



Global tracking control of quadrotor based on adaptive dynamic surface control

Bara J. Emran¹ · Homayoun Najjaran¹

Received: 16 January 2020 / Revised: 3 April 2020 / Accepted: 10 April 2020 / Published online: 30 April 2020
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

Abstract

This paper presents a global nonlinear tracking control system for a quadrotor unmanned aerial vehicle (UAV) in the presence of underactuation, external disturbances and model uncertainties. Quadrotor systems lack enough independent control inputs to control their entire configuration space directly due to underactuation. The proposed solution is to adopt a cascade feedback technique that splits the system dynamics into attitude and position dynamics. The proposed controller is developed directly on the special Euclidean group with a region of attraction covering the entire configuration space where its stability is proven using Lyapunov functions. The controller guarantees the asymptotical convergence of tracking error in the presence of model uncertainties and external disturbances. In particular, the control method combines three techniques: a second-order sliding mode control (SMC), a dynamic surface control, and a non-parametric adaptation mechanism. The SMC is used to stabilize the position dynamics (internal dynamics) by generating a proper attitude command for the attitude controller. The DMC control guarantees the attitude dynamics stability globally and tracking performance while avoiding the mathematical complexities associated with the highly nonlinear dynamics. The adaptation mechanism includes a radial basis function neural network to observe uncertainties without the need for prior training. The uncertainties considered include unmodeled dynamics, external disturbances and parameter uncertainties including the mass and inertial matrices as well as motor coefficients. The desirable features of the proposed control system are illustrated by both numerical simulation and experiments on a UAV testbed.

Keywords Dynamic surface control · System uncertainties · Unmanned aerial vehicle

1 Introduction

For over a decade, the attention of robotics researchers worldwide has been drawn towards unmanned aerial vehicles (UAVs) due to their numerous applications. Among different types of multi-rotor UAVs, quadrotors are considered the most preferred platform for critical applications, especially where safe performance in complex and cluttered environments is required [1–4]. This is thanks to their great dynamical maneuverability, vertical takeoff/landing ability, lightweight and, most importantly, their simplicity in design. However, the quadrotor advantages come at some cost. Specifically, their control challenges including system underactuation, model uncertainties, and subsystem coupling complicate their stabilization and trajectory tracking,

significantly. Furthermore, other aspects such as actuator saturation, disturbances, noises influence control performance. For a control system to have acceptable performance, these challenges need to be taken into consideration. The work represented here mainly aims to tackle these control problems by designing a globally robust tracking controller. The main shortcoming of a quadrotor system is its lack of enough independent control inputs i.e., the quadrotor is an underactuated system. In other words, the lack of adequate control actions in the quadrotor's configuration space prevents them from following unrestricted flight in full vector space. Consequently, different techniques have been developed to deal with this challenge. One of the most practical and simple strategies is known as the cascade feedback strategy [5–10] that uses the dynamic inversion scheme [11]. It is based on dividing the system into two subsystems including an outer-loop and an inner-loop that form the position dynamics and attitude dynamics, respectively. Afterward, the subsystems are connected after designing a feedback control for each loop separately. Even though applying this technique

✉ Bara J. Emran
Bara.emran@gmail.com

¹ University of British Columbia, Kelowna, BC V1V 1V7, Canada

has simplified the control problem, one still needs to guarantee the stability of the internal dynamics as they may lead the system to instability. Hence, one needs non-traditional control algorithms to deal with the underactuation of the system and the underlying internal dynamics. More recent attempts for controlling quadrotor involve the use of backstepping control (BC) [12–14]. Basically, the BC approach is developed for designing stabilizing controls and for a special class of nonlinear dynamical systems (systems with cascaded and coupled dynamics). The BC design is a recursive process that involves stabilizing the system states step-by-step to guarantee the stability of the entire system. However, the use of BC can be problematic because of the “explosion of terms” in its control law. Despite this challenge, BC is one of the commonly used approaches for quadrotor control in the literature. Another approach is the incorporation of discontinuous solutions through a hybrid control architecture or a switching mechanism [15–17]. In the last decade, the interest in the use of discontinuous solutions has increased, especially for cases in which a suitable solution cannot be provided using traditional control algorithms [18]. However, the use of this approach can also lead to other challenges such as chattering of the input signal due to switching that may excite some unmodeled dynamics and consequently destabilize the system. Furthermore, proving the stability of the overall system may be nontrivial, as there may exist a specific switching sequence that can cause system instability.

