2024 Tenth Indian Control Conference (ICC)
9 - 11 December, 2024
IISER Bhopal, India

REINFORCEMENT LEARNING BASED ATTITUDE CONTROL OF QUADCOPTER

Sneha Mohan K
Dept. of Electrical Engineering
College of Engineering Trivandrum
snehamohank15@gmail.com

Arlene Davidson R
Dept. of Electrical Engineering
College of Engineering Trivandrum
arlene@cet.ac.in

Deepak Mishra
Dept. of Avionics
IIST Trivandrum
deepak.mishra@iist.ac.in

***Abstract*—This study investigates the use of reinforcement learning (RL) for controlling a quadcopter's attitude in a pick-and-place situation. Controlling the quadcopter's pitch, roll, and yaw angles precisely and steadily is the main goal. Traditional control systems for quadcopters often rely on predefined models and PID controllers, which may struggle with dynamic environments and uncertainties. The actor-critic method, a powerful RL technique, is employed to train that can effectively stabilize the quadcopter's orientation. The actor-critic method comprises two main components: the actor, which proposes actions based on current policy and the critic, which evaluates these actions by computing the value function. This dual mechanism enables the quadcopter to adaptive improve its control policy through continuous interaction with the environment. The training process involves the quadcopter performing a series of attitude variations, receiving rewards based on its performance, and updating its policy accordingly. Through continuous interaction with the environment, the actor learns to improve its policy, while the critic provides valuable feedback to guide this learning process. An reinforcement learning (RL) based approach for quadcopter attitude control with a focus on pick and place application.**

***Index Terms*—Reinforcement learning, Actor-Critic, Quadcopter, Attitude, Training**

I. INTRODUCTION

Quadcopter is a type of unmanned aerial vehicle (UAV), have become increasingly popular due to its versatile capabilities. It is equipped with four rotors arranged in a square configuration, offers vertical takeoff and landing, hovering, and agile maneuvering. A quadcopter's dynamics are extremely non-linear. In addition, the system has four control inputs and six degrees of freedom, making it under-actuated. The control inputs that govern the vehicle's motion are the propulsion and the torques needed to tilt the quadcopter [1]. The rotor speeds are changed to provide both thrust and torques. The quadcopter's center axis is always the direction of the thrust produced by the rotor blades [2]-[6] Therefore, the quadcopter's axis needs to be tilted in order to accomplish propulsion in a certain direction concerning the vertical. Quadcopter dynamics and control are therefore extremely difficult since the translational motion of a quadcopter is connected with its angular orientation. By varying the torque and speed of each rotor, which is attached to a motor, the quadcopter may be made to fly. The interaction of the rotor forces allows the quadcopter to control its orientation and movement in three dimensions. The two major configurations for the motors are the plus and X configurations. The quadcopter's simultaneous control of its rotor speeds allows it to do a variety of maneuvers, such as rolling, pitching, yawing, and rising or falling. Pitch (forward/backward tilt), roll (sideways tilt), and yaw (rotation around the vertical axis) are the three angular motions that the quadcopter may do by adjusting the speed of its opposing rotors [7]-[10].

Quadcopters have their unique ability to take off, hover, and maneuver in all directions making them ideal for tasks ranging from aerial photography and videography to search and rescue operations and precision agriculture, etc. However, maintaining a stable orientation (attitude) during flight presents a significant challenge. Underactuation and non-linear dynamics are the two main causes of this control complexity. Underactuation occurs when the quadcopter experiences less degrees of freedom (movement in pitch, roll, yaw, and altitude) than control inputs (usually the speed of each rotor). Because of this limitation, complicated control schemes are required to accomplish the desired movements. Additionally, the non-linear dynamics of a quadcopter refer to the unpredictable behavior of the system under certain conditions. Factors like wind gusts, varying payloads, and even small manufacturing imperfections can significantly alter the quadcopter's response to control inputs. The traditional methods are well-established and effective for many applications, they often require meticulous tuning and may struggle to adapt to dynamic or unpredictable environments. As a result, there is a growing interest in exploring advanced control techniques that offer greater adaptability and robustness[11]-[15].

