Control of Attitude Dynamics of an Unmanned Aerial Vehicle with Reinforcement Learning Algorithms

Nurten Emer[1*], Necdet Sinan Özbek[2]

[1] Adana Alparslan Türkeş Bilim ve Teknoloji Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Adana, Türkiye (ORCID: 0000-0003-2529-7739)

[2] Adana Alparslan Türkeş Bilim ve Teknoloji Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Adana, Türkiye (ORCID: 0000-0002-7184-9015)

Abstract

In this study, some applications of model-dependent and model-free learning based control techniques are presented for the control of attitude dynamics of vertical take-off and landing unmanned aerial vehicle. Towards this goal, reinforcement learning control algorithms are examined. Control algorithms are discussed and the main differences are presented. A number of numerical simulations are carried out on the attitude control of the system and the results are discussed. Performance evaluation of the proposed learning-based control method has been carried out.

Keywords: Model-free control, Reinforcement Learning Algorithms, Control Application, Stability, Unammaned Aerial Vehicles

Pekiştirmeli Öğrenme Algoritmaları ile İnsansız Hava Aracının Duruş Dinamiklerinin Kontrolü

Özet

Bu çalışmada, dikey kalkış ve iniş yapabilen insansız bir hava aracının duruş dinamiklerinin kontrolü için modele bağlı ve modelden bağımsız öğrenme tabanlı kontrol tekniklerinin bazı uygulamaları sunulmaktadır. Bu amaç için, pekiştirmeli öğrenme kontrol algoritmaları incelenmiştir. Kontrol algoritmaları ele alınmış ve temel farklar sunulmuştur. Sistemin durum kontrolü üzerinde bir takım sayısal benzetimler gerçekleştirilmiş ve sonuçlar tartışılmıştır. Önerilen öğrenmeye dayalı kontrol yönteminin performans değerlendirmesi yapılmıştır.

Anahtar Kelimeler: Modelden bağımsız kontrol, Pekiştirmeli Öğrenme Algoritmaları, Kontrol Uygulamaları, Kararlılık, İnsansız Hava Araçları

* Corresponding Author: Adana Alparslan Türkeş Bilim ve Teknoloji Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, Adana, Türkiye, ORCID: 0000-0003-2529-7739, nurtenemer2505@gmail.com

1. Introduction

Control systems and applications, which act key role in many technological developments, have rapidly evolving dynamics. In many technological development areas, research studies are carried out focusing on the basics such as being safer, being less costly, requiring less personnel power and saving time. Autonomous systems are important products of recent developments. Along with the developing technology, Unmanned Aerial Vehicles (UAV) has recently become a development area that includes these foundations. UAVs can perform the necessary tasks without any onboard operator (Özbek, Önkol, & Efe, 2016) and can be controlled remotely or automatically. UAVs provide advantages over other aircraft owing to the ability to hover in the air, vertical take-off-landing, and superior mobility and maneuverability (Hayat, Yanmaz, & Muzaffar, 2016). Furthermore, UAVs offer a number of benefits such as reduced operating costs and increased safety (Sadraey, 2017). The ability of unmanned aerial vehicles to perform many different tasks, such as target tracking, observation, etc., has enabled them to be used actively in defense industry and civil applications. With these advantages, its popularity has increased in numerous tasks such as industrial, academic, government, military and civilian. They tend to be useful for many civil tasks such as forest fire detection (Lesecq, Gentil, & Daraoui, 2014), security patrol (Zhong, Zhang, Zhang, Zuo, & Zhan, 2018), inspection of power lines (Luque-Vega, Castillo-Toledo, Loukianov, & Gonzalez-Jimenez, 2014), inspection of civil infrastructure, large-scale map making, documentation of archaeological sites (Çömert, Avdan, & Şenkal, 2012), visual inspection-manufacturing, and disaster management.