Through the literature, numerous control systems have been proposed for quadrotors with different representations of the UAV attitude. In some cases, the control design is based on local coordinates for simplicity [9]. However, this restricts the ability of the control system to achieve complex maneuvers, which is critical for multi-rotor vehicles. In most cases, the control systems are based on Euler-angles as their attitude representation [6–8]. This allows the control system to achieve better stability and performance than that based on local coordinates. Nonetheless, the control system still suffers from singularities in their attitude representation. Furthermore, the control law will involve complicated expressions with trigonometric functions, which cannot be linearized for complex maneuvers. Other control approaches are based on quaternion for their attitude representation. Quaternion-based control systems are preferable over Euler-angle based ones, as they do not suffer from singularities in their representation [10]. Still, quaternion representations can cause ambiguity, as one attitude can be represented by two different representations. Unless this ambiguity is resolved by the control system, it will lead to undesired behavior where the quadrotor unnecessarily intends to rotate a large angle. Recently, the focus of researchers is shifted towards designing control systems based on the special Euclidean group, $SE(3)$ to represent the attitude [19]. Although the latter is not as compact as the quaternion representation, it has been

favored in recent work since it is a globally valid expression even for complex maneuvers while avoiding singularities and ambiguities without complex mathematical expressions.

In general, traditional control methods are based on model approach, as they start with a mathematical model that describes the system operation. However, the developed mathematical model is usually inaccurate in capturing the process dynamics leading to model uncertainties. For example, a quadrotor carrying unknown payloads in the package delivery application is subject to a large parameter uncertainty attributed to variations of mass and inertia [3,20]. More generally, uncertainties can be attributed to unknown parameters, external disturbances and unmodeled nonlinear dynamics e.g., wind. Thus, solutions with adaptive features are required to overcome potential model uncertainties and control the real plant [9,16]. These control solutions can be interpreted as online techniques capable of controlling systems with bounded uncertainties in their dynamics model. However, using only an adaptation technique in the control system can still destabilize the system in the presence of small disturbances. This is because typical adaptive mechanisms work only with structured uncertainties while performing poorly with unstructured and non-parametric ones. Consequently, adaptive controllers need to be enhanced with robust features to effectively tackle unmodeled dynamics or external disturbances [21,22]. On the other hand, several researchers investigated the integration of artificial intelligence (AI) techniques into an adaptive controller to improve the overall performance [6,23]. One of the main advantages of AI techniques is their use of complex learning models, unlike most adaptive approaches that can take simple models with only a limited number of control parameters into consideration. The drawback is that AI techniques typically require high computational power and prior training.

Literature about the control of quadrotors is vast. A great review of the recent work in the quadrotor system can be found in [24]. Table 1 summarizes a review of recent work on control systems describing the general framework, proposed method, challenges and drawbacks, robustness, attitude representation as well as the verification and demonstration of each method. In this work, we utilize an alternative control technique called dynamic surface control (DSC). Similar to the backstepping control (BC) method, this technique uses a step-by-step recursive process utilizing multiple sliding surfaces to stabilize the dynamic system. However, it avoids the drawback of BC that is the “explosion of terms” when it is applied to highly complicated systems. Specifically, it does that by incorporating a series of first-order low-pass filters into the recursive process [25]. The proposed control scheme combines the second-order sliding mode controller with the DSC technique to enable the globally robust tracking control of a quadrotor. The contributions of the control system proposed in this paper compared to other studies [6,16,26] are as

