International Journal of Sciences and Innovation Engineering
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
website: www.ijsci.com
Vol.03 No.01
E-ISSN : 3049-0251
DOI: https://doi.org/10.70849/ijsci0301260586711

Vertical Movement Control and Kalman Filter Design for a VTOL UAV

Mehmet Karahan[1], Muhammed Halil Guler[2], Berkin Kirmizigul[3], Emre Ulucay[4], Aydin Eralp Sirmali[5]
[1,2,3,4,5] *TOBB University of Economics and Technology, Electrical and Electronics Engineering Department, Engineering Faculty, Cankaya, Ankara, Turkiye.*

Article Info

Article History:

Published:21 Jan 2026

Publication Issue:
Volume 3, Issue 01
January-2026

Page Number:
501-513

Corresponding Author:
Mehmet Karahan

Abstract:
Vertical take-off and landing (VTOL) unmanned aerial vehicles (UAVs) are widely used today. Their ability to take off and land in difficult terrain without needing a runway, their ability to hover, and their ability to rotate around their own axis are the most important features that distinguish them from fixed-wing UAVs. For VTOL UAVs to perform take-off and landing maneuvers smoothly and reach given reference values, controller design and Kalman filter design are of great importance. In this research, a proportional integral derivative (PID) controller for altitude control of a VTOL UAV was developed, and a Kalman filter was designed for speed and position estimation. Modeling and simulations were performed using the MATLAB Simulink program.
Keywords: autonomous unmanned aerial vehicle, vertical take-off and landing, Kalman filter, PID control

1. Introduction

Vertical take-off and landing (VTOL) unmanned aerial vehicles (UAVs) have numerous advantages over fixed-wing UAVs (Rehan et al., 2022). They do not require a runway for take-off. Unlike fixed-wing UAVs, they have the ability to hover in the air (Hadi et al., 2015). They can also rotate around their own axis in the air (Panigrahi et al., 2021). Their ability to hover makes them more advantageous in terms of payload carrying (Ahmed Snikdha & Chen, 2023). Furthermore, their hovering capability allows them to capture more stable images and videos (Karahan, Kurt & Kasnakoglu, 2022). VTOL UAVs are used for a wide variety of purposes, including surveillance, firefighting, agricultural spraying, aerial photography, cargo transport, mining, and wildlife observation (Karahan, 2025). VTOL UAVs are also widely used in search and rescue operations (Lyu et al., 2023). Advancing technology has enabled VTOL UAVs to make progress in obstacle avoidance, trajectory planning, and autonomous driving (Karahan & Kasnakoglu, 2021). Advancing machine learning technologies enable VTOL UAVs to more easily detect people and objects being searched for (Karahan et al., 2021).

This study addresses the vertical movement control and Kalman filter design of a VTOL UAV. Modeling studies for this work were performed using MATLAB Simulink. Thanks to the Kalman filter, the system's states can be predicted from input and output data, along with the model's prior information. The designed Kalman filter was used to predict the speed and position of the VTOL UAV. Numerous simulations were performed regarding the VTOL UAV's speed, position, and predicted states using the Kalman filter. These simulations confirmed the successful operation of the controller design and the Kalman filter.

2. Literature Review

Numerous researchers have conducted studies on the control of VTOL UAVs. Ilyas et al. (2025) designed a Backstepping Controller and a Super Twisting Sliding Mode controller for a VTOL UAV and compared their performance. They observed that the backstepping controller followed the references with 5% less error. Daud Filho & Bello (2024) numerically simulated the transition of a VTOL UAV from hovering to cruising and from cruising to hovering. They also designed an LQR controller for the VTOL UAV. Deng Wu & You (2021) developed a 6-degree-of-freedom model of a VTOL UAV. They subsequently developed an extended-state observer-based PID controller for the VTOL UAV. Pokrovska and Goetzendorf-Grabowski (2025) developed a three-degree-of-freedom model of an electric VTOL UAV. They then designed an LQR controller for this model.

There are numerous scientists who have designed Kalman filters for VTOL UAVs. Han et al. (2022) designed a second-order Kalman filter to detect inertial measurement unit (IMU) sensor failure in a VTOL UAV. Ambroziak et al. (2022) designed an extended Kalman filter that enables precise positioning of a hybrid VTOL UAV relative to the landing strip, allowing it to take off and land from a moving ship. Cohen & Forbes (2020) designed an Invariant Extended Kalman Filter (IEKF) to estimate the navigation states, sensor deviations, and wind speed of a VTOL UAV. Patan & Ustaoglu (2026) have developed a robust augmented-state extended Kalman filter (RASEKF) for sensor and actuator fault detection and isolation (FDI) for a VTOL UAV.

3. Case and Methodology

In this section, the nonlinear differential equations describing the vertical motion of the UAV are modeled in the MATLAB/Simulink environment. The aim is to observe the physical behavior of the system in a virtual environment and to create a test environment for the Kalman filter and controller algorithms that will be designed in subsequent stages.

Before starting the system simulation, the equilibrium input value u_{eq}, which ensures the UAV remains suspended in the air (hover state), was calculated. At equilibrium, the vertical acceleration $\ddot{z}$ and velocity $\dot{z}$ are assumed to be zero.