Control systems are the building blocks of many technological developments. It has been the subject of many articles based on model-based and model free control systems. In most control techniques, it is necessary to create a mathematical model of the system. The correct determination of the dynamics of the system is the basic step in the creation of the mathematical model (Boubakir, Labiod, Boudjema, & Plestan, 2014). However, systems are affected by many different uncertainties and external disturbances. These undesirable disturbances may cause poorly modeled systems and provide inefficient control. Model-free control systems are offered as a solution to eliminate this drawback (Ei, n.d.) (Coelho, Pessôa, Rodrigues Sumar, & Augusto Rodrigues Coelho, 2010).

The rapid development of unmanned aerial vehicles (UAVs) has brought a big impression on many countries to intensify their research and development activities in these areas. This has increased the investments made in the high-safety efficient systems and improved their production capabilities (Papachristos et al., 2018). It is known that the autonomous vehicles can perform their duties, with a number of sub-components such as the payload, battery system, actuators, control system,

navigation system, etc. according to the operational needs (Qi, Li, Ren, Lei, & Yang, 2021).

The decision-making mechanisms of unmanned systems are currently provided via the ground station or the command center. However, the developments of self-deciding drone systems and their active use in the near future are targeted by many countries. With this motivation in mind, this study elaborates a number of learning algorithms for the attitude control of quadrotor type aerial robots. The algorithms are addressed with an in depth discussion.

Furthermore, recent results on model-based and model-free control of UAVs can be summarized as follows. The elaborated control techniques are fuzzy control, artificial neural network-based control, feedback linearization, sliding mode control method, back stepping control method, and model predictive control. Fuzzy control of an unammanned aerial robot is demonstrated in (Santoso, Garratt, & Anavatti, 2020), wherein a PD-type fuzzy algorithm is integrated to follow a predetermined path. Yet another work elaborates high-order sliding mode control (Rezoug, Hamerlain, Achour, & Tadjine, 2015), in which the observer and the controller show satisfactorily good results in the presence of the external disturbances. A further study presents a reinforcement control of an autonomous aerial vehicle. Trajectory planning of load transportation with multi-quadrotors based on reinforcement learning algorithm is proposed in (Li, Zhang, & Han, 2021). An integral sliding mode controller combined with super-twisting algorithm is proposed in (Efe, 2011).

The contributions of this study are given as follows: The reinforcement learning algorithms are discussed with and in-depth assesment. Furthermore, control of the attitude dynamics, which is the core of unmanned aerial vehicle applications, are investigated through learning algorithms. From these aspects, the present research provides a practical guide of reinforcement learning algorithms for unmanned aerial vehicles.

This paper is organized as follows. The vehicle dynamics are presented in section II. The theoretical background and recent studies on the reinforcement learning algorithms are presented in Section III. Performance results for controlling the attitude dynamics are demonstrated in section IV. Finally, the concluding remarks are addressed in the last section.

2. The Vehicle Dynamics

UAVs, which can be designed for different purposes, sizes, and weights, are especially designed for dangerous missions (Loh, Yi Bian, & Roe, 2009). The vehicle control system is the basic unit that manages the UAV. Control system provides an adequate coordination with sub-systems such as actuators, guidance system, battery management system, sensors in the UAV. A case study for the simulation test of the controller is investigated to understand the inner and the outer loop of the UAV (Wang, 2020).

$$\ddot{\phi} = \dot{\theta}\dot{\psi}\left[\frac{I_{yy} - I_{zz}}{I_{xx}}\right] + \frac{J_r}{I_{xx}}\dot{\theta}\Omega_d + \frac{l}{I_{xx}}U_2$$

$$\ddot{\theta} = \dot{\phi}\dot{\psi}\left[\frac{I_{zz} - I_{xx}}{I_{yy}}\right] - \frac{J_r}{I_{yy}}\dot{\phi}\Omega_d + \frac{l}{I_{yy}}U_3 \qquad (1)$$

$$\ddot{\psi} = \dot{\theta}\dot{\phi}\left[\frac{I_{xx} - I_{yy}}{I_{zz}}\right] + \frac{1}{I_{zz}}U_4$$

$$\Omega_d = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4 \qquad (2)$$

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & -b & 0 & b \\ -b & 0 & b & 0 \\ d & -d & d & -d \end{bmatrix}\begin{bmatrix} \Omega_1^2 \\ \Omega_2^2 \\ \Omega_3^2 \\ \Omega_4^2 \end{bmatrix} \qquad (3)$$

$$\dot{X} = f(X,U) \qquad (4)$$

The states and input vectors given in (4) are presented in (5) and (6) (Özbek, Önkol, & Efe, 2014).