Reinforcement learning (RL) is an area of artificial intelligence where an agent learns to make decisions by interacting with its environment to maximize cumulative rewards. Unlike supervised learning, which relies on labeled training data, RL involves learning optimal policies through trial and error. This makes RL particularly well-suited for control tasks where the system must adapt to changing conditions and uncertainties. Applying RL to quadcopter attitude control presents an opportunity to develop controllers that can autonomously learn and adapt to achieve desired performance. By continuously interacting with the environment,

979-8-3315-1721-2/24/$31.00 ©2024 IEEE

an RL-based controller can improve over time, enhancing the quadcopter's ability to handle a variety of operational scenarios, including disturbances and sensor noise[16]-[18]. Autopilot systems for UAVs often use PID controllers, but these struggle in unpredictable environments. Recent research explores reinforcement learning (RL) algorithms like DDPG, TRPO, and PPO for improved flight control. Studies using the GYM FC simulation environment show that RL, especially PPO, outperforms PID controllers in precision tasks. It examines training methods, neural network architectures, and robustness against challenges like motor failure and wind gusts. Evaluations using signal temporal logic (STL) show RL controllers outperform PID controllers[19]-[20].The novelties of this paper is briefly highlighted as follows,

- The controller design demonstrates the application of reinforcement learning (RL) technique, specifically the actor-critic method for the attitude control of a quadcopter in a pick and place scenario.
- Trains the quadcopter by having it perform a series of attitude adjustments, where it receives rewards based on its performance and continuously updates its control policy to improve stability and precision through RL techniques.
- Develops a comprehensive model of the quadcopter, to accurately simulate and control its attitude during the training and testing phases.

II. MODELLING OF QUADCOPTER

In the inertial frame x, y, and z axes, the quadcopter's absolute linear position is specified by ξ. The roll, pitch and yaw Euler angles η describe the attitude or angular position in the inertial frame. The vectors of linear and angular positions are found in vector q.

$$q = \begin{bmatrix} \xi & \eta \end{bmatrix}^T \tag{1}$$

Where,

$$\begin{aligned} \xi &= \begin{bmatrix} x \\ y \\ z \end{bmatrix} \\ \eta &= \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \end{aligned} \tag{2}$$

The angular and linear velocities in the body frame are represented by ν and V_B, respectively.

$$\begin{aligned} V_B &= \begin{bmatrix} V_x & V_y & V_z \end{bmatrix}^T \\ \nu &= \begin{bmatrix} p & q & r \end{bmatrix}^T \end{aligned} \tag{3}$$

A. Assumptions

- The quadcopter is assumed to be a rigid body.
- The center of the body frame coincides with the center of gravity.
- The quadcopter has a symmetrical construction.

The rotation matrix from the body frame to the inertial frame is

$$R = \begin{bmatrix} C_\theta C_\phi & C_\theta S_\phi S_\psi - S_\theta C_\psi & C_\theta S_\phi C_\psi + S_\theta S_\psi \\ S_\theta C_\phi & S_\theta S_\phi S_\psi + C_\theta C_\psi & S_\theta S_\phi C_\psi - C_\theta S_\psi \\ -S_\phi & C_\phi S_\psi & C_\phi C_\psi \end{bmatrix} \tag{4}$$

in this equation which C_x represents $cos(x)$ and S_x represents $sin(x)$. Since R is an orthogonal rotation matrix, the rotation matrix from the inertial frame to the body frame is R^{-1} = R^T.