The equation of motion of the system is given in equation (1).

$$m\ddot{z} = k_u u^2 - B\dot{z} - mg \tag{1}$$

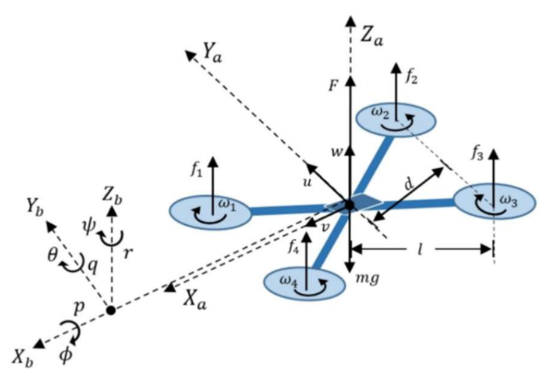
In the equilibrium state ($\ddot{z} = 0$ and $\dot{z} = 0$), the equation takes the following form:

$$0 = k_u u_{eq}^2 - mg \tag{2}$$

Calculation of the balance input value (u_{eq}):

$$u_{eq} = \sqrt{\frac{mg}{k_u}} \tag{3}$$

Figure 1
Schematic and Free Body Diagram (FBD) of the UAV



The physical parameters of the UAV used in this project are summarized in Table 1.

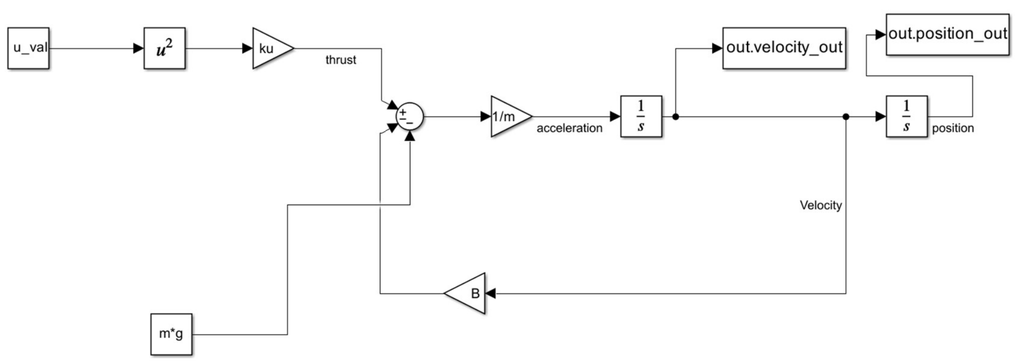
Table 1
Physical parameters of the UAV

Parameter	Symbol	Value	Unit	Description
Mass	m	2.5	kg	Total weight of the UAV
Gravity	g	9.81	m/s^2	Gravitational acceleration
Thrust coefficient	k_u	30	N	Lift force constant
Damping coefficient	B	0.5	Ns/m	Aerodynamic friction/drag
Equilibrium input	u_{eq}	0.9042	-	Control input required for hover
Arm length	l	0.25	m	Distance from center to motors (Estimated)
Inertia (z-axis)	I_{zz}	0.05	$kg.m^2$	Rotational inertia (Estimated)

When the parameters used in the project ($m = 2.5$ kg, $g = 9.81$ m/s², $k_u = 30$ N) were substituted, the theoretical equilibrium input u_{eq} was found to be 9042. This value will be used as a reference input in simulations to prevent the system from initially dropping or rising disproportionately.

In the next stage, the Simulink model of the system was created. Since the system is second-order, two consecutive integrator blocks were used to obtain velocity and position information. A Math Function was used to square the control input u, and a Gain (k_u) block was used to generate the thrust force. The impulse force (positive direction), the gravitational force (mg, negative direction), and the friction force ($B\dot{z}$) are combined in a summation block to obtain the net force. The net force was divided by the system mass (1/m) to obtain the acceleration, which was then fed to the integrator input. The vertical position and velocity model of the UAV is given in Figure 2.

Figure 2
UAV vertical position and velocity model



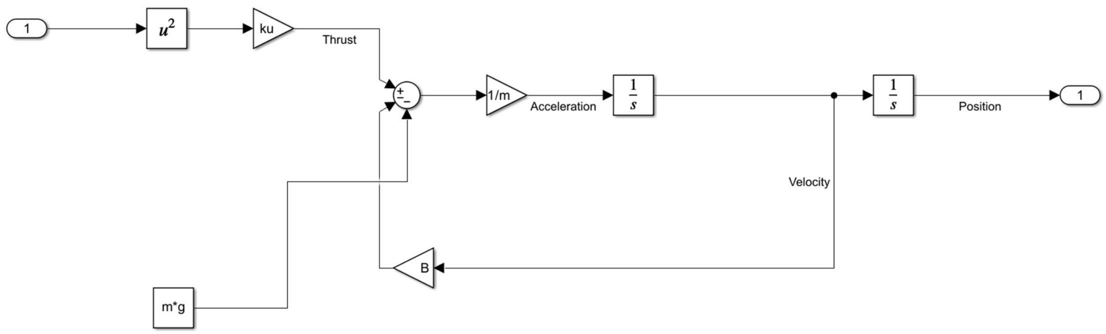
The vertical motion dynamics of the UAV are nonlinear, particularly due to the quadratic nature of the thrust force. However, modern control methods such as the Kalman Filter and LQR require linear system models. Therefore, a linear state-space model was obtained by deriving the Jacobian matrices of the system around the previously calculated equilibrium point (hover state).