Table 1 Review of the method, approach and challenges of recent control systems

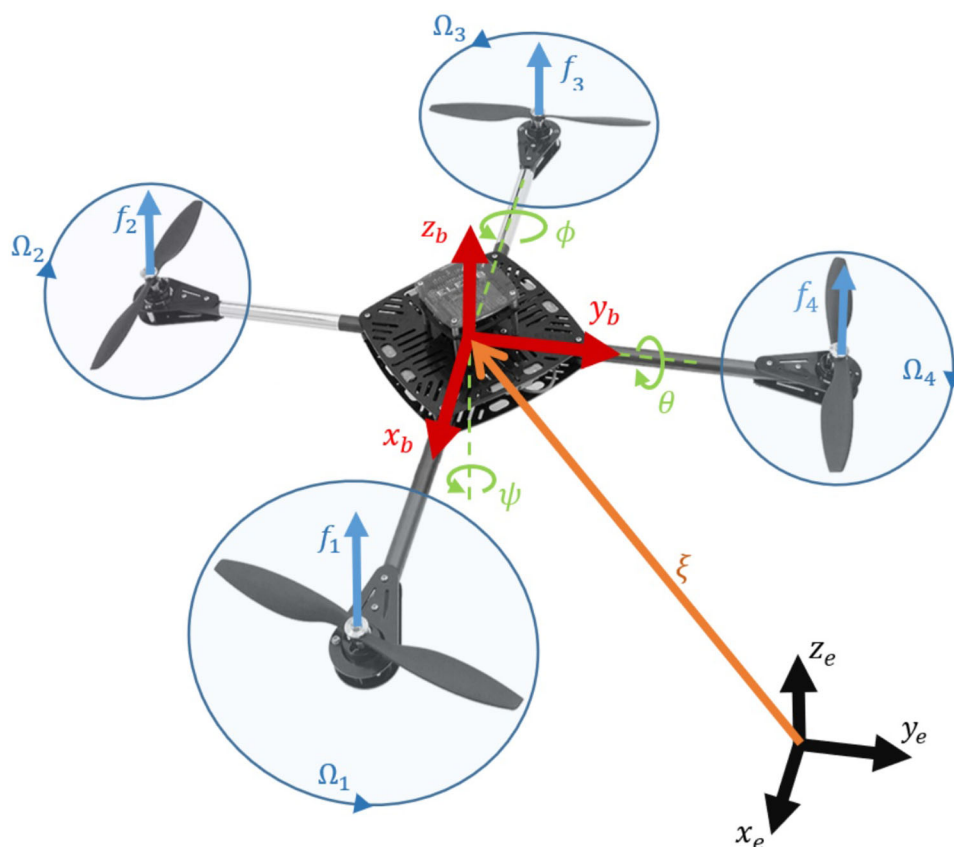
Framework	Refs	Description	Shortcomings	Robust ^a	Attitude ^b	Demo ^c
Cascade approach	[5]	Different feedback control systems and shows their implementation	Ignores internal dynamics that may cause instability	N/A	SE	S
	[6]	A robust nonlinear composite adaptive control based on single layer RBF-NN to estimate uncertainties manifested in the model parameters	Assumes the availability of the quadrotor states	AI	EA	S
	[7]	A tracking control utilize a terminal sliding mode control in order to improve the system tracking and to reduce the presence of chattering in the switching signal	Ignores measurement noise	N/A	EA	S
	[8]	A second order sliding mode control is used to address the position and attitude tracking problem	Results in a slow convergence rate	N/A	EA	S
	[9]	A combination of an adaptive controller to improve the robustness to parametric uncertainties	Uses small angles approximation	A	SA	MC
Backstepping	[10]	A robust control based on PD and a first-order compensator	Lack focus on position control	R	Q	HIL
	[12]	Combine PID in Backstepping approach with NN to estimate the unmodeled dynamics and improve the control system robustness	Moderate complexity and it depends on the initial conditions	AI	EA	S
	[13]	A nonlinear controller backstepping-like feedback linearization method	Requires on the exact plant parameters identified through rigorous experiments	N/A	EA	R
	[14]	A projection-based adaptive controller based on BC approach for mixed quadrotor-type	Focuses only on uncertainty in mass	A	SE	R
Discontinues Switching Solutions	[15]	Novel switching algorithm based on adaptive reference control and multiple Lyapunov functions to solve the system uncertainty and underactuation problems	Chattering effects appear in the control signals due to the switching behaviour	RA	EA	S
	[16]	Switching between two controllers for hovering and agile maneuvers Consider uncertainty model parameters and external disturbance	An asymptotical tracking performance with a relatively small attitude error but with a large angular velocity error	A	SE	MC
	[17]	Utilize a hybrid automaton to capture the dynamics of different flight regimes that the UAV crossover, for example flying close to a landing platform and free-flight maneuvers	Require extra information of the environment also it contains unconventional maneuver	N/A	EA	MC

