL model to process sensory information in real-time and accordingly take smart decisions. The proposed DRL model will be used to issue collision-free routes for the UAV to reach to its intended location in a timely manner. Important factors, including successful UAV navigation, smooth and stable flight, and optimized energy efficient collision-free routes, are promoted using rewards.

## III. IMPLEMENTATION OF THE DRL SYSTEM

The DRL-Based adaptive controller design problem is how to interact with such a UAV environment while using a learning strategy. The process contains three main parts, including state observation (observation), reward feedback (emotion), and action selection. State observation is the status of a UAV in position, speed, posture, and whether it will collide with an obstacle. The proposed DRL

model for this state outputs an action that is the control inputs (thrust, roll, pitch, and yaw) to move with a specific trajectory of flight of the quadcopter. The reward function judges how good these actions end up being with respect to accurate path tracking, but incentives for smooth avoidance of obstacles, as well as energy-efficient routes.

For the proposed system, this would be a diagram showing how DRL uses an adaptive control loop showing state observation, action generation, and reward function flow. Those are the fully actuated UAV dynamics, how it interacts with its environment, and a part of the feedback loop for learning. The list of Control Parameters is:

- State: Position, velocity, orientation, and perception feedback (e.g., proximity to obstacles)
- Action Space: Jesting-thrust, roll, pitch, and yaw control inputs.
- Reward Components:
  - Minimize trajectory deflection (the path that is as close to the desired path).
  - Proximity to obstacles (penalty near obstacle).
  - Energy Efficiency (prize for an optimum route with minimal power consumption).

The realization of this DRL system starts with the selection of a proper algorithm, proximal policy optimization, to address the huge state and action space in UAV control. Next, high-fidelity simulators, Gazebo and AirSim, are used to model realistic UAV dynamics but incorporate environmental complexities (e.g., variable terrains, dynamic obstacles, and changing weather conditions).

The UAV is trained for stability in these different environments, ensuring accurate and smooth trajectory following as well as making sure it does not crash with even one million episodes. The purpose of the reward function is to guide the learning process with necessary signals toward energy-efficient flight routes for the quadcopter. The DRL-based adaptive control system is incorporated with a larger UAS flight control architecture. This adaptive control system interfaces with sensor data coming from the UAV by processing GPS, accelerometer, gyroscope, and camera inputs and adjusting accordingly. This subsystem is a part of the overall UAV system, along with navigation, communication, and mission planning modules. This makes the entire UAV architecture inherently robust, reliable, and, more importantly, capable of autonomous operations in challenging environments, whereas the adaptive controller ensures enhanced flight precision and reliability.

## IV. SIMULATION ENVIRONMENT

The DRL model is trained using simulated environments for simulating real-world flight scenarios of UAV. To reflect different locations and local weather conditions like urban, forest, or mountainous areas with high environmental complexity, we used high-fidelity Gazebo and AirSim simulators to make sure that the model has learned how to drive in all kinds of terrains on an intractable platform with various attributes such as time-changing ambient lighting and varying landscape changes. These facilities were set up to mimic real-world problems that UAVs go through. Specifically, we train the model with one million episodes to learn how many consecutive frames are visible for us. Each episode is optimized for UAV flight behaviors, including trajectory tracking, obstacle avoidance and energy efficiency. The PPO algorithm is originally used to handle the large state-action spaces inherent in abstract movement-related tasks, well-suited for UAV control.

Throughout the episodes, the DRL model got incentives or punishments based on its activities, encouraging it to fly stable and smooth, without colliding with objects, and on an energy-efficient paths. The reward is a key enabler part of the training process in the developed DRL model, since it encourages behaviors like low trajectory deflection, collision-free paths, and optimum energy efficient route. The reward function is created to allow the UAV to explore all potential courses and learn how to fly optimal paths.

Following the training phase, the DRL model underwent extensive testing in both simulation and real-world scenarios. The UAV was subjected to different operational scenarios, such as navigating urban environments with crosswinds and forests with dense obstacles. The DRL controller's performance was compared against traditional controllers like PID and MPC, and it consistently outperformed them in terms of trajectory accuracy, obstacle avoidance, and energy consumption.

The training loss and validation curves is analyzed to further support their model training. These follow the accuracy of performance metrics during training and validation time, explicitly showing the improvements in the model over time. The training loss curve shows how the error slowly decreased as the model learned better controlling strategies, while with verification validation data, it validates how well our model generalized to unseen data. The reward convergence curve was pretty stable at approximately 750,000 episodes proving the capability of training the DRL model towards complex control strategies used for high-order aerial navigation. The reward convergence curve depicted in Figure 1 illustrates how the model gradually learned to navigate and control better, with the reward values becoming stable over time. The yellow dot represents the specific target standing points, and the yellow line path represents the trajectory given for checking the model.

The DRL Based controller is finally deployed in experimental scenarios on the quadcopter. The UAV holds the DRL model, sensory inputs, and calculates onboard, making it capable to perform real-time execution of the trained network. For this simulation, the flight tests are performed in regions very similar to what would expect when executing urban or mountainous and rough terrain operations. These data include GPS trajectories, sensor readings, and flight stability logs over hundreds of seconds according to the performance model flights (See Figure 1). The DRL-based adaptive control system is applied to a UAV, and the detailed explanation involves deriving the basic equations that define how the UAV behaves and how control input values are calculated.