The transformation matrix for angular velocities from the body frame to the inertial frame is W_η^{-1}.

$$\dot{\eta} = W_\eta^{-1} \nu \tag{5}$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & S_\phi T_\theta & C_\phi T_\theta \\ 0 & C_\phi & -S_\phi \\ 0 & S_\phi / C_\theta & C_\phi / C_\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{6}$$

The transformation matrix for angular velocities from the inertial frame to the body frame is W_η.

$$\nu = W_\eta \dot{\eta} \tag{7}$$

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -S_\theta \\ 0 & C_\phi & C_\theta S_\phi \\ 0 & -S_\phi & C_\theta C_\phi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{8}$$

The moment required for stopping a rotating item and to rotate a motionless object is determined by the moment of inertia. The moment around all three axes of the quadcopter is needed. So the matrix is given by the equation (9).

$$I = \begin{bmatrix} I_{xx} & -I_{yx} & -I_{xz} \\ -I_{xy} & I_{yy} & -I_{yz} \\ -I_{xz} & -I_{yz} & I_{zz} \end{bmatrix} \tag{9}$$

For a symmetrical body the moment of inertia on opposite sides of the quadcopter cancel each other. Here it is assumed that the quadcopter have symmetric structure with the four arms aligned with the body x-axes and y-axes. Then the inertia matrix is diagonal matrix I.

$$I = \begin{bmatrix} \mathrm{I}_{xx} & 0 & 0 \\ 0 & \mathrm{I}_{yy} & 0 \\ 0 & 0 & \mathrm{I}_{zz} \end{bmatrix} \tag{10}$$

The force f_i is generated in the direction of the rotor axis by the angular velocity of rotor i, represented by the representation of ω_i. The force f_i generated by rotor, i is proportional to the square of its angular velocity ω_i.

$$f_i = k\omega_i^2 \tag{11}$$

where (k) is the lift constant. Torque τ_{Mi} is also generated around the rotor axis by the rotor's acceleration and angular velocity. The torque τ_{Mi} around the rotor axis is also proportional to the square of the angular velocity.

$$\tau_{Mi} = b\omega_i^2 + I_M \dot{\omega}_i \tag{12}$$

where b is the drag constant and I_M is the inertia moment of the rotor. Often, the effect of the angular velocity ω_i on the torque is considered negligible and thus omitted from the equation.

The combined forces of rotors create thrust T in the direction of the body z-axis. The total thrust T generated by the quadcopter is the sum of the thrust forces f_i produced by each of the four rotors (i = 1 to 4).

$$T = \sum_{i=1}^{4} f_i = k \sum_{i=1}^{4} \omega_i^2 \tag{13}$$

Since the thrust acts primarily along the z-axis of the body frame then the equation (13) becomes,

$$T^B = \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix} \tag{14}$$

The torques τ_ϕ, τ_θ and τ_ψ in the direction of the respective body frame angles make up torque T_B, where l is the distance between the quadcopter's center of mass and rotor. These are used to calculate the moment generated by each rotor's thrust.

$$T_B = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = k \begin{bmatrix} lk(-\omega_2^2 + \omega_4^2) \\ lk(-\omega_1^2 + \omega_3^2) \\ \sum_{i=1}^{4} \tau_{Mi} \end{bmatrix} \tag{15}$$

The moments around the roll and pitch axes are generated by the difference in torques between opposing rotors (e.g., rotor 1 and 3 for pitch). The equation uses the squared motor speeds (ω_1^2 and ω_2^2) to account for the direction of the propeller rotation. The moment around the yaw axis is calculated by summing the moments τ_{Mi} generated by each rotor. In conventional quadcopters, two motors spin clockwise and two spin counter-clockwise to cancel out any net yaw torque when the rotors spin at the same speed. By increasing the speed of one pair of rotors relative to the other pair, a yaw moment is created.