^aUncertainties methods are selected as follows: not applicable (N/A), adaptive (A), robust (R), robust adaptive (RA), artificial intelligent (AI)

^bAttitude representation are selected as follows: Small angle approximation (SA), Euler-Angles (EA), quaternion (Q), special Euclidean group (SE)

^cDemonstration methods are selected as follows: hardware in the loop (HIL), motion capture system (MC), simulation (S) and real system (RS)

Fig. 1 Quadrotor's structure and reference frames. The body reference frame (B-frame) is a right-handed frame where its origin (o_B) is selected at the center of mass of the quadrotor. Its axis points toward the front (x_B), left (y_B) and up (z_B). The earth inertial reference frame (E-frame) is right-handed frame selected in North-East-Up configuration



follows. First, it is developed directly on the nonlinear configuration manifold to enable robust tracking with a region of attraction that covers the entire configuration space. Also, it is developed for the full six degrees of freedom (DOF) dynamic model of a small-scale quadrotor without any simplification or assumptions except omitting the motor dynamics. Second, the control system combines a second-order sliding mode controller for high accuracy tracking the performance of the translation dynamics and the dynamic surface control (DSC) technique that handles the coupling effects between the transitional and rotational dynamics. Third, the proposed attitude controller is based on DSC and developed directly on the special Euclidean group with a region of attraction covering the configuration space globally. This will allow the controller to maintain both dynamics stability and tracking performance while avoiding the singularity associated with orientation representations. Fourth, a combination of a robust adaptive technique based mainly on radial basis function neural network (RBF-NN) is proposed. This will guarantee the asymptotical convergence of tracking error in the presence of external disturbances and uncertainties in unmodeled dynamics and system parameters including mass and inertial matrices as well as motor coefficients. Fifth, the proposed algorithm is validated with simulation for agile maneuver and further examined using a testbed subjected to high dis-

turbances. The testbed has 3 DOF for the attitude dynamics and another 3 DOF implemented in simulation for the translation dynamics since the testbed rotates around 3 axes, but its position is fixed. The numerical simulations are carried out using MATLAB SIMULINK[®]. The illustrative results prove that the proposed control approach outperforms similar approaches in terms of both accuracy and convergence rate. This paper is organized as follows. The dynamic model of the quadrotor system is described in detail in Sect. 2. The problem statement and proposed control system are described in Sects. 3 and 4, respectively. The results are discussed in Sect. 5. In Sect. 5.1, the numerical simulation results are illustrated to highlight the overall performance and the effectiveness of the designed controllers. Section 5.2 describes the testbed and experimental results. Concluding remarks are provided in Section 6.

2 Quadrotor dynamic system

In general, quadrotors are preferable UAV platform for applications with cluttered and complex environments due to their unique features. A quadrotor has 6 DOF with only four actuators. Typically, the four actuators have fixed and parallel axes of rotation and they form a multi-rotor cross-platform.

Its symmetry allows the control system and payload to be centralized. Moreover, it uses fixed-pitch propellers that allow the rotors to create a downward push and lifts the vehicle. In contrast to helicopter UAV which requires a tail rotor for stabilization, quadrotor eliminates that need by configuring the rotation directions of its rotors in a certain way. In particular, the rotation direction of the left and right rotors are in the opposite direction of the rotation of the rear and front rotors. The altitude and attitude of the vehicle are controlled by changing the rotational speed of the rotors to generate forces and torques along and around the axes of the vehicle. This section starts by outlining the reference frames and followed by describing the nonlinear dynamical model of the system.

2.1 Reference frames

The first step in modeling mechanical systems involves defining proper reference frames to simplify the procedure. In this work, two right-hand reference frames were selected to describe the motion of the quadrotor’s rigid body as shown in Fig. 1. The frames are body reference frame (B-frame) and earth inertial reference frame (E-frame).