The linear model of the system can be shown as follows:

$$\dot{x} = Ax + Bu \quad (4)$$
$$y = Cx + Du \quad (5)$$

The linearized UAV model created in Simulink is shown in Figure 3.

Figure 3
Linearized UAV model



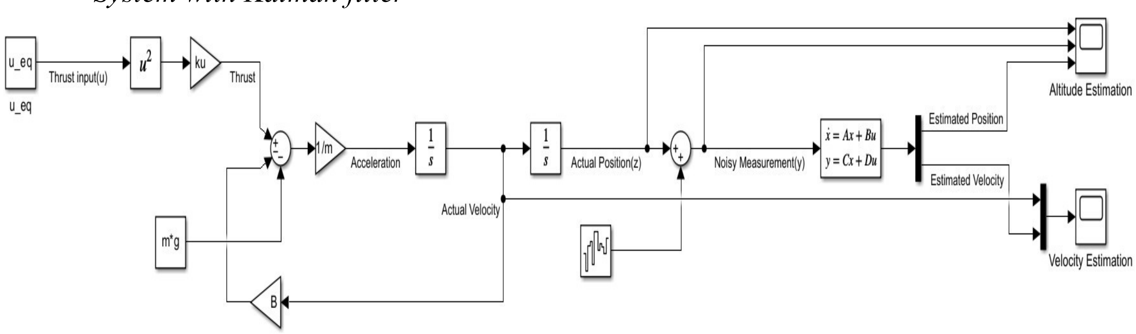
In real-world applications, sensors always contain noise, and not all state variables (especially speed) can be measured directly. In this research, a Kalman filter was designed to filter out the noise affecting the altitude sensor (barometer/GPS) of the UAV and to estimate the unmeasurable vertical speed.

The LQE (Linear Quadratic Estimator) method was used in the filter design, and the following covariance matrices were defined: Process Noise Covariance (Q) represents the uncertainty in the system model. In this research, Q = I = 2×2 × 0.1 was chosen. Measurement Noise Covariance (R) represents the error margin of the sensor. In this research, R = 0.01 was chosen for the initial design. The gain of the Kalman filter was calculated as follows.

$$K = \begin{matrix} 3.8608 \\ 2.4528 \end{matrix} \tag{6}$$

The designed filter was tested under Band-Limited White Noise, which was added to the system in the Simulink environment. The Simulink model with the Kalman filter added is presented in Figure 4.

Figure 4
System with Kalman filter



The aim was then to ensure that the unmanned aerial vehicle (UAV) tracked its altitude along the vertical axis safely, stably, and with damping to a predetermined reference value. With the designed controller, the system's gravitational effect was compensated, permanent error was eliminated, and closed-loop stability was ensured.

The UAV's vertical motion was modeled by the following differential equation:

$$\mathrm{m}\ddot{\mathrm{z}} = \mathrm{T} - \mathrm{mg} - \mathrm{B}\dot{\mathrm{z}} \tag{7}$$

The parameters here can be explained as follows: z is altitude (m), T is total thrust force (N), m is vehicle mass (kg), g is gravity, and B is the linear friction coefficient.

Error and speed feedback for altitude control are defined as in equations (8) and (9).

$$e_z = z_{ref} - z \tag{8}$$

$$e_v = -\dot{z} \tag{9}$$

Here, position error represents the difference between the reference altitude and the actual altitude; velocity error represents the negative feedback of velocity to ensure system damping (Karahan, 2024).

An acceleration command in the PID structure is generated using error signals. Thanks to this structure:

- The proportional term ensures the system quickly returns to the reference,
- The derivative term reduces overshoot,
- The integral term eliminates the permanent error.

Using these error equations, an acceleration command in the PID structure given in equation (10) was generated (Karahan, Kasnakoglu, & Akay, 2023).

$$a_{cmd} = K_p e_z + K_i \int e_z \, dt + K_d e_v \tag{10}$$

Here, K_p, K_i, and K_d are the proportional, integral and derivative gains, respectively (Karahan & Kasnakoglu, 2021).

Gravity has a dominant effect on the UAV's vertical movement. The term "feedforward" is used to compensate for this effect directly, rather than indirectly, by the controller. The total thrust command is calculated as follows:

$$T_{cmd} = m(g + a_{cmd}) \tag{11}$$

Thanks to this approach, the system can remain stationary in the air by balancing gravity even in a zero-error state. Thus, the controller is only responsible for the dynamic behavior.