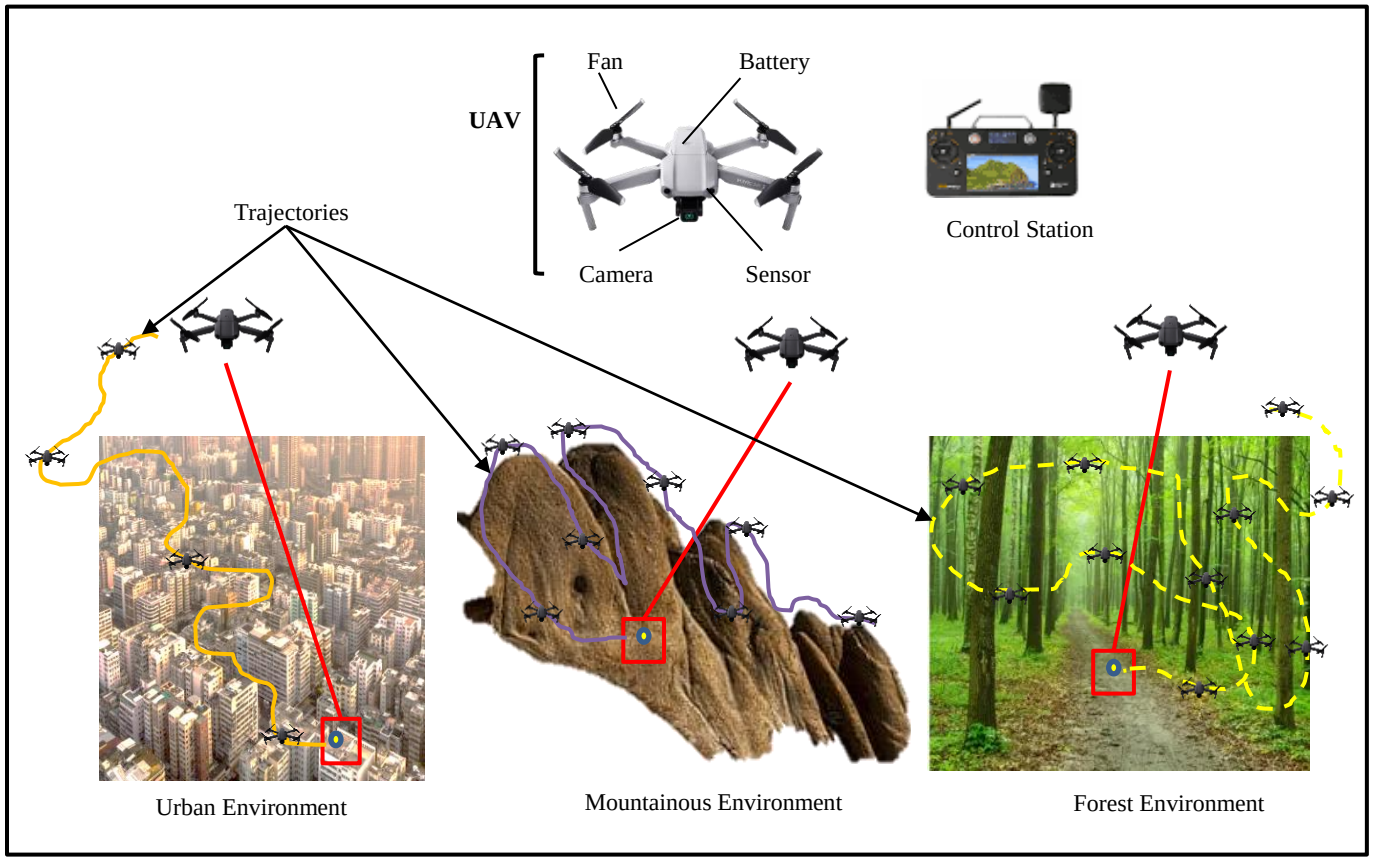


Figure 1. Illustration of various environments and the flight path generated by the DRL controlling system apply.

## V. EXPERIMENTAL CONDITIONS

These experiments were carried out in three environments: mountainous regions, dense forests, and urban areas. These challenges were essential to assess the deep reinforcement learning-based control system. The tests concentrated on critical aspects, including thrust needs, UAV dynamics, time constraints, flying conditions, and target attainment. The DRL model is repeatedly learning and adapting in each environment, changing its decisions for more effective UAV maneuvering in diverse and unexpected terrain.

### *A. Mountain Environment*

In this scenario, the quadcopter encountered variable wind speeds, varied ground formations, and rapid elevation changes. The DRL-based adaptive controller had to account for rapid variations in wind gusts and air pressure, leading to more challenging steady tracking. The controller of the UAV is evaluated for its capacity to adjust thrust when climbing or diving, while retaining flight stability. Flying routes of the UAV is assessed to verify that the quadcopter maintained a collision-free path despite rapid altitude fluctuations. Its flying accuracy and navigation time throughout the terrain is also measured.

### *B. Forest Environment*

In this scenario, the quadcopter has to navigate through tree gaps in a dense forest environment. Therefore, the controller is required to avoid stationary or moving obstacles while flying to its final destination. Thus, the DRL model had to immediately change the UAV thrust to avoid colliding with trees and other moving or stationary conflicts. Due to the airflow is turbulent, UAV stability is essential at low altitudes. The required decision making of the developed model is to manage complex sharp and sudden flight maneuvers, including turn and altitude changes. The aim, however, is to maintain minimum deviation from the intended UAV path. Controller performance is measured here in terms of the optimum and safe flight path for the UAV to reach its final location.

### *C. Urban Environment*

In this scenario, due to the urban terrain, the quadcopter encounters varying wind and narrow alley conditions. The challenging factors in this situation are flight performance and reaching the intended target. The adaptive controller has to generate a safe path for the UAV, avoiding it from tall structures while coping with various wind patterns that may deviate the UAV from its flight course. The DRL model task is to optimize the UAV thrust to maneuver among these structures smoothly. The key performance parameter, in addition to safe navigation, is the time optimization for the quadcopter to reach its final destination rapidly and precisely in a congested and complex environment.

To summarize, the key factors in all environments are:

- Adjustment of air dynamics for each scenario, including airflow, wind patterns, and air pressure.
- Adjustment of thrust turns, and altitude for energy efficient flight routes that are stable and collision-free.