The system configuration is represented using its position and attitude with respect to the E-Frame. The distance vector between the origins of the E-frame and B-frame is known as the linear position vector $p = [x \ y \ z]^T$. The linear position $p = [x \ y \ z]^T$ represent the distance from E-frame to B-frame. The vehicle’s attitude angles can be defined as the rotation angles between the B-frame and E-Frame represented in the E-frame. These angles are known as Euler angles and donated by $\Theta = [\phi \ \theta \ \psi]^T$. However, the angular velocity denoted by $\omega = [p \ q \ r]^T$ is described with respect to B-Frame. In addition, all the control inputs are defined in the Body frame (B-Frame), including torques $\tau \in \mathbb{R}^3$ and thrust $F \in \mathbb{R}$ signals. The rotation matrix $R \in \mathbb{R}^{3 \times 3}$ is a special orthogonal group SO(3) used to map a vector from the B-frame to the E-frame and has the following properties:

$$R^T R = I_{3 \times 3}, \det(R) = 1 \tag{1}$$

Using R, the direction of the i th axis of the B-frame is represented by its i th column and is given by:

$$b_i = R e_i, \quad i = \{1, 2, 3\}$$

$$e_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad e_2 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \quad e_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{2}$$

There are different types of forces acting on the vehicle body, however, we are going to focus only on weight and thrust.

The vehicle weight acts along the direction of z-axis in the E-frame and denoted by mg , where $m \in \mathbb{R}$ represents the quadrotor’s mass in (kg) and g is the gravitational acceleration in (m/s^2 kg). The thrust forces are denoted by f_i and generated by the rotors which act in the direction of the B-frame z-axis. Additionally, a reaction torque τ_i is generated by the rotation of the propeller and acts on the vehicle body in the same direction of rotation. The thrust and torque generated by each rotor are directly proportional to its rotation speed of Ω_i and defined as follows:

$$f_i = K_b \Omega_i^2 \tag{3}$$

$$\tau_i = K_d f_i \tag{4}$$

where $i = \{1, 2, 3, 4\}$ represents the number of rotors, $\Omega_i > 0$ is the speed of the i th rotor in (rad/s), K_b represents the rotor’s thrust factor in ($N s^2$), K_d is thrust to torque ratio. The total thrust acting on the body F is equal to the summation of all thrust forces generated by the rotors. Additionally, the overall moments acting upon the body, which consist of the reaction torques and the moments due to the rotor forces, are expressed as follows

$$\begin{bmatrix} F \\ \tau_p \\ \tau_q \\ \tau_r \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & l & 0 & -l \\ -l & 0 & l & 0 \\ K_d & -K_d & K_d & -K_d \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \tag{5}$$

where l represents the arm distance, the distance between the center of each rotor to the center of mass in (m). It was found that the dynamics of the rotors have a low influence on the resultant system behavior. Hence, it was omitted from the system dynamics. For simplicity, the following assumptions were made:

1. The speeds of the propellers are controlled by high bandwidth motors.
2. The motor speed is unaffected by the quadrotor motion.
3. The propeller’s rotational inertia is much smaller than the vehicle inertia.

Through the literature, various research articles are focusing on the modeling of the quadrotor system [27,28]. Due to the different assumptions and simplifications used by these studies, different models have been proposed. Here, we are adopting one of the most common dynamical models in the literature. The dynamic equation is described using the Euler-Lagrange approach as follows:

$$\dot{p} = v \tag{6}$$

$$m\dot{v} = -mge_3 + FR e_3 + d_v \tag{7}$$

$$\dot{R} = R\omega_{\times} \tag{8}$$

$$J\dot{\omega} + \omega \times J\omega = \tau + d_\omega \tag{9}$$

where $J \in \mathbb{R}^{3 \times 3}$ represents the inertia matrix in (N m s²), $d_v \in \mathbb{R}^3$ and $d_\omega \in \mathbb{R}^3$ represent the modeling error and uncertainties in the translation and rotation dynamics, respectively, The second term of the dynamic equation of angular velocity expresses the cross-coupling of the angular momentum in the system.