$$x_1 = \phi, \ x_2 = \dot{\phi}, \ x_3 = \theta, \ x_4 = \dot{\theta}, \ x_5 = \Psi, \ x_6 = \dot{\Psi},$$
$$x_7 = x, \ x_8 = \dot{x}, \ x_9 = y, \ x_{10} = \dot{y}, \ x_{11} = z, \ x_{12} = \dot{z} \qquad (5)$$

$$f(X,U) = \begin{bmatrix} x_2 \\ a_1 x_4 x_6 - a_2 x_4 \Omega_d + a_3 U_2 \\ x_4 \\ a_4 x_2 x_6 + a_5 x_2 \Omega_d + a_6 U_3 \\ x_6 \\ a_7 x_4 x_2 + a_8 U_4 \\ x_8 \\ \left(c_\phi s_\theta c_\psi + s_\phi s_\psi\right)\frac{1}{m}U_1 \\ x_{10} \\ \left(c_\phi s_\theta s_\psi + s_\phi c_\psi\right)\frac{1}{m}U_1 \\ x_{12} \\ -g + \left(c_\phi c_\theta\right)\frac{1}{m}U_1 \end{bmatrix} \qquad (6)$$

Where the parameter are given as: $a_1 = (I_{yy} - I_{zz})/I_{xx}$, $a_2 = J_r/I_{xx}$, $a_3 = l/I_{xx}$, $a_4 = (I_{zz} - I_{xx})/I_{yy}$, $a_5 = J_r/I_{yy}$, $a_6 = l/I_{yy}$, $a_7 = (I_{xx} - I_{yy})/I_{zz}$, $a_8 = 1/I_{zz}$. As presented in the above literature, the critical part is to control the attitude dynamics of the system. Then, the tracking can be provided in Cartesian space. While determining the control parameters in the process of creating the model, the control system toolbox and trial-error PID tuning method are used (Ozbek, 2019).

3. Reinforcement Learning Algorithms

Reinforcement learning is one of the control methods frequently used in unmanned aerial vehicle systems. In this context, control of the vehicle is control with reinforcement learning (Yoo, Jang, Kim, & Johansson, 2021). In classical-based control methods that are not based on learning, it is necessary to determine the necessary parameters to control the system and to know the exact model of the system. However, it is not possible to accurately obtain the models of the complex

real systems, so the model that is closest to the system behavior is tried to be determined. In this context, the controller parameters obtained in computer simulations cannot provide the system to respond as desired. In cases where the model cannot be precisely known or extracted, learning the movements of the systems allows the system to be controlled. Reinforcement learning methods, which have a learning structure, consist of model-based and model-free algorithms (Zhu, Yao, Liu, Liu, & Liang, 2020).

Reinforcement learning techniques generate a reward or punishment value by considering whether the amount of error made is less or more than the amount of error made in the previous step. According to this calculated reward or penalty score, the system learning is continued and the parameters that will provide the best control are tried to be obtained. As a result of the determination of these parameters, a state action chart will be obtained, this chart will be used to provide control.

The reinforcement learning algorithms are classified into model-based and model-free learning algorithms. To mention a few, Value Iteration model is in the class of model-based algorithms. However, Q-Learning, Sarsa and Sarsa (λ) algorithms do not need a model of the system (Hwangbo, Sa, Siegwart, & Hutter, 2017).