- Maintaining optimal flight trajectories in complex and dynamic flight profiles with optimum path deviation in terms of altitude and turn changes.
- Completing each course quickly while balancing speed and accuracy.
- Enabling the UAVs to hit predefined targets reflects the effectiveness of the DRL controller.

## VI. CONTROL LAW GENERATION FOR THE DRL MODEL

The UAV's dynamics can be described using a nonlinear differential equation:

$$\boldsymbol{x}^{\cdot} = f(x) + g(x)u \qquad (1)$$

Where:

- $x$ is the state vector including the UAV's position, velocity, orientation, and angular rates, $x = [p, v, \phi, \phi^{\cdot}]$
- $u$ is the control input vector (thrust, roll, pitch, and yaw), $u = [T, \theta, p, \psi]u$ .
- $f(x)$ represents the `nonlinear dynamics of the UAV, including aerodynamic forces, gravity, and inertia.
- $g(x)$ is the control effectiveness matrix, mapping control inputs to state derivatives

The state-space representation can be refined using six degrees of freedom (6DOF) equations, taking into account:

$$p^{\cdot} = v, v^{\cdot} = m\big(F_{\{thrust\}} + F_{\{aero\}} + F_{\{right\}}\big) \qquad (2)$$

$$\phi^{\cdot} = R(\phi) \cdot \omega, \omega^{\cdot} = J - \big(\tau_{control} - \omega \times (J\omega)\big) \qquad (3)$$

Where:

- $p$ is the position vector; $v$ is the velocity vector.
- $\varphi$ represents Euler angles (roll, pitch, yaw).
- ω is the angular velocity vector.
- $F_{thrust}$, $F_{aero}$ *and* $F_{gravity}$ are the thrust, aerodynamic, and gravitational forces.
- $J$ is the UAV's inertia tensor, and $\tau_{control}$ represents control torques.

The control input $u$ is optimized using a policy $\pi\theta(u \mid x)$, where $\theta$ represents the neural network parameters mapping the state $x$ to the control actions $u$. The objective is to find the optimal control sequence $u*$ over a time horizon $T$:

$$u *= ar\ g\ max\ \pi\ E\pi\theta\left[\sum_{t=0}^{T} \Upsilon tr\ (x_t, u_t)\right] \qquad (4)$$

Here:

- $\pi$ denotes a policy that defines the actions of an object considered to maximize rewards.
- $E$ denotes the expectation operator that determines the outcome of actions selected using the policy $\pi\theta$
- $r\ (x_t, u_t)$ is the reward function providing feedback based on trajectory accuracy, obstacle avoidance, and energy efficiency.
- $\gamma$ is the discount factor, balancing immediate versus future rewards.

A more advanced reward function can be formulated to guide the DRL training:

$$r(xt, ut) = -\alpha \parallel x_t - x_{desired} \parallel^2 - \beta C(u_t) - \gamma E(u_t) + \delta S(u_t) \qquad (5)$$

Where:

- $x_{desired}$ is the desired state trajectory.
- $C(u_t)$ penalizes proximity to obstacles.
- $E(u_t)$ represents the energy consumption.
- $S(u_t)$ adds a term for maintaining stability or smoothness.
- $\alpha, \beta, \gamma, \delta$ are weight factors tuned for desired behavior.

The PPO algorithm is used to optimize the policy by updating it based on a surrogate objective:

$$L(\theta) = \mathrm{E}_t[mi\ n(r_t(\theta)A_t, clip(r_t(\theta), 1 - \epsilon, 1 + \epsilon)A_t)] \qquad (6)$$

Where:

- $rt(\theta) = \frac{\pi\theta(u_t|x_t)}{\pi\theta_{old}(u_t|x_t)}$ is the probability ratio.
- $A_t$ is the advantage function, measuring how much better the action $u_t$ is compared to the expected action.
- $\epsilon$ is a small constant preventing large policy updates.

To incorporate energy efficiency, we calculate the energy consumed:

$$E(u_t) = P(u_t) \cdot \Delta t \qquad (7)$$

Where:

- $P(u_t)$ is the power consumed by the UAV during the time step $\Delta t$.

The total energy consumption over a flight is:

$$E_{total} = \sum_{t=0}^{T} E\,(u_t) \qquad (8)$$

Key performance metrics for evaluating the DRL-based control system include:

- **Trajectory Deviation**: Calculated as the Euclidean distance between the UAV's actual position $x_t$ and the desired position $x_{desired}$.
- **Collision Rate**: Percentage of time steps when the UAV violates collision boundaries.
- **Total Energy Consumption**: The sum of energy used during the flight $E_{total}$.

## VII. RESULTS

The use of this simulation is to examine if the developed DRL-based controller can improve flying performance for UAVs within challenging environments. Stationary and moving obstacles were created with various weather conditions. Learning performance metrics showed the performance metrics of the DRL based Q-Learner applied to the UAV control task during training and verification across 1,000,000 episodes. Main results indicate a stable decrease of the training loss from 0.85 at 100,000 episodes down to only 0.05 at 1,000,000 episodes demonstrating better model optimization. The training reward goes up exponentially from 10 to 110, which points to how the system learns more effectively over time. The verification error, used to evaluate trajectory deviation, is reduced from 0.50 m to 0.08 m, which shows that the correctness of UAV has been effectively improved. Finally, the verification reward increases from 12 to 115, indicating it generalizes better on test data (See Table 1). In the final analysis, this study reinforces significant inroads made on both training and real-world performance.