Newton-Euler equations can be used to describe its dynamics and the quadcopter is assumed to be rigid body. The force required to accelerate the quadcopter's mass (m) is denoted by $m\dot{V}_B$. The centrifugal force acting on the quadcopter due to its velocity (v) is represented by the cross product $v \times (mV_B)$. G represents the gravitational acceleration acting downwards and $R^T G$ represents the total thrust force generated by the rotors. These forces are summed together according to Newton's second law to determine the net force acting on the quadcopter in the body frame which is given by the equation (16).

$$m\dot{V}_B + v \times (mV_B) = R^T G + T_B \tag{16}$$

In the inertial frame, the centrifugal force is nullified. Thus, only the gravitational force and the magnitude and direction of the thrust are contributing in the acceleration of the quadcopter.

$$m\ddot{\xi} = G + RT_B \tag{17}$$

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{T}{m} \begin{bmatrix} C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi S_\theta C_\phi - C_\psi S_\phi \\ C_\theta C_\phi \end{bmatrix} \tag{18}$$

In the body frame, the angular acceleration of the inertia $I\dot{\nu}$, the centripetal forces $\nu \times (I\nu)$ and the gyroscopic forces are equal to the external torque τ.

$$I\dot{\nu} + \nu \times (I\nu) + \Gamma = \tau \tag{19}$$

$$\dot{\nu} = I^{-1} \left(- \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} I_{xx} p \\ I_{yy} q \\ I_{zz} r \end{bmatrix} - I_r \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega_\Gamma + \tau \right) \tag{20}$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} (I_{yy} - I_{zz})\, qr/I_{xx} \\ (I_{zz} - I_{xx})\, pr/I_{yy} \\ (I_{xx} - I_{yy})\, pq/I_{zz} \end{bmatrix} - I_r \begin{bmatrix} q/I_{xx} \\ -p/I_{yy} \\ 0 \end{bmatrix} \omega_\Gamma + \begin{bmatrix} \tau_\phi/I_{xx} \\ \tau_\theta/I_{yy} \\ \tau_\psi/I_{zz} \end{bmatrix} \tag{21}$$

By modeling the forces and torques generated by the rotors, and accounting for the quadcopter's dynamics through the Newton-Euler equations, the control system can adjust the rotor speeds to achieve desired movements. This allows the quadcopter to maintain or change its attitude effectively in response to control inputs and environmental interactions.

III. Reinforcement learning Controller

Reinforcement learning is a subset of machine learning where an agent learns to make decisions by interacting with an environment. It learns to achieve a goal through trial and error, receiving feedback in the form of rewards or penalties. It is utilized by a variety of software and devices to determine the optimal course of action or behavior for a given circumstance. In this paper actor-crtic method is used for controlling the quadcopter.

Actor-critic reinforcement learning (RL) is a hybrid

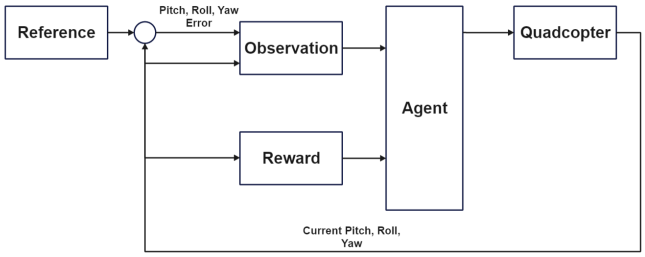


Fig. 1. RL Quadcopter Blockdiagram

approach that combines elements of both value-based (critic) and policy-based (actor) reinforcement learning methods. In classical reinforcement learning, an agent engages with the environment in order to maximize cumulative rewards. By concurrently learning a policy (Actor) and estimating value functions (Critic), the actor-critic algorithm improves this method of learning. The block diagram of quadcopter

control by reinforcement learning is shown in Fig. 1.

Algorithm:

1) Initialize the actor network ($\pi(S;\theta)$) and the critic network ($Q(S,a;\phi)$) with random weights.
2) Initialize target networks π' and Q' with the same weights as the original networks.
3) For the current observation S, the agent selects an action A=($\pi(S;\theta)$)
4) The action A is executed in the environment, resulting in the next observation S' and reward R.
5) Store the experience tuple (S,A,R,S') in the experience buffer.
6) Sample a random mini-batch of experiences (S_i, A_i, R_i, S_i') from the experience buffer.
7) Compute Target Q-Value: $y_i = R_i + \gamma Q_t(S_i', \pi_t, S_i'; \theta_t); \phi_t)$
8) Update Critic by minimize the loss function $L = \frac{1}{2M}\sum_{n=1}^{M}(y_i - Q(S_i, A_i; \phi))$
9) Update Actor $\nabla_\theta J \approx \frac{1}{M}\sum_{n=1}^{M} \nabla_A Q(S_i, A; \phi)|_{(A=\phi(Si;\theta)} \nabla_\theta \pi(S_i;\theta)$
10) Depending on the target update method, update the target networks