The engine and thrust equation is given in equation (12).

$$T = k_u u^2 \tag{12}$$

The engine command is found from here as follows.

$$u_{cmd} = \sqrt{\frac{T_{cmd}}{k_u}} \tag{13}$$

Due to physical limitations, the thrust is restricted to the following range:

$$0 \le T_{cmd} \le T_{max} \tag{14}$$

T_{max} in Equation (14) is equal to 3mg. To prevent the integral term from becoming excessively large, the integral output is limited.

$$\xi_{min} \le \int e_z \, dt \le \xi_{max} \tag{15}$$

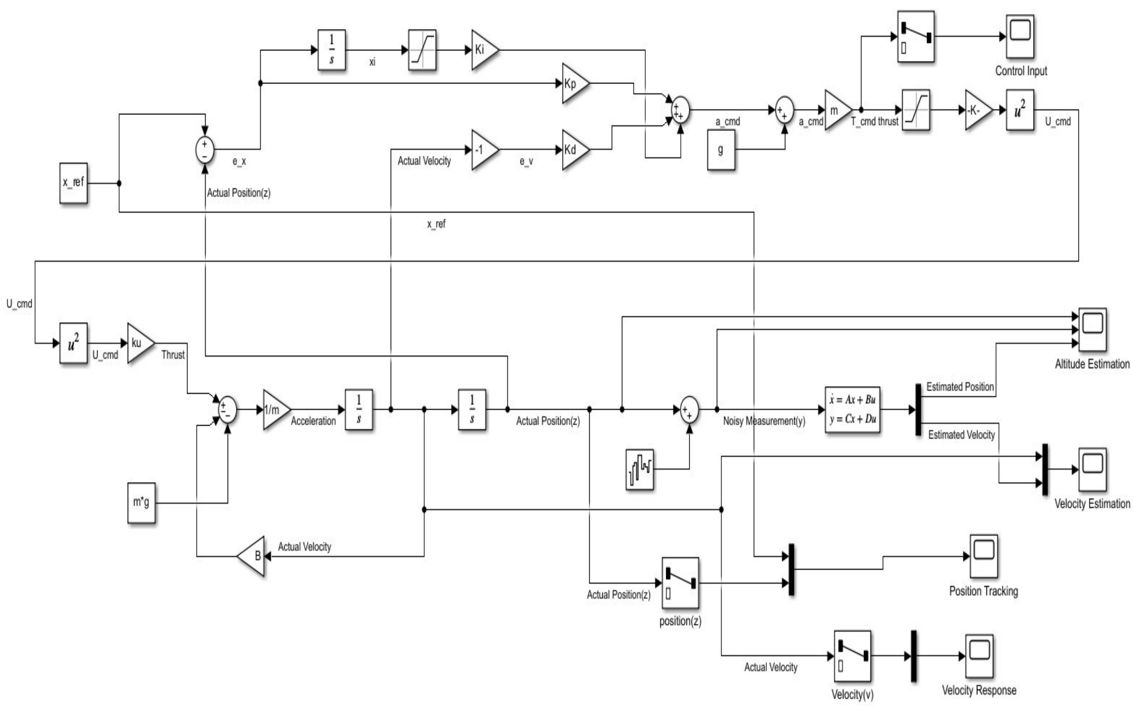
The control parameters used in the equations mentioned above are given in Table 2.

Table 2
Control parameters

Term	Value
z_{ref}	10 m
K_{p}	0.6
K_i	0.08
K_d	1.6
ξ_{min}	-20
ξ_{max}	20
T_{max}	73.575

The Simulink block diagram showing the system's position and velocity simulations is shown in Figure 5. Actual velocity and position data were used in the scopes. The graphs generated when pressing "position tracking" and "velocity tracking" are evaluated in the Results section.

Figure 5
UAV control model block diagram

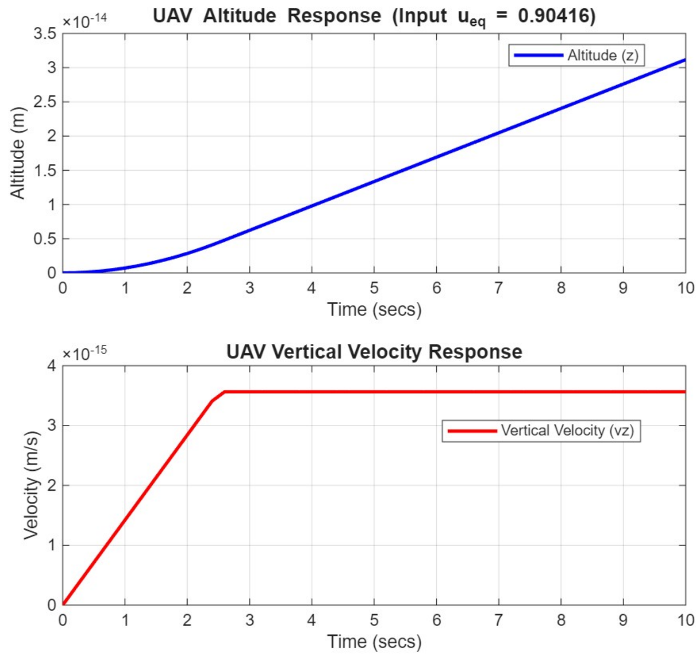


4. Results & Analysis

In this section, simulation results are analyzed. First, the Simulink model given in Figure 1 was run to obtain the following UAV altitude response and UAV vertical velocity response graphs.

Figure 6

UAV altitude response and UAV vertical velocity response



The y-axis multiplier of the altitude graph is on the order of $10^{(-14)}$. The change of approximately $3\times10^{(-14)}$ meters seen at the end of the simulation is not physically significant. This value is very small and negligible.

Looking at the vertical velocity, the graph is seen to be on the order of $10^{(-15)}$. This can be taken as zero.

In conclusion, the simulations clearly demonstrate that the thrust force applied to the system net counteracts the gravitational force. The graphs prove the consistency of our mathematical model and the accuracy of our equilibrium point. As expected, the system maintained its 'hover' state.