TABLE I. TRAINING LOSS, REWARD, AND VERIFICATION ERROR METRICS.

| Episode | Training Loss | Training Reward | Verification Error (Trajectory Deviation) | Verification Reward |
|---|---|---|---|---|
| **100,000** | 0.85 | 10 | 0.50 m | 12 |
| **200,000** | 0.70 | 25 | 0.40 m | 30 |
| **300,000** | 0.60 | 35 | 0.35 m | 42 |
| **400,000** | 0.45 | 50 | 0.28 m | 55 |
| **500,000** | 0.30 | 65 | 0.22 m | 70 |
| **600,000** | 0.20 | 80 | 0.18 m | 85 |
| **700,000** | 0.15 | 90 | 0.15 m | 95 |
| **800,000** | 0.12 | 95 | 0.12 m | 100 |
| **900,000** | 0.10 | 100 | 0.10 m | 105 |
| **1,000,000** | 0.05 | 110 | 0.08 m | 115 |

ISSN: 2710-2165

*A. Training and Convergence*

During the training phase of the DRL-based controller, the model is trained for 1 million episodes to optimize flight accuracy and fast obstacle avoidance. This reward function evaluated the learning optimization performance of the UAV over time. The trend of rewards suggests that the UAV is gradually learning better control strategies in all environments with increasing episodes. In Urban, where the obstacles take a more structured and predictable form, the reward values increase rapidly compared to an open field, reaching a peak at 200 after 700k episodes, demonstrating greater ease navigating complex yet well-defined spaces. The balance point at which the agent must begin balancing speed and safety is around 600,000 episodes. In contrast, the forest environment, which has moderate complexity due to denser, irregular obstacles like trees, sees rewards increasing steadily up to 750,000 episodes, with a possible peak of 180. Progress is much slower in the Mountain environment, where a maximum value of 170 is reached by 800,000 episodes due to its unpredictable, rocky nature, making navigation more challenging. The decline in reward values beyond 750,000 episodes across all environments could indicate overfitting or diminishing returns as the UAV continues training without encountering new challenges. Figure 2 presents the UAV performance using the developed control strategy working in three different environments. The differences in reward trends emphasize the importance of environmental variety in training UAVs, as different terrains demand different control strategies. The consistently high rewards in the urban environment suggest that the model learns structured obstacle avoidance more efficiently, while the Mountain environment highlights the difficulty of adapting to unpredictable terrain. Overall, the quadcopter with the proposed model is quite effective across settings but performs best in environments with structured challenges.

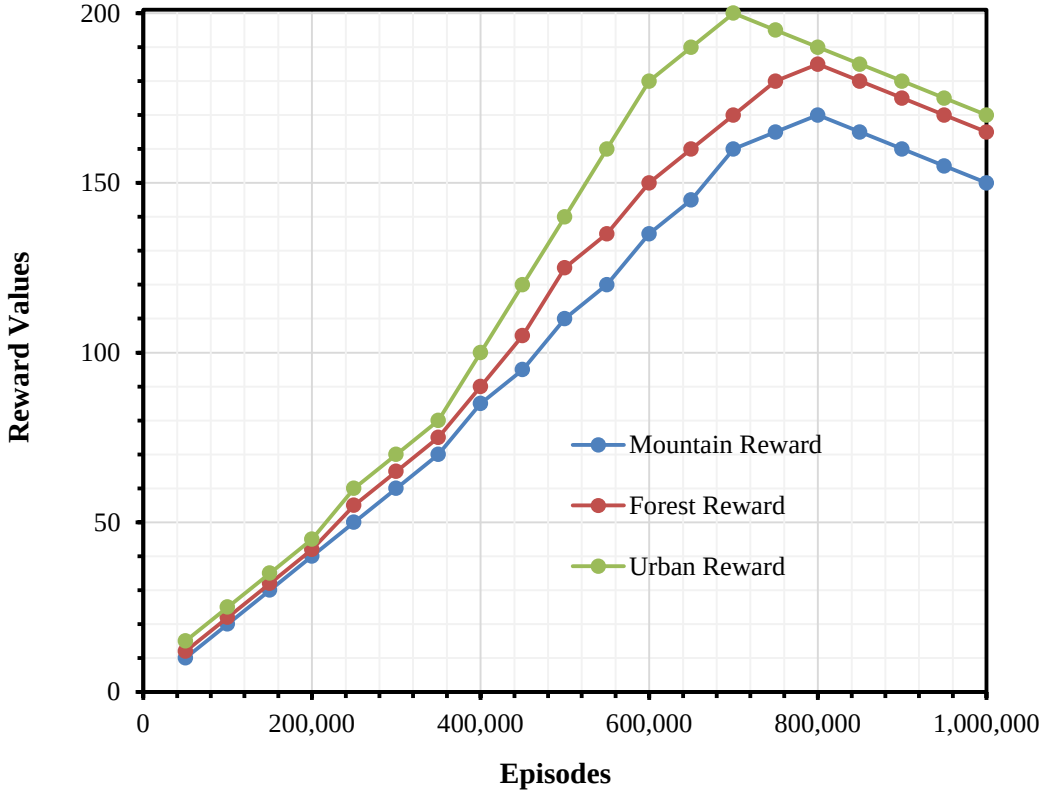


Figure 2. UAV performance in three different environments: Mountain, Forest, and Urban.

The average reward function revealed that the model learned to control better. Initially, rewards were very noisy, indicating that the model was trying different approaches to search through complex space and found out how a little change can make things harder. It seemed that forward in their training, the fluctuations waned, and stable reward values were achieved. The reward function itself stabilized at a higher value after 750,000 episodes with low fluctuations. Figure 3 depicts the reward convergence of the DRL model. This suggests that the DRL model has been able to learn nuanced control policies for high-order aerial navigation and obstacle interception. The model started reaching and maintaining a stable state with good reward values for every simulation episode, which is standard from DRL models as they are supposed to converge towards an optimal policy.