A. *Agent*

In quadcopter attitude control, the agent terms refers to the entire control system responsible for maintaining the quadcopter's orientation. The agent is made up of two components, actor and critic. In the actor-critic framework, a value function (the critic) assesses the actor's actions after the actor (the agent) learns a policy to make decisions.

*1) **Actor Network**:* This network takes the current state of the environment as input and outputs the best action that the agent should take in that state. The actor network consists of four layers, which are the input layer (*input*), the first fully connected layer (*fc_1*), the activation layer (*layer*) and the second fully connected layer (*fc_2*).

*2) **Critic Network**:* The purpose of the critic network is to assess the level of an actor's actions in an environment. It predicts the Q-value for state-action pairs. The critic network consist mainly of three parts which are state path, action path and common path. The state path consist of four layers which are feature input layer (*netObsin*), fully connected layer (*fc*), activation layer (*relu*) and the fully connected layer (*CriticStateFC2*). The action path consist of two layers which are feature input layer (*netActin*) and fully connected layer (*CriticActionFC1*). The common path consisting of three layers. They are additional layer (*add*), activation layer (*relu_1*) and fully connected layer (*CriticOutput*).

B. *Observation*

The observation space for the quadcopter's attitude control consists of three primary observation vectors, each corresponding to one of the principal axes of rotation which are pitch, roll, and yaw. The agent receives nine observation signals for this environment, each of which is chosen as a vector signal. The nine observation signals includes the integrals signals of pitch, roll, yaw and the current error signals of pitch, roll, yaw and the remaining three signals are pitch, roll and yaw. The observation vector for pitch, roll and yaw is follows,

$$\text{Observation vector for pitch} = \begin{bmatrix}\int e_p dt & e_p & p\end{bmatrix}^T \quad (22)$$

$$\text{Observation vector for roll} = \begin{bmatrix}\int e_r dt & e_r & r\end{bmatrix}^T \quad (23)$$

$$\text{Observation vector for yaw} = \begin{bmatrix}\int e_y dt & e_y & y\end{bmatrix}^T \quad (24)$$

Here, $\int e_p dt$, $\int e_r dt$ and $\int e_y dt$ represent the integrals of the pitch, roll, and yaw errors over time, respectively. The e_p, e_r and e_y are the current errors for pitch, roll, and yaw. Finally, *p*, *r*, and *y* are the current values of the pitch, roll, and yaw angles, respectively. For the complete observation space that considers all three axes (pitch, roll, and yaw), the combined observation vector is represented as

$$\text{Combined Observation Vector} = \begin{bmatrix}\int e_p dt & e_p & p & \int e_r dt & e_r & r & \int e_y dt & e_y & y\end{bmatrix}^T \quad (25)$$

By incorporating the integral of the error, current error, and current orientation values, the agent can effectively learn to control the quadcopter's attitude.

C. *Reward*

The reward function quantifies how desirable a particular state or action is within the environment. The reward functions for pitch, roll, and yaw control of a quadcopter. The components of a reward function are reward for small error, penalty for large error and penalty for extreme angles.

$$\text{Pitch Reward} = 10(|e_p| < 0.01) - 1(|e_p| \geq 0.01) - 100(p \leq 10 || p \geq 10) \quad (26)$$

$$\text{Roll Reward} = 10(|e_r| < 0.01) - 1(|e_r| \geq 0.01) - 100(r \leq 10 || r \geq 10) \quad (27)$$

$$\text{Yaw Reward} = 10(|e_y| < 0.01) - 1(|e_y| \geq 0.01) - 100(y \leq 10 || y \geq 10) \quad (28)$$