The basic philosophy of the reinforcement learning approaches is to provide a desired result by arranging the reward or punishment progress. The tabulated values of the state vector,

the possible values of the control signal, and the reward values aredetermined for each combination than the control signal that will produce the desired behavior is selected among the states with the highest reward value (Elhaki & Shojaei, 2021). This learning method can be performed both depending on the model and free of the system model. A number of numerical simulations were carried out on the attitude control of the system and the results are discussed.

3.1. Model-based learning algorithms

In the model-based learning method, it is assumed that the environmental model parameters $p(r_{t+1}|s_t, d_t)$ and $p(s_{t+1}|s_t, a_t)$ are known (Polydoros & Nalpantidis, 2017a). Under these assumptions, the optimal value function and rule can be calculated directly using dynamic programming without the need for any model analysis. Once we know the optimal value function, the optimal policy (rule) is to choose the move that maximizes the reward value in the next case.

In cases where system dynamics and model parameters are known, methods that train using the model are widely used. The model-based Value Iteration algorithm tries to determine the most appropriate response in the situations to be encountered by visiting all possible state and motion pairs. When the appropriate V^* value is found in the Value Iteration algorithm, the algorithm will converge to a fixed value. The pseudocode of the method can be given as follows (Polydoros & Nalpantidis, 2017b).

Table 1. Pseudocode for model-based learning

Determine the values V(s) arbitrarly

Do

    $\forall s, s \in S$

    $\forall a, a \in A$

        $Q(s,a) \longleftarrow E[r \,|\, s,a] + \gamma V(s')$

    $V(s) \longleftarrow \max l_a Q(s,a)$

$V(s) < \kappa$

Here, the variable s is the state, and the variable S is the predetermined state space. Further, a is the move in the predetermined move space, $Q(s,a)$ is the reward value corresponding to the a move. V(s) is the value at which Q(s,a) is the largest, V(s') is the highest reward value obtained for the expected situation with the applied movement, and γ is the efficiency value.

A threshold value is set (κ) to determine the convergence of the algorithm, and if the change in reward value between two iterations is less than this threshold value, the most appropriate move for that situation is said to have been

determined. Provided that l is the iteration counter, the convergence formula can be given as follows,

$$\max_{s \in S} \left| V^{l+1}(s) - V^l(s) \right| < \kappa \qquad (7)$$

In the rule iteration method, we save and update the rule directly instead of updating the rule over the reward values. Each iteration of the algorithm takes more time than the value iteration algorithm, but it needs less iteration than the rule iteration value iteration algorithm. The pseudocode of the algorithm is as follows.

Table 2. Pseudocode for modified model-based learning

Start the π rule arbitrarily

Do

  $\pi \longleftarrow \pi'$

  By solving the linear equations to π value.

  $$V^{\pi}(s) = E\big[r \mid s, \pi(s)\big] + \gamma \sum_{s' \in S} P(s' \mid s, \pi(s)) V^{\pi}(s)$$

  Update the rule

  $$\pi'(s) \leftarrow \arg\max\left(E[r \mid s, a] + \gamma \sum_{s' \in S} P(s' \mid s, a) V^{\pi}(s') \right)$$

Under these assumptions, the optimal value function and rule can be calculated directly using dynamic programming without the need for any model analysis. Once we know the optimal value function, the optimal policy (rule) is to choose the move that maximizes the reward value in the next case.

3.2. Model-free learning algorithms

Model-free learning-based control methods play an important role in increasing the control performance of various complex systems. If the results and rewards of the moves are not deterministic, there is a probabilistic distribution for the reward value $p(r_{t+1} \mid s_t, a_t)$. This allows us to model uncontrollable environmental uncertainties in the system. In such a case, the reward definition of the state-action pair can be made as follows,

$$Q(s_t, a_t) = E(r_{t+1}) + \gamma \sum_{s_{t+1}} P(s_{t+1} \mid s_t, a_t) \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) \tag{8}$$

Direct value assignment cannot be made, as different responses may be obtained or different situations can be reached when the same action is applied for the same situation at different times. In this case, the Q-Learning algorithm is used. The update function used in the Q-Learning algorithm is as follows.