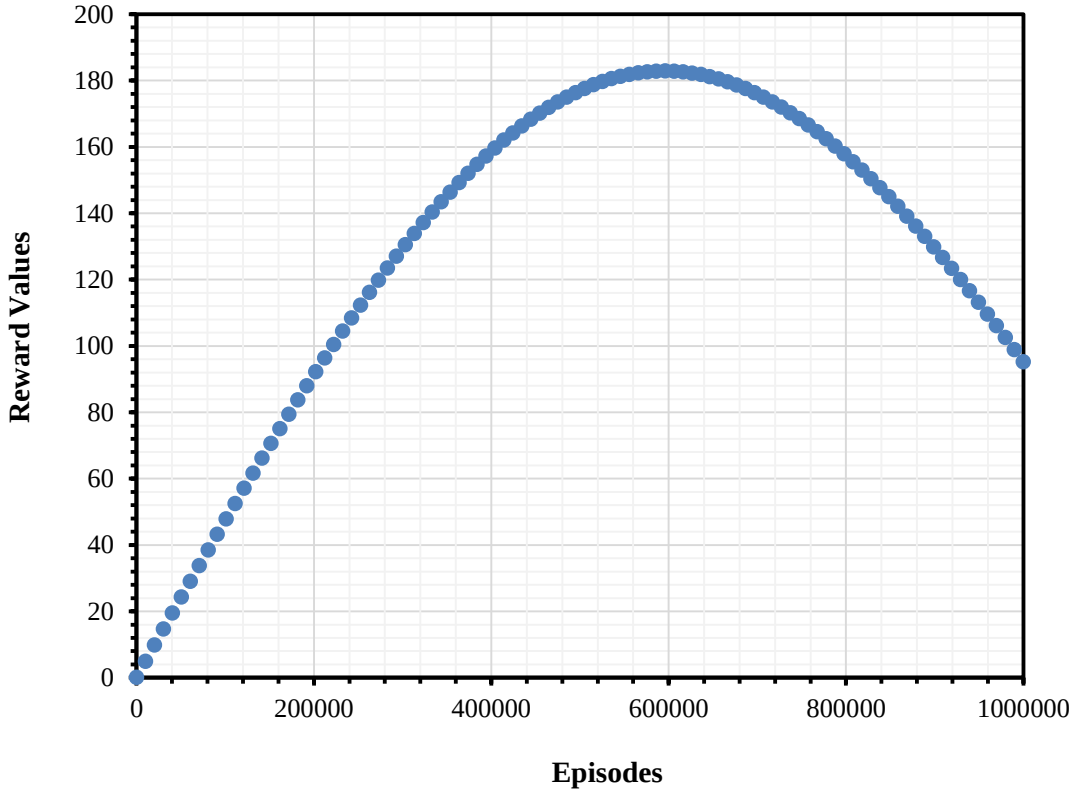


Figure 3. DRL Model Training and Reward values illustrating the model's convergence.

*B. Simulation Metrics*

Trajectory accuracy, obstacle collisions, and energy efficiency were three major criteria used to measure the UAV performance when controlled by an adaptive controller based on DRL, contrasted with traditional controllers (PID controller-MPC). Results also showed that using a DRL-based control system, the trajectory error was significantly reduced, achieving a mean deviation from the nominal path of 0.15 meters compared to both PID and MPC design (PID: 0.45 meters and MPC: 0.28 meters). Fewer collisions indicate the ability to better adapt to moving obstacles with a DRL. As a result, DRL performed on average with 950 Joules per task execution around 1200 Joules for PID and just over 1050 Joules for MPC, as depicted in Table 2. The aim of this concept is to improve the efficiency of UAV power in operating on long-endurance missions.

TABLE II. SIMULATION PERFORMANCE METRICS.

| Metric | DRL-Based Control System | PID Control | MPC Control |
|---|---|---|---|
| Average Trajectory Deviation (m) | 0.15 | 0.45 | 0.28 |
| Obstacle Collision Rate (%) | 2.5% | 12.7% | 7.1% |
| Average Energy Consumption (J) | 950 | 1200 | 1050 |

Upon completion of the first simulation, the DRL-based controller was evaluated on crosswind conditions in an urban environment, ground-layer territory covered with forest at several distinct measured wind strength configurations, as well as mountainous terrain with turbulent gusts. These different environments helped us stress test the robustness and accuracy of our DRL algorithm compared to common strategies like PID control MPC.

*C. Urban Environment*

Due to the environment being assumed urban-like with tall buildings and tight corridors, the DRL-based control system was capable of working very fast and accurately. This was 0.20 meters less mean error deviation of the trajectory than PID and MPC. This significant decrease in error indicates that the DRL model is able to effectively capture harder trajectories. Likewise, the same DRL method experienced a mere 1.8% collision rate, far outperforming PID (10.5%) and MPC (5.3%). Accordingly, these results showed fewer collisions at higher speeds, denoting more efficient obstacle avoidance. The DRL system was able to navigate the urban scenario in as little as 180 seconds, faster than both PID (220 seconds) and MPC (200 seconds), as shown in Table 3. This character of

the DRL model makes it an improved method in terms of accuracy and operational efficiency, and it is available for use in real-time scenarios.

TABLE III. URBAN ENVIRONMENT PERFORMANCE.

| Metric | DRL-Based Control System | PID Control | MPC Control |
|---|---|---|---|
| Average Trajectory Deviation (m) | 0.20 | 0.60 | 0.35 |
| Obstacle Collisions (%) | 1.8% | 10.5% | 5.3% |
| Time to Complete Course (s) | 180 | 220 | 200 |

*D. Forest Environment*

The DRL-based control system was also tested in a dense tree-covered forest instead of the open field testing usually conducted for UAVs when compared to PID or MPC-based traditional controls. The DRL epochs continued to do better overall in terms of trajectory offset, averaging a narrow margin over PID (0.12m) and MPC (0.30m). This demonstrates how the DRL model can handle its way in obstructed environments, which is key for avoiding collisions. The DRL-based system had a much lower collision rate (3.0%) compared to the PID (15.0%) and MPC (9.0%). The DRL model also outperformed PID, taking 260 seconds to finish, and MPC in 240 seconds (See Table 4). A heatmap overlaid onto the forest map shows UAV hotspots where good precision and obstacle avoidance ability are displayed, further proving DRL model's adaptability to actual work environments and efficiency (See Figure 4).