When simulating the linearized system with the added Kalman filter, the following graphs are obtained. The position and velocity graphs obtained for R = 0.01 are presented in Figure 7 and Figure 8.

Figure 7

Kalman filter position simulation for R=0.01

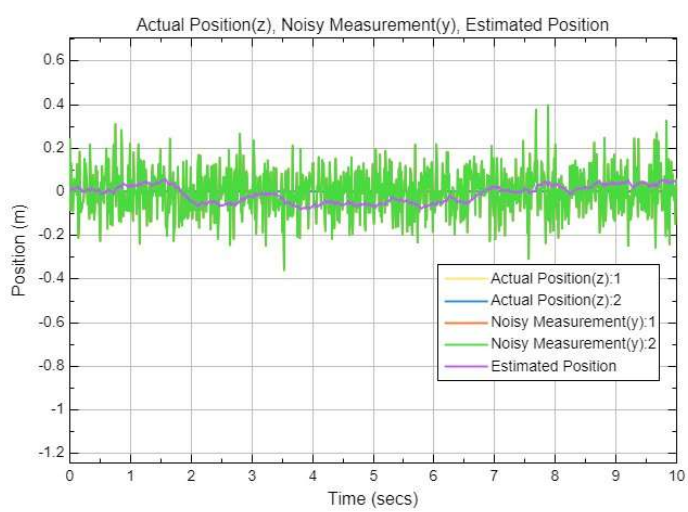


Figure 8
Kalman filter velocity simulation for R=0.01

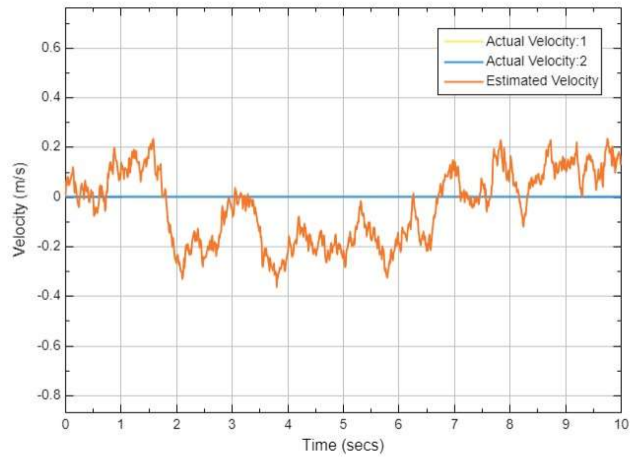


Figure 7 shows the altitude (position) estimate. The green line in the graph shows the sensor data distorted by artificial noise (white noise) added to the system. As seen, the measurement signal has high-frequency distortions with an amplitude of ± 0.2 meters. This is a very good result. The purple line (Kalman filter output) processed this noisy data and produced a very good estimate. The filter did not succumb to the instantaneous changes in noise and successfully tracked the true equilibrium position of the system (0 meters). This indicates that the selected R = 0.01 and Q = 0.1 values are suitable for the system (for a balanced system).

Figure 8 shows the velocity estimation. The system does not have a physical sensor measuring vertical velocity. Despite this, the Kalman filter successfully predicted the velocity using the system's mathematical model. The orange line shows the filter's velocity prediction. The actual velocity (blue line) is zero (at equilibrium). It is expected that the predicted velocity will oscillate around zero because the filter detects the velocity from the derivative variations of the noisy position data. The average value of the prediction is 0 m/s, and the information that the system has stopped (hover) has been successfully generated.

For an R value of 10000, these graphs were reinterpreted to examine whether the filter was working.