$$\hat{Q}(s_t, a_t) \leftarrow \hat{Q}(s_t, a_t) + \eta\left(r_{t+1} + \gamma \max_{a_{t+1}} \hat{Q}(s_{t+1}, a_{t+1}) - \hat{Q}(s_t, a_t)\right) \tag{9}$$

The value of η given in Equation (9) is known as the degree of learning and training is done by gradually decreasing it. The reward matrix $Q(s_t, a_t)$ obtained as a result of the Q-Learning algorithm will be composed of the most appropriate (optimal) values. It is to be noted that the Q-Learning algorithm is an off-policy algorithm, that is, the result of the best rule to be applied in the next step is determined without using it.

As previously stated, the Q-Learning algorithm is an off-policy type method. However, the Sarsa algorithm is obtained by developing the Q-Learning algorithm. Sarsa algorithm is on-policy type, that is, the reward matrix can be calculated by considering the rule to be applied in the next step. In the Sarsa algorithm, instead of looking at all possible moves and choosing the best one, the rule derived from the reward values obtained for the next single move is used and the temporary difference is calculated using the reward value. On-policy methods try to estimate the value of the rule while using the rule to find motion values. The pseudocode of the Sarsa algorithm is as follows,

Table 3. Pseudocode for Sarsa-learning algorithm

Determine the $Q(s, a)$ values arbitrarily

  Determine the s

  Use the ε gready algoirthm to find the apprpriate movement (a) in Q matrix

  Do the iteration

    Apply (a) to the system, observe r and s' values

    Use the ε gready algoirthm to find the apprpriate movement (a') in Q matrix

  Update $Q(s, a)$

  $$Q(s, a) \longleftarrow Q(s, a) + \eta(r + \gamma Q(s', a') - Q(s, a))$$

$s \longleftarrow s'$, $a \longleftarrow a'$

However, in Sarsa (λ), the frequency of visits of previously visited situations is kept in memory and the reward matrix is updated by taking into account the frequency of visits. The pseudocode of the Sarsa (λ) algorithm is as follows.

Table 4. Pseudocode for Sarsa (λ) learning algorithm

Determine the $Q(s,a)$ values arbitrarily

  Determine the s

  Use the ε gready algoirthm to find the appropriate movement (a) in Q matrix

  Do the iteration

    Apply a to the system, observe r and s' values

    Use the ε gready algoirthm to find the appropriate movement (a') in Q matrix

  Update $Q(s,a)$

$$Q(s,a) \longleftarrow Q(s,a) + \eta(r + \gamma Q(s',a') - Q(s,a))$$

$$s \longleftarrow s' , \ a \longleftarrow a'$$

4. Results and Discussion

The controller designed for the vehicle with reinforcement learning is obtained using the Sarsa (λ) learning algorithm. A reference of the vehicle is to keep the vehicle in its stable position around the zero. Angular position errors are given in Figure 1, wherein the results are observed fairly well.
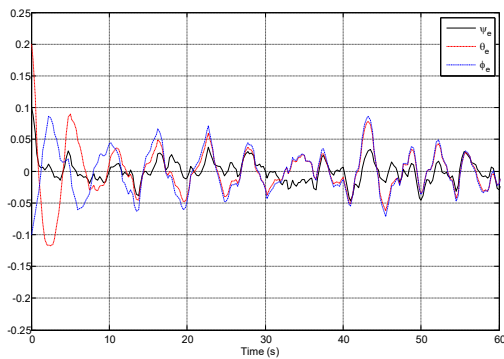

Figure 1. Stabilization of attitude dynamics with reinforcement learning algorithm