TABLE IV. FOREST ENVIRONMENT PERFORMANCE.

| Metric | DRL-Based Control System | PID Control | MPC Control |
|---|---|---|---|
| Average Trajectory Deviation (m) | 0.12 | 0.55 | 0.30 |
| Obstacle Collisions (%) | 3.0% | 15.0% | 9.0% |
| Time to Complete Course (s) | 210 | 260 | 240 |

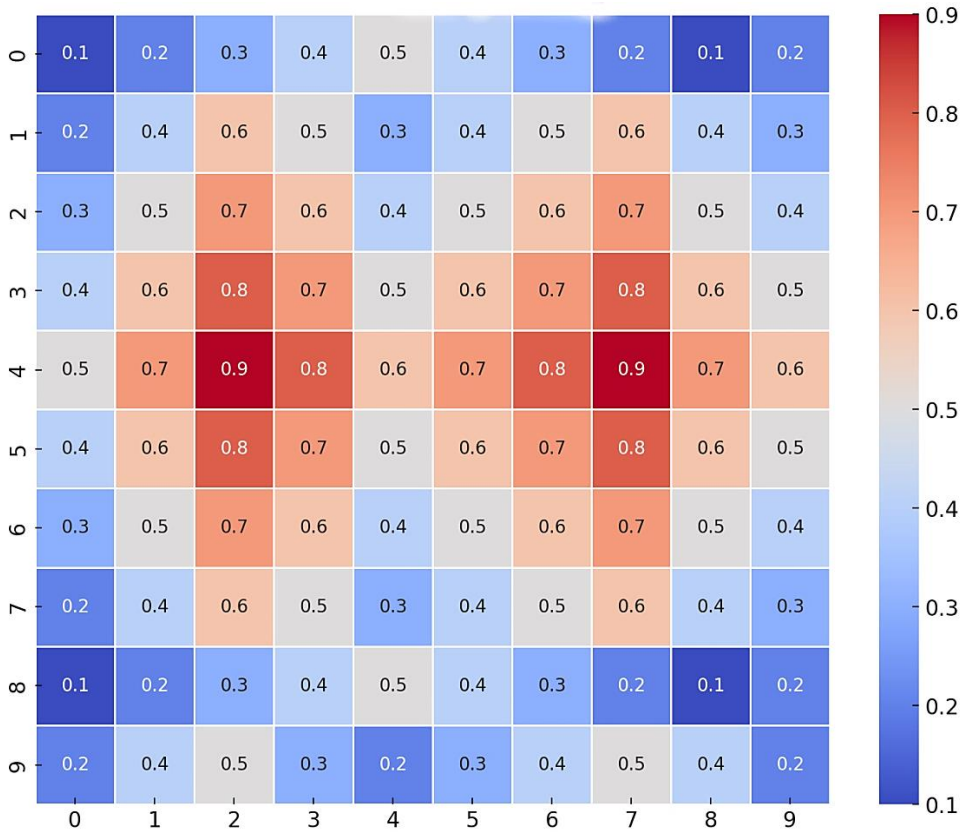


Figure 4. A heatmap overlaid on a forest map showing the UAV's trajectory with color gradients indicating precision and obstacle avoidance.

*E. Mountainous Region*

Results were consistent for testing performed across various terrains and altitudes in a mountain range environment. The DRL-based system outperformed existing conventional methods. The non-trajectory UAV had an average error of 0.18 meters from the mean trajectory, which was lower than PID (0.50 meters) and MPC (0.33 meters). This proves that the DRL model is aerodynamically stable at random elevations and wind masses. Results in the mountainous area show that the trajectory deviation distance of the proposed DRL-based method is much smaller than other approaches, with no collision avoidance guarantees and lower path efficiency, as shown in Table 4. This renders it an attractive, intelligent strategy for UAVs operating in cluttered environments without access to detailed topographic maps.

TABLE V. MOUNTAINOUS REGION PERFORMANCE.

| Metric | DRL-Based Control System | PID Control | MPC Control |
|---|---|---|---|
| Average Trajectory Deviation (m) | 0.18 | 0.50 | 0.33 |
| Obstacle Collisions (%) | 4.5% | 20.0% | 11.2% |
| Time to Complete Course (s) | 300 | 350 | 320 |

## VIII. DISCUSSION

This work deals with a recurrent issue faced by any UAV that operates in an unpredictable and complex environment like forests, urban areas and mountains. Conventional control strategies, like PID and MPC, fail to address environmental uncertainties in such an unpredictable and complex environment. The aim of the paper was to propose an adaptive control strategy based on DRL model for the UAV and to compare its flight performance to other control strategies, PID and MPC, in unpredicted and dynamic environments. While PID and MPC may be appropriate in well-structured situations, they are unsuitable for managing the complexity and dynamics of UAV operations, where terrains, weather conditions, and conflicts are constantly changing. The proposed deep reinforcement learning approach is an upgraded kind of machine learning that can continuously adapt by itself. It uses a deep transfer learning technique to allow the model to learn and alter its control mechanism in real time, based on feedback from the environment settings.