Figure 9
Kalman filter position simulation for R=10000

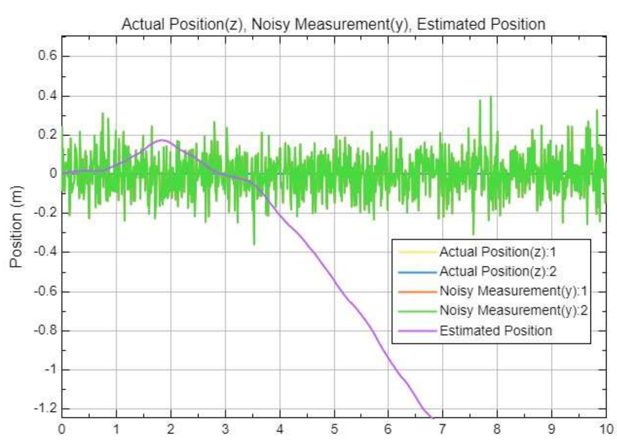


Figure 10
Kalman filter velocity simulation for R=10000

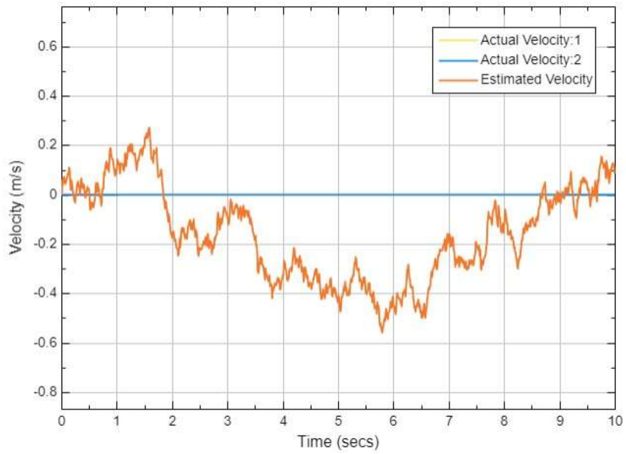


Figure 9 shows the altitude (position) estimate. As seen in the figure, the measurement noise covariance matrix R is set to a very high value, such as 10000. In this case, the Kalman filter minimizes its reliance on the data from the sensor (green line). The estimated position (purple line) does not respond to any of the high-frequency changes in the sensor noise. As a result, the noise suppression success is at its maximum level, but since the filter has become very insensitive to the measurements, the system's ability to capture sudden changes that may occur in real-world conditions (bandwidth) is significantly reduced. This situation is called "Over-filtering".

Figure 10 shows the velocity estimate. Compared to the velocity graph in the steady state (R = 0.01), the velocity estimate in this scenario (orange line) shows much slower changes. As a result, the filter does not take into account instantaneous changes in the measurement data, and it also behaves very slowly when updating the velocity estimate. The velocity estimate oscillates very slowly around zero. These graphs have shown us how a high R value distorts the system response. This is what was expected; our filter is working correctly.

In the next stage, these graphs were interpreted for R = 0.000001 and the performance of the Kalman filter was observed.

Figure 11
Kalman filter position simulation for R=0.000001

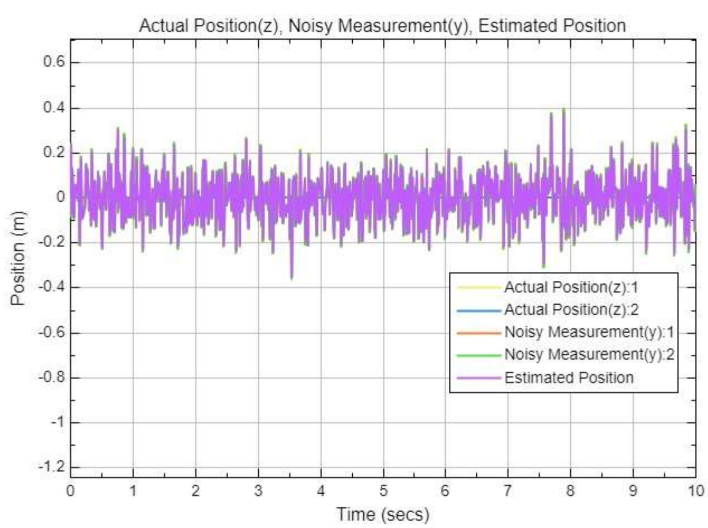


Figure 12
Kalman filter velocity simulation for R=0.000001

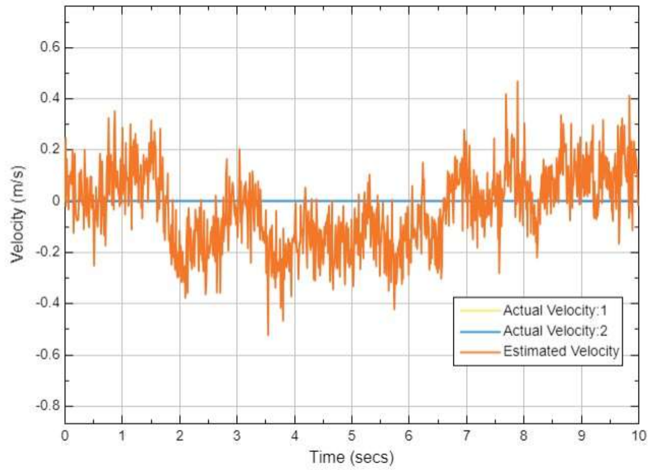


Figure 11 shows the altitude (position) estimation graph. The purple line (Estimation) in the left graph almost perfectly matches the green line (Noisy Measurement) in the background. The reason the green line is not visible is that the purple line follows it very well. Since the R value was chosen close to zero, the Kalman filter was given the information that the sensor was perfect. In return, the filter accepted every piece of data from the sensor (even the noise) as "real movement" and kept itself the same as the measurement. As a result, the filtering feature was completely disabled. This situation is called "Under-filtering".

Figure 12 shows the velocity estimate. The orange line (Estimated Velocity) is noisy and fluctuates randomly within the ± 0.4 range. Velocity is the derivative of position (rate of change). Because the filter mistakes instantaneous noise jumps in position for "real motion," it calculates high velocity changes by taking the derivative of these jumps. As a result, the noise has been further amplified by the derivative process, making the velocity estimate unusable.

In general, choosing R too low has excessively increased the Kalman Gain (L). This causes the filter to directly pick up and even amplify the noise instead of suppressing it. It can be said that our filter is working properly.

In the next stage, simulations of the UAV's position tracking and velocity behavior were performed for 20 seconds.