IV. Simulation Results

A. *Training*

The graph shown in Fig. 2 shows the performance of a DDPG agent trained on a quadcopter control task for 2250 training episodes. The agent has gone through a training episode of 2250. The agent achieved an average reward of 148.45 and the highest episode reward achieved was 160 over the 2250 episodes.The episode Q_0 obtained for the highest episode reward is 593.7894. Overall, increasing the number of training episodes generally leads to better performance

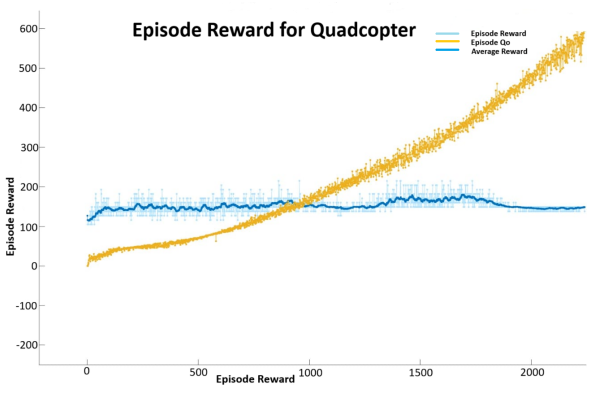


Fig. 2. Training for 2250 Episode

in reinforcement learning. The Q_0 value increases when increasing the number of episodes.

B. *Attitude Control*

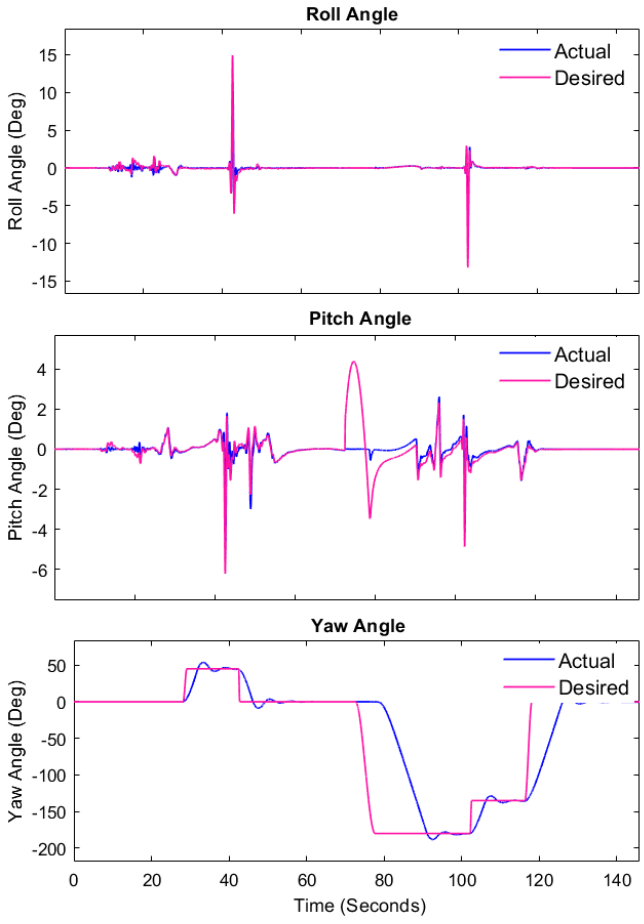


Fig. 3. Angles of Quadcopter

The angular movements of a quadcopter, showing the roll, pitch, and yaw angles over time is illustrated in the Fig. 3. Each subplot represents the actual and desired values for one of the three angles. The roll and pitch angles show relatively good tracking performance with minor deviations, indicating stable flight with occasional disturbances. The yaw angle shows larger deviations, suggesting more corrections needed by increasing the training period of the quadcopter.