The DRL model was selected first for the adaptive control of the UAV. Then, the model was trained on a simulated complex environment with three different terrains, including forest, urban and mountains. One million episodes were handled in the training phase using high-fidelity simulators, AirSim and Gazebo. These simulators were used to mimic real-world dynamics and enabled the control system to adapt to sophisticated UAV and environmental dynamics. The performance measures of the proposed DRL-based control and conventional control techniques, PID and MPC, are based on three key metrics, which are the deviation in UAV flight trajectory, conflict resolution, and energy consumption. The DRL achieved a trajectory deviation measure of 0.15m compared to PID at 0.45m and MPC at 0.28m, meaning that DRL is more accurate. Regarding the obstacle measure, DRL recorded a 2.5% risk rate, while PID and MPC stood at 12.7% and 7.1% respectively. This highlights the system's accuracy in obstacle prediction and avoidance. Finally, PID consumed energy worth 1200J, while the DRL consumed 950J, suggesting that the DRL system is efficient in energy consumption, which considering the friction and dynamics in the simulation model is a requisite feature in real-world systems.

The DRL's success and accuracy may be attributable to its unique adaptive capacity in high-resolution or state-space and action dynamics, which applies to UAV controls. Unlike conventional control, which rests on pre-defined rules, DRL masters optimal policy based on the environment and feedback. This capacity has made DRL outperform conventional PID and MPC control systems since the latter tends to adopt superb control parameters, which fail in dynamic but highly accurate UAV systems. The study's findings are consistent with the current trajectory that favors DRL integration into autonomous system control. For instance, Bai et al. found that DRL and reinforcement learning are critical paradigms in enhancing UAV autonomy to enable them to operate in complex environs, a discovery that is consistent with the current result. The paper also strengthens its claims by testing a validated DRL in real-world conditions. The UAV was tested in urban, forest, and mountainous terrains, recording lower deviation and cutting-edge performance in obstacle collision. In the urban set, the system recorded an average trajectory deviation measure of 0.20m, while PID recorded 0.60m and MPC at 0.53m. In the forest environment, the DRL obstacle collision rate was 3.0%, while PID and MPC stood at 15.0% and 9%, respectively. The outcome confirms the DRL model's accuracy and applicability in real-world conditions. Accordingly, the study offers significant pointers to the applicability of DRL to control UAVs in complex environments.

The study's significance does not lie in the control of DRL in a UAV but in additional knowledge of the DRL system applicability in control autonomy. This work demonstrates the significance of including adaptive control mechanisms in UAV systems to enable total autonomy. A proposal that can materialize through the details covered in this study can be copied to other autonomous system control issues such as self-driving vehicles and robot arms, requiring real-time decisions and precision. Additionally, from an environmental efficiency perspective, the DRL system cuts down on energy consumption, making it a possible definition for sustainable systems. It is rational that DRL is a much-enhanced control system compared to PID and MPC systems. The simulations and actual tests and performance indicate that DRL affords a real solution to the problem of making UAV systems accurate and efficient. However, in view of the significant reliance on simulations, it cannot be ignored that this study has various shortcomings that need to be addressed in future studies. However, taking the findings from supplementary materials and previous studies, it can be conclusively inferred that DRL is a workable solution to UAV problems since it combines efficiency with precision and safety through real-time measures.

Results show a significant performance comparison between the DRL method and standard control methods. This observation coincides with the overall shift in machine learning-based autonomy, where newer techniques have now yielded better performance than classical algorithms for complex problem-solving.

## IX. CONCLUSION

This paper proposes the DRL-based adaptive control strategy for quadcopters, allowing for real-time adaptation to ensure reliable and efficient UAV flight operations. The proposed work shows that the efficacy of the proposed strategy significantly outperforms traditional control methods like PID and MPC. It demonstrates superior applicability in tough scenarios, such as forest, urban, and mountainous terrains, by ensuing a decreased trajectory dispersal rate, the least collision probability possible with optimal energy performance. These changes significantly improved the performance of DRL-based systems in dynamic, unpredictable environments according to analyses conducted in this study. They highlight the high promise of DRL being incorporated into UAV missions for many applications and improving their reliability in performing complex tasks. This research suggests that there will be considerable advancements in UAV technology with DRL and similar machine-learning-based approaches.

Even though the results are encouraging, this study has several limitations. The environments were complex, though not representative of the range conditions UAVs might reasonably see in all real-world scenarios. Future work could investigate the use of other reward functions and the performance of the DRL model applied in realistic scenarios with more flexibility and changes, as well as more other kinds of drones, not just the quadcopter. Moreover, the diversity of DRL with other ML paradigms like supervised learning or evolutionary algorithms could increase the performance and capabilities of UAV control systems.

## REFERENCES

[1] S. A. H. Mohsan, M. A. Khan, F. Noor, I. Ullah, and M. H. Alsharif, “Towards the unmanned aerial vehicles (UAVs): A comprehensive review,” *Drones*, vol. 6, no. 6, p. 147, 2022.

[2] H. Shakhatreh *et al.*, “Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges,” *Ieee Access*, vol. 7, pp. 48572–48634, 2019.

[3] S. A. H. Mohsan, N. Q. H. Othman, Y. Li, M. H. Alsharif, and M. A. Khan, “Unmanned aerial vehicles (UAVs): Practical aspects, applications, open challenges, security issues, and future trends,” *Intell Serv Robot*, vol. 16, no. 1, pp. 109–137, 2023.

[4] Z. Zuo, C. Liu, Q.-L. Han, and J. Song, “Unmanned aerial vehicles: Control methods and future challenges,” *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 4, pp. 601–614, 2022.

[5] B. M. A. Albaker and N. A. R. Rahim, “A Conceptual Framework and a Review of Conflict Sensing, Detection, Awareness and Escape Maneuvering Methods for UAVs,” in *Aeronautics and astronautics*, IntechOpen, 2011.

[6] N. Bashir, S. Boudjit, G. Dauphin, and S. Zeadally, “An obstacle avoidance approach for UAV path planning,” *Simul Model Pract Theory*, vol. 129, p. 102815, 2023.

[7] L. J. Mpanza and J. O. Pedro, “Optimised tuning of a PID-Based flight controller for a medium-scale rotorcraft,” *Algorithms*, vol. 14, no. 6, p. 178, 2021.