The times map $\bullet_\times : \mathbb{R}^3 \rightarrow SO(3)$, known also as skew matrix, is defined by the condition that $a_\times b = a \times b$ for all $a, b \in \mathbb{R}^3$, the matrix a_\times is given by:

$$a_\times = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{10}$$

where the inverse of times map is denoted by the *vee* map $\bullet^\vee : SO(3) \rightarrow \mathbb{R}^3$ that is $(a_\times)^\vee = a$.

3 Problem definition

In general, quadrotor systems suffer from uncertainties in their parameters and models which may cause inaccuracy or instability for the control system. Thus, the control systems require having some robustness to overcome the uncertainties effects and adaptation mechanisms to improve the system response with time. Consider a nonlinear system described by (6-9) and rewrite it as follows:

$$\dot{p} = v \tag{11}$$

$$\dot{v} = F_v + \theta_v G_v u_v + D_v \tag{12}$$

$$\dot{R} = F_R(R, \omega) \tag{13}$$

$$\dot{\omega} = F_\omega + \theta_\omega u_\omega + D_\omega \tag{14}$$

where

$$\left. \begin{aligned}
 &F_v = -g e_3, \quad G_v = R e_3 \\
 &F_R(R, \omega) = R \omega_\times \\
 &\theta_v = \frac{K_b}{m}, \quad D_v = \frac{d_v}{m} \\
 &J = J_0 + \Delta J \\
 &J^{-1}(d_\omega - \omega \times J\omega) = F_\omega + D_\omega \\
 &F_\omega = -J_0^{-1}(\omega \times J_0\omega) \\
 &D_\omega = \Delta F_\omega + J^{-1}d_\omega \\
 &\theta_\omega = J^{-1} \begin{bmatrix} K_b & 0 & 0 \\ 0 & K_b & 0 \\ 0 & 0 & K_d \end{bmatrix} \\
 &u = \begin{bmatrix} u_v \\ u_\omega \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & l & 0 & -l \\ -l & 0 & l & 0 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \\ \Omega_4 \end{bmatrix}
 \end{aligned} \right\} \tag{15}$$

where $u(t) \in \mathbb{R}^4$ is the input vector, $y(t) = [P^T, \psi]^T \in \mathbb{R}^4$ is the output vector, $x(t) = [P, v, R, \omega]^T \in \mathbb{R}^{18}$ is the state vector, $F_v \in \mathbb{R}^3$ and $F_\omega \in \mathbb{R}^3$ are the state function, $\theta_v \in \mathbb{R}$ and $\theta_\omega \in \mathbb{R}^{3 \times 3}$ are the system's uncertainties and $D_v \in \mathbb{R}^3$ and $D_\omega \in \mathbb{R}^3$ are the lumped uncertainties in the system model including unmodeled dynamics and external disturbances. The goal of this research is to design a robust adaptive nonlinear motion controller for a 6DOF UAV to achieve a proper path tracking in presence of external disturbances and system underactuation. Here are the main considered assumptions:

1. All of the state variables, $x(t)$, are measurable.
2. θ_v and θ_ω are unknown constant parameters with known sign and lower threshold $\bar{\theta}_v$ and $\bar{\theta}_\omega$, respectively.
3. Upper bounds for the vehicle uncertainties $D_v(t)$ and $D_\omega(t)$ are assumed $D_v(t) \leq D_{vmax}$ and $D_\omega(t) \leq D_{\omega max}$. This assumption can be achieved by using a nonlinear model with a good level of accuracy.

With the above assumptions, the system stability can be guaranteed using the proposed control systems.

4 Control system development

In general, the purpose of the developed control system is to guarantee system stability and achieve proper tracking for the desired command signals selected as a position in 3 DOF $[x_d, y_d, z_d]$ and heading angle ψ_d . However, quadrotors are an underactuated system. In this research, a cascade feedback strategy is adopted by the control system using a dynamic inversion technique to control the 6 DOF quadrotor [11]. This strategy involves splitting the UAV dynamics into two subsystems; attitude dynamics and position dynamics. Then control each subsystem separately by designing two feedback controllers. Each of these controllers is based on a different control technique suitable for their individual control challenges. Finally, the controls are connected together to control the overall system. In particular, the control signals of the outer controller (position controller) is used to provide a reference signal to the inner controller (attitude controller). Figure 2 illustrates the block diagram of the overall control system.