The position errors that occur with the sarsa lambda algorithm of the revolving wing system are given in Figure 2. The performance of the controller shows satisfactorily good results to track the reference trajectory in Cartesian space.
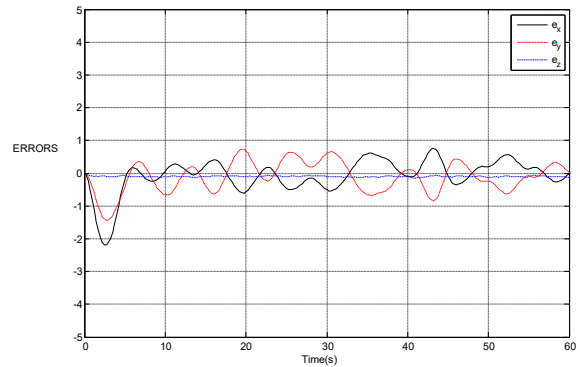

Figure 2. Position errors with reinforcement learning algorithms

5. Conclusions and Recommendations

The main goal of the current research is to discuss the reinforcement learning algorithms for UAV. Furthermore, it is aimed to guide the control researchers on the modelling and control of a vertical take-off and landing aerial robot. With this motivation in mind, recent studies on the learning algorihtms for UAV discussed. Furthermore, the numerical simulations of learning-based algorithms for the attidue control of the UAV are presented. Future works will focus on the design, analysis, and implementation of different model-free algorithms for UAVs.

6. Acknowledgement

References

Boubakir, A., Labiod, S., Boudjema, F., & Plestan, F. (2014). Model-free controller with an observer applied in real-time to a 3-DOF helicopter. *Turkish Journal of Electrical Engineering*

and Computer Sciences, *22*(6), 1564–1581. https://doi.org/10.3906/elk-1204-54

Coelho, L. dos S., Pessôa, M. W., Rodrigues Sumar, R., & Augusto Rodrigues Coelho, A. (2010). Model-free adaptive control design using evolutionary-neural compensator. *Expert Systems with Applications*, *37*(1), 499–508. https://doi.org/10.1016/j.eswa.2009.05.042

Çömert, R., Avdan, U., & Şenkal, E. (2012). *İnsansiz hava araçlarinin kullanim alanlari ve gelecekteki beklentiler*. 16–19.

Efe, M. Ö. (2011). Integral sliding mode control of a quadrotor with fractional order reaching dynamics. *Transactions of the Institute of Measurement and Control*, *33*(8), 985–1003. https://doi.org/10.1177/0142331210377227

Ei, G. R. (n.d.). *Model-free control of dc / dc converters*.

Elhaki, O., & Shojaei, K. (2021). A novel model-free robust saturated reinforcement learning-based controller for quadrotors guaranteeing prescribed transient and steady state performance. *Aerospace Science and Technology*, *119*, 107128. https://doi.org/10.1016/j.ast.2021.107128

Hayat, S., Yanmaz, E., & Muzaffar, R. (2016). Survey on Unmanned Aerial Vehicle Networks for Civil Applications: A Communications Viewpoint. *IEEE Communications Surveys and Tutorials*, *18*(4), 2624–2661. https://doi.org/10.1109/COMST.2016.2560343

Hwangbo, J., Sa, I., Siegwart, R., & Hutter, M. (2017). Control of a Quadrotor With Reinforcement Learning. *IEEE Robotics and Automation Letters*, *2*(4), 2096–2103. https://doi.org/10.1109/LRA.2017.2720851

Lesecq, S., Gentil, S., & Daraoui, N. (2014). Quadrotor attitude estimation with data losses. *2009 European Control Conference, ECC 2009*, 3851–3856. https://doi.org/10.23919/ecc.2009.7075000

Li, X., Zhang, J., & Han, J. (2021). Trajectory planning of load transportation with multi-quadrotors based on reinforcement learning algorithm. *Aerospace Science and Technology*, *116*, 106887. https://doi.org/10.1016/j.ast.2021.106887

Loh, R., Yi Bian, & Roe, T. (2009). UAVs in civil airspace: Safety requirements. *IEEE Aerospace and Electronic Systems Magazine*, *24*(1), 5–17. https://doi.org/10.1109/MAES.2009.4772749

Luque-Vega, L. F., Castillo-Toledo, B., Loukianov, A., & Gonzalez-Jimenez, L. E. (2014). Power line inspection via an unmanned aerial system based on the quadrotor helicopter. *Proceedings of the Mediterranean Electrotechnical Conference - MELECON*, (April), 393–397. https://doi.org/10.1109/MELCON.2014.6820566