Figure 13
Position Tracking Graph

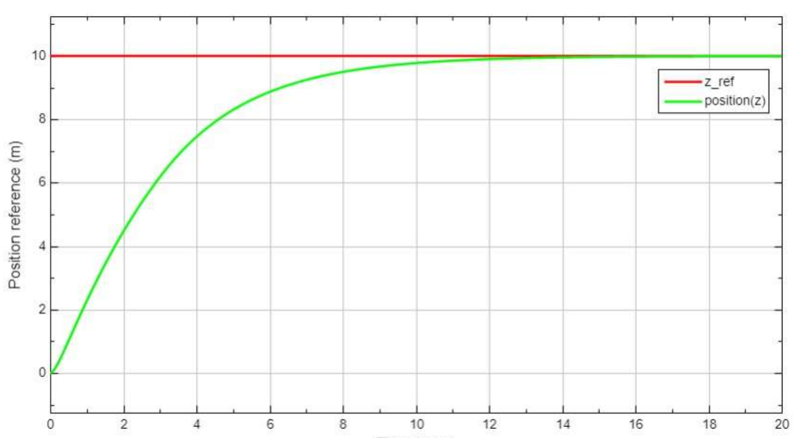
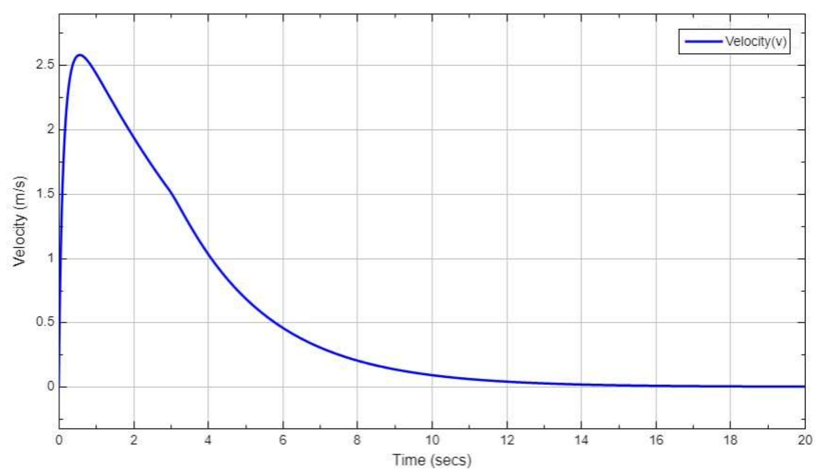


Figure 13 shows the altitude (position) tracking result obtained with a simulation duration of 20 seconds. In the graph, the yellow line shows the reference altitude of the system (z_ref = 10 m), and the blue line shows the actual altitude response of the UAV. At the initial time, the vehicle is at ground level (z = 0), and it appears that the altitude value is increasing monotonically with the effect of the controller. As seen in the graph, the system approaches the reference altitude (10 m) without exceeding it. This clearly shows that the system exhibits a controlled ascent without oscillation thanks to the damping effect provided by the derivative term. The rapid increase in altitude, especially in the first few seconds, reveals that the proportional gain was chosen appropriately for the system dynamics. It is observed that the altitude value approaches the reference value quite closely between approximately 10-12 seconds and settles completely at the reference altitude around the 20th second. This behavior shows that the integral term comes into play and eliminates the permanent error over time. By limiting the integral gain, the system did not experience wind-up and stable convergence was achieved. Furthermore, it was observed that the system remained within physical limits throughout the simulation period and the control input did not remain in saturation for an extended period. Implementing gravity compensation with a feedforward structure allowed the controller to correct only dynamic errors, resulting in a smoother transition.

In conclusion, the position tracking graph presented in Figure 13 clearly demonstrates that the designed PID-based altitude controller exhibits stable, overshoot-free, and persistently error-free performance. These results confirm that the controller design is suitable and effective for the altitude tracking problem.

Figure 14 shows the velocity behavior graph.

Figure 14
Velocity Behavior Graph



As seen in the graph, the speed increases rapidly immediately after the initial moment and reaches a maximum value of approximately 2–3 m/s. This behavior can be explained by the proportional control effect becoming dominant due to the initially large altitude error. At this stage, the system's rapid ascent acceleration improves altitude tracking performance.

In the subsequent time interval, the speed decreases monotonically as the altitude error decreases. This is a result of the damping effect of the derivative term and gravity on the system. The fact that the speed does not drop to negative values and does not oscillate indicates that the controller gains are appropriately adjusted and the system avoids overreaction.

In approximately 14–20 seconds, the speed value approaches zero and the system becomes stable. This shows that the UAV ends its vertical movement after reaching the target altitude and remains suspended at a constant altitude. The smooth approach of the speed to zero indicates that there is no sudden braking or vibration in the system. In conclusion, the velocity behavior presented in Figure 14 demonstrates that the designed altitude controller produces a stable, smooth transition and physically realistic dynamic response. The peak value of the velocity profile remained limited, the transient regime was fast but controlled, and the system reached zero velocity in the steady state. These results confirm that a consistent and balanced control performance was achieved between position tracking and velocity dynamics.

5. Conclusion

In this research, the PID-based control structure designed for altitude control of the UAV was modeled in the Simulink environment, and the simulation results were examined in detail. The results obtained are as follows: The system follows the reference altitude stably and without overshoot. The velocity response remained limited and exhibited damped behavior. The thrust signal remained within saturation limits, and physical realism was preserved. Permanent error was eliminated thanks to the integral term, and altitude tracking was ensured accurately. In conclusion, it was observed that the designed altitude controller offers a suitable, stable, and reliable solution for UAV systems.

Funding Details

This research has no external funding.