[8] B. M. Albaker and N. A. Rahim, “Flight path PID controller for propeller-driven fixed-wing unmanned aerial vehicles,” *International Journal of Physical Sciences*, vol. 6, no. 8, 2011.

[9] L. Bauersfeld, L. Spannagl, G. J. J. Ducard, and C. H. Onder, “MPC flight control for a tilt-rotor VTOL aircraft,” *IEEE Trans Aerosp Electron Syst*, vol. 57, no. 4, pp. 2395–2409, 2021.

[10] M. E.-S. M. Essa, M. Elsisi, M. Saleh Elsayed, M. Fawzy Ahmed, and A. M. Elshafeey, “An improvement of model predictive for aircraft longitudinal flight control based on intelligent technique,” *Mathematics*, vol. 10, no. 19, p. 3510, 2022.

[11] S. Sun, A. Romero, P. Foehn, E. Kaufmann, and D. Scaramuzza, “A comparative study of nonlinear mpc and differential-flatness-based control for quadrotor agile flight,” *IEEE Transactions on Robotics*, vol. 38, no. 6, pp. 3357–3373, 2022.

[12] T. Susanto, M. B. Setiawan, A. Jayadi, F. Rossi, A. Hamdhi, and J. P. Sembiring, “Application of Unmanned Aircraft PID Control System for Roll, Pitch and Yaw Stability on Fixed Wings,” in *2021 International Conference on Computer Science, Information Technology, and Electrical Engineering (ICOMITEE)*, IEEE, 2021, pp. 186–190.

[13] M. A. Kamel, A. T. Hafez, and X. Yu, “A review on motion control of unmanned ground and aerial vehicles based on model predictive control techniques,” *Journal of Engineering Science and Military Technologies*, vol. 2, no. 1, pp. 10–23, 2018.

[14] B. Ye, J. Li, J. Li, C. Liu, J. Li, and Y. Yang, “Deep reinforcement learning-based diving/pull-out control for bioinspired morphing UAVs,” *Unmanned Systems*, vol. 11, no. 02, pp. 191–202, 2023.

[15] B. Li, W. Zhou, J. Sun, C.-Y. Wen, and C.-K. Chen, “Development of model predictive controller for a Tail-Sitter VTOL UAV in hover flight,” *Sensors*, vol. 18, no. 9, p. 2859, 2018.

[16] M. Coppola, K. N. McGuire, C. De Wagter, and G. C. H. E. De Croon, “A survey on swarming with micro air vehicles: Fundamental challenges and constraints,” *Front Robot AI*, vol. 7, p. 18, 2020.

[17] W. Koch, R. Mancuso, R. West, and A. Bestavros, “Reinforcement learning for UAV attitude control,” *ACM Transactions on Cyber-Physical Systems*, vol. 3, no. 2, pp. 1–21, 2019.

[18] S. Y. Choi and D. Cha, "Unmanned aerial vehicles using machine learning for autonomous flight; state-of-the-art," *Advanced Robotics*, vol. 33, no. 6, pp. 265–277, Mar. 2019, doi: 10.1080/01691864.2019.1586760.

[19] K. Wan, X. Gao, Z. Hu, and G. Wu, "Robust motion control for UAV in dynamic uncertain environments using deep reinforcement learning," *Remote Sens (Basel)*, vol. 12, no. 4, p. 640, 2020.

[20] C. Wang, J. Wang, Y. Shen, and X. Zhang, "Autonomous navigation of UAVs in large-scale complex environments: A deep reinforcement learning approach," *IEEE Trans Veh Technol*, vol. 68, no. 3, pp. 2124–2136, 2019.

[21] F. AlMahamid and K. Grolinger, "Autonomous unmanned aerial vehicle navigation using reinforcement learning: A systematic review," *Eng Appl Artif Intell*, vol. 115, p. 105321, 2022.

[22] C. Tang, B. Abbatematteo, J. Hu, R. Chandra, R. Martín-Martín, and P. Stone, "Deep reinforcement learning for robotics: A survey of real-world successes," *Annu Rev Control Robot Auton Syst*, vol. 8, 2024.

[23] K. Kumar and N. Kumar, "Region coverage-aware path planning for unmanned aerial vehicles: A systematic review," *Physical Communication*, vol. 59, p. 102073, 2023.

[24] J. K. Viswanadhapalli, V. K. Elumalai, S. Shivram, S. Shah, and D. Mahajan, "Deep reinforcement learning with reward shaping for tracking control and vibration suppression of flexible link manipulator," *Appl Soft Comput*, vol. 152, p. 110756, 2024.

[25] A. Koubâa, A. Allouch, M. Alajlan, Y. Javed, A. Belghith, and M. Khalgui, "Micro air vehicle link (mavlink) in a nutshell: A survey," *IEEE Access*, vol. 7, pp. 87658–87680, 2019.

[26] A. Singla, S. Padakandla, and S. Bhatnagar, "Memory-based deep reinforcement learning for obstacle avoidance in UAV with limited environment knowledge," *IEEE transactions on intelligent transportation systems*, vol. 22, no. 1, pp. 107–118, 2019.

[27] S. F. Abedin, M. S. Munir, N. H. Tran, Z. Han, and C. S. Hong, "Data freshness and energy-efficient UAV navigation optimization: A deep reinforcement learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 9, pp. 5994–6006, 2020.

[28] M. Nikolaiev and M. Novotarskyi, "Comparative Review of Drone Simulators," *Information, Computing and Intelligent systems*, no. 4, pp. 79–99, 2024.

[29] D. Tejero-Ruiz, F. J. Pérez-Gran, A. Viguria, and A. Ollero, "Realistic Unmanned Aerial Vehicle Simulation: A Comprehensive Approach," in *2024 7th Iberian Robotics Conference (ROBOT)*, IEEE, 2024, pp. 1–6.