Ozbek, N. S. (2019). An Evaluation of Model-Free Control Strategies for Quadrotor Type Unmanned Aerial Vehicles. *2019 International Conference on Applied Automation and Industrial Diagnostics (ICAAID)*, 1–6. https://doi.org/10.1109/ICAAID.2019.8935001

Özbek, N. S., Önkol, M., & Efe, M. Ö. (2014). Dönerkanat Tipinde Bir İnsansız Hava Aracının Farklı Yöntemlerle Kontrolü ve Performans Analizi. *Otomatik Kontrol Ulusal Toplantısı*, 11–13. Kocaeli.

Özbek, N. S., Önkol, M., & Efe, M. Ö. (2016). Feedback control strategies for quadrotor-type aerial robots: a survey. *Transactions of the Institute of Measurement and Control*, *38*(5), 529–554. https://doi.org/10.1177/0142331215608427

Papachristos, C., Dang, T., Khattak, S., Mascarich, F., Khedekar, N., & Alexis, K. (2018). Modeling, Control, State Estimation and Path Planning Methods for Autonomous Multirotor Aerial Robots. *Foundations and Trends in Robotics*, *7*(3), 180–250. https://doi.org/10.1561/2300000058

Polydoros, A. S., & Nalpantidis, L. (2017a). Survey of Model-Based Reinforcement Learning: Applications on Robotics. *Journal of Intelligent & Robotic Systems*, *86*(2), 153–173. https://doi.org/10.1007/s10846-017-0468-y

Qi, D., Li, Z., Ren, B., Lei, P., & Yang, X. (2021). Detection and Tracking of a Moving Target for UAV Based on Machine Vision. *2021 7th International Conference on Control, Automation and Robotics (ICCAR)*, 173–178. https://doi.org/10.1109/ICCAR52225.2021.9463501

Rezoug, A., Hamerlain, M., Achour, Z., & Tadjine, M. (2015). Applied of an adaptive Higher order sliding mode controller to quadrotor trajectory tracking. *2015 IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, 353–358. https://doi.org/10.1109/ICCSCE.2015.7482211

Sadraey, M. (2017). Unmanned Aircraft Design: A Review of Fundamentals. *Synthesis Lectures on Mechanical Engineering*, *1*(2), i–193. https://doi.org/10.2200/S00789ED1V01Y201707MEC004

Santoso, F., Garratt, M. A., & Anavatti, S. G. (2020). Hybrid PD-Fuzzy and PD Controllers for Trajectory Tracking of a Quadrotor Unmanned Aerial Vehicle: Autopilot Designs and Real-Time Flight Tests. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 1–13. https://doi.org/10.1109/TSMC.2019.2906320

Wang, L. (2020). PID Control of Multi-rotor Unmanned Aerial Vehicles. In *PID Control System Design and Automatic Tuning using MATLAB/Simulink* (pp. 305–326). https://doi.org/10.1002/9781119469414.ch10

Yoo, J., Jang, D., Kim, H. J., & Johansson, K. H. (2021). Hybrid Reinforcement Learning Control for a Micro Quadrotor Flight. *IEEE Control Systems Letters*, *5*(2), 505–510. https://doi.org/10.1109/LCSYS.2020.3001663

Zhong, Y., Zhang, Y., Zhang, W., Zuo, J., & Zhan, H. (2018). Robust Actuator Fault Detection and Diagnosis for a Quadrotor UAV with External Disturbances. *IEEE Access*, *6*, 48169–48180. https://doi.org/10.1109/ACCESS.2018.2867574

Zhu, P., Yao, S., Liu, Y., Liu, S., & Liang, X. (2020). Autonomous Reinforcement Control of Underwater Vehicles based on Monocular Depth Vision. *IFAC-PapersOnLine*, *53*(2), 9201–9206. https://doi.org/10.1016/j.ifacol.2020.12.2186.