Disclosure Statement

The authors have no relevant financial or non-financial interests to disclose

References

1. M. Rehan, F. Akram, A. Shahzad, T.A. Shams, Q. Ali, Vertical take-off and landing hybrid unmanned aerial vehicles: An overview. The Aeronautical Journal, 2022, Vol. 126, Issue 1306, pp. 2017-2057.
2. Hadi, G. S., Kusnaedi, M. R., Dewi, P., Budiyarto, A., & Budiyono, A. (2015). Design of avionics system and control scenario of small hybrid vertical take-off and landing (VTOL) UAV. The Journal of Instrumentation, Automation and Systems, Vol. 2, Issue 2, pp. 66-71.
3. S. Panigrahi, Y. S. S. Krishna, & A. Thondiyath, Design, Analysis, and Testing of a Hybrid VTOL Tilt-Rotor UAV for Increased Endurance. Sensors, 2021, Vol. 21, Issue 18, 5987.
4. S. S. Ahmed Snikdha, & S.-H. Chen, A Computational Investigation of the Hover Mechanism of an Innovated Disc-Shaped VTOL UAV. Drones, 2023, Vol. 7, Issue 2, 105.
5. M. Karahan, H. Kurt and C. Kasnakoglu, Moving Object Detection and Counting in Traffic with Gaussian Mixture Models and Vehicle License Plate Recognition with Prewitt Method, 2022 International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Ankara, Turkey, 2022, pp. 32-36.
6. M. Karahan, Development Stages of Quadrotors from Past to Present: A Review. Drones, 2025, Vol. 9, Issue 12, 840.

7. M. Lyu, Y. Zhao, C. Huang, & H. Huang, (2023). Unmanned aerial vehicles for search and rescue: A survey. Remote Sensing, 2023, Vol. 15, Issue 13, 3266.
8. M. Karahan, & C. Kasnakoglu, Path planning and collision avoidance with artificial intelligence for a quadrotor UAV. In 2021 International Conference Automatics and Informatics (ICAI), 2021, pp. 163-166.
9. M. Karahan, F. Lacinkaya, K. Erdonmez, E. D. Eminagaoglu, C. Kasnakoglu, Face detection and facial feature extraction with machine learning. In International conference on intelligent and fuzzy systems, 2021, pp. 205-213.
10. M. Ilyas, A. Khan, Z. Ahmed, D. -W. Jung & R. Khan, Altitude Stabilization of VTOL System With Nonlinear Control Strategy, IEEE Access, 2025, Vol. 13, pp. 41300-41308.
11. A. C. Daud Filho & E. M. Belo, A tilt-wing VTOL UAV configuration: Flight dynamics modelling and transition control simulation, The Aeronautical Journal, 2024, Vol. 128, Issue. 1319, pp. 152–177.
12. Z. Deng, L. Wu, & Y. You, Modeling and Design of an Aircraft-Mode Controller for a Fixed-Wing VTOL UAV. Mathematical Problems in Engineering, 2021, Vol. 2021, Issue 1, 7902134.
13. K. Pobikrowska, T. Goetzendorf-Grabowski, VTOL UAV optimal control in the transition phase, Aircraft Engineering and Aerospace Technology: An International Journal, 2025, Vol. 97, Issue. 9, pp. 1085–1094,
14. X. Han, Y. Hu, A. Xie, X. Yan, X. Wang, C. Pei, & D. Zhang, Quadratic-Kalman-filter-based sensor fault detection approach for unmanned aerial vehicles. IEEE Sensors Journal, 2022, Vol. 22, Issue, 19, pp.18669-18683.
15. L. Ambroziak, M. Ciężkowski, A. Wolniakowski, S. Romaniuk, A. Bożko, D. Ołdziej, & C. Kownacki, Experimental tests of hybrid VTOL unmanned aerial vehicle designed for surveillance missions and operations in maritime conditions from ship-based helipads. Journal of Field Robotics, 2022, Vol. 39, Issue 3, pp. 203-217.
16. M. R. Cohen and J. R. Forbes, Navigation and Control of Unconventional VTOL UAVs in Forward-Flight With Explicit Wind Velocity Estimation, IEEE Robotics and Automation Letters, 2020, Vol. 5, Issue 2, pp. 1151-1158.
17. M. G. Patan, & I. Ustoglu, Robust Augmented State Extended Kalman Filter for Actuator/Sensor Fault Detection and Isolation in Quadrotor Unmanned Aerial Vehicles, Journal of Dynamic Systems, Measurement, and Control, 2026, Vol. 148, Issue, 1, 011001.
18. M. Karahan, Nonlinear Modelling and Robust Backstepping Control of a Quadcopter in Aggressive Maneuvering. Studies in Informatics and Control, 2024, Vol. 33, Issue 3, pp. 29-38.
19. M. Karahan, C. Kasnakoglu, & A. N. Akay, Robust backstepping control of a quadrotor uav under pink noise and sinusoidal disturbance, Studies in Informatics and Control, 2023, Vol. 32, Issue 2, pp. 15-24.
20. M. Karahan, C. Kasnakoglu, Modeling a Quadrotor Unmanned Aerial Vehicle and robustness analysis of different controller designs under parameter uncertainty and noise disturbance. Journal of Control Engineering and Applied Informatics, 2021, Vol. 23, Issue 4, pp. 13-24.