C. *Pick and Place*

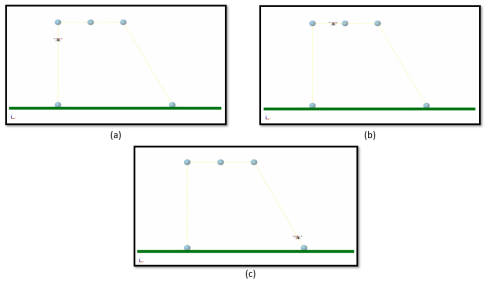


Fig. 4. Pick and Place

Fig. 4 illustrates the pick-and-place operation of a quadcopter. The quadcopter starts on the ground at the initial position. The first point above the initial position indicates the takeoff maneuver in the Fig. 4(a). The quadcopter ascends vertically to a certain height. After reaching the desired altitude, the quadcopter moves horizontally to the pick position which is shown in Fig. 4(b). The quadcopter descends vertically to reach the object at the placing position shown in Fig. 4(c). Upon reaching the object, the quadcopter placing it up using a gripping mechanism. The return movement of the quadcopter is shown in the Fig. 5.

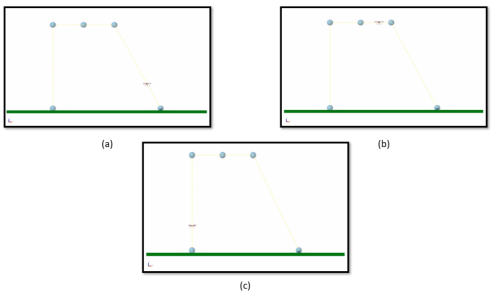


Fig. 5. Pick and Place

V. Conclusion

This research work explored the application of reinforcement learning, specifically the actor-critic method, to control the attitude of a quadcopter in pick and place application. The primary focus was on achieving stable and precise control over the quadcopter's pitch, roll, and yaw angles.

The quadcopter system was trained for 1000, and 2250 episodes. The results from training the quadcopter over different episode lengths provided valuable insights into the effectiveness and areas for improvement of the reinforcement learning-based control system. It was observed that as the number of training episodes increased, the overall

stability and control accuracy of the quadcopter improved. This demonstrates the importance of extended training for RL models to achieve better performance. Through iterative training, these networks learn to optimize the quadcopter's control inputs to achieve desired roll, pitch, and yaw angles. Among the three attitude angles, the yaw angle exhibited larger deviations compared to the pitch and roll angles. This suggests that the current training configuration and reinforcement learning parameters might require further tuning, especially for the yaw control. It is evident that increasing the training episodes could mitigate the yaw angle deviations. Therefore, further extending the training beyond 2250 episodes is a promising approach to enhance the overall control performance of the quadcopter. The actor-critic method successfully enabled the quadcopter to maintain a desired attitude during simulated pick and place maneuvers.

VI. Future Work

The future scope of this project focuses on enhancing the DDPG-based quadcopter attitude control system by increasing the number of training episodes and transitioning to practical implementation. Exploring advanced RL algorithms such as Proximal Policy Optimization (PPO), Twin Delayed DDPG (TD3), or Soft Actor-Critic (SAC) might address the limitations observed in yaw control. Extending the training duration can lead to a more refined and robust control policy, improved stability. Practical implementation involves integrating the control system with physical quadcopter hardware, conducting extensive real-world testing under various conditions, and ensuring robustness and safety.

References

[1] Harkare O, Maan R. "Design and Control of a Quadcopter". International Journal of Engineering and Technical Research. 2021;10(257):05

[2] Etemadi M." Mathematical dynamics, kinematics modeling and PID equation controller of quadcopter". International Journal of Applied Operational Research-An Open Access Journal. 2017 Jan 10;7(1):77-85

[3] Khan M. "Quadcopter flight dynamics". International journal of scientific technology research. 2014 Aug;3(8):130-

[4] M. S. Esmail, M. H. Merzban, A. A. M. Khalaf, H. F. A. Hamed and A. I. Hussein, "Attitude and Altitude Tracking Controller for Quadcopter Dynamical Systems," in IEEE Access, vol. 10, pp. 53344-53358, May 23 2022, doi: 10.1109/ACCESS.2022.3173739

[5] S. -K. Kim and C. K. Ahn,"Auto-Tuner-Based Controller for Quadcopter Attitude Tracking Applications," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 66, no. 12, pp. 2012-2016, Dec. 2019, doi: 10.1109/TCSII.2019.2896591

[6] M. S. Esmail, M. H. Merzban, A. A. M. Khalaf, H. F. A. Hamed and A. I. Hussein,"Attitude and Altitude Nonlinear Control Regulation of a Quadcopter Using Quaternion Representation," in IEEE Access, vol. 10, pp. 5884-5894,January 18 2022, doi: 10.1109/ACCESS.2022.3141544

[7] Liang T, Yang K, Han Q, Li C, Li J, Deng Q, Chen S, Tuo X. "Attitude Estimation of Quadrotor UAV based on QUKF". IEEE Access. 2023 Sep 29

[8] Praveen V, Pillai S." Modeling and simulation of quadcopter using PID controller". International Journal of Control Theory and Applications. 2016 Jan;V 9(15):PP.7151-8

[9] Chan HL, Woo KT." Design and control of small quadcopter system with motor closed loop speed control". International Journal of Mechanical Engineering and Robotics Research. 2015 Oct 1;4(4):287-92.

[10] Najm AA, Ibraheem IK."Nonlinear PID controller design for a 6-DOF UAV quadrotor system. Engineering Science and Technology, an International Journal. 2019 Aug 1;22(4):1087-97

[11] Olaz X, Alaez D, Prieto M, Villadangos J, Astrain JJ. Quadcopter neural controller for take-off and landing in windy environments. Expert Systems with Applications.V 1;PP.225:120146 2023 Sep

[12] Thu KM, Gavrilov AI."Designing and modeling of quadcopter control system using L1 adaptive control". Procedia Computer Science. 2017 Jan 1;103:528-35

[13] Kuantama E, Vesselenyi T, Dzitac S, Tarca R." PID and Fuzzy-PID control model for quadcopter attitude with disturbance parameter". International journal of computers communications control. 2017 Jun 29;12(4):519-32

[14] Heryanto MA, Kusumoputro B." Attitude and Altitude Control of Quadcopter Maneuvers using Neural Network–Based Direct Inverse Control". International Journal of Technology. 2021 Oct;12(4):843

[15] F. Jiang, F. Pourpanah and Q. Hao,"Design, Implementation, and Evaluation of a Neural-Network-Based Quadcopter UAV System," in IEEE Transactions on Industrial Electronics, vol. 67, no. 3, pp. 2076-2085, March 2020, doi: 10.1109/TIE.2019.2905808.

[16] A. Traue, G. Book, W. Kirchg assner and O. Wallscheid,"Toward a Reinforcement Learning Environment Toolbox for Intelligent Electric Motor Control," in IEEE Transactions on Neural Networks and Learning Systems, vol. 33, no. 3, pp. 919-928, March 2022, doi: 10.1109/TNNLS.2020.302957

[17] Zhao J, Yang C, Gao W, Zhou L." Reinforcement learning and optimal control of PMSM speed servo system". IEEE Transactions on Industrial Electronics. 2022 Nov 14;70(8):8305-13.

[18] Pi CH, Hu KC, Cheng S, Wu IC. "Low-level autonomous control and tracking of quadrotor using reinforcement learning". Control Engineering Practice. 2020 Feb 1;V 95:PP.104222

[19] Bernini N, Bessa M, Delmas R, Gold A, Goubault E, Pennec R, Putot S, Sillion F. A few lessons learned in reinforcement learning for quadcopter attitude control. InProceedings of the 24th International Conference on Hybrid Systems: Computation and Control 2021 May 19 (pp. 1-11).

[20] Koch W, Mancuso R, West R, Bestavros A. Reinforcement learning for UAV attitude control. ACM Transactions on Cyber-Physical Systems. 2019 Feb 13;3(2):1-21.