4.1 Position flight control

The outer-loop dynamics are introduced using the feedback linearization technique and given by the Eqs. (11–12). The outer-loop includes the internal dynamics of the translation motion in, x and y axes along with the altitude motion in, z-axis. The control system must stabilize the linear dynamics

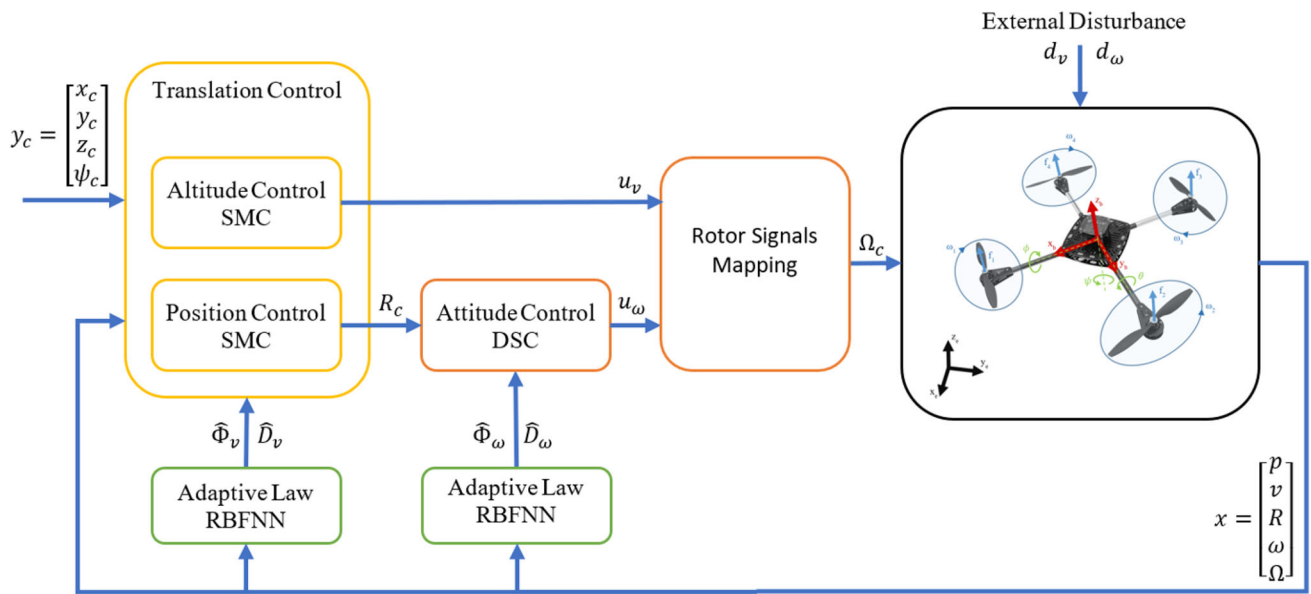


Fig. 2 Block diagram of the control system. The outer-loop assigns a command rotation (R_c) and control signal u_v from the command signal of position and heading angle. The inner-loop control the underactuated

dynamics u_ω . The rotor control takes the input signal and controls the rotor speed by generating a proper signal command Ω_c . The adaptive laws estimate the disturbances \hat{D}_i and unknown parameters $\hat{\Phi}_i$

(internal dynamics) to guarantee the stability of the overall system. The position flight control is divided into two controllers, planner control and altitude control.

4.1.1 Planar control

The goal of the position control is to produce a suitable attitude command signal R_c for the attitude controller to track. The command signal is developed in the $SO(3)$, therefore it avoids singularities of Euler angles and unwinding quaternions. Let us start by assuming a smooth desired position tracking command signals $p_d = [x_d \ y_d \ z_d]^T \in \mathbb{R}^3$ is given. Define a tracking error signal for position states as

$$e_p = \begin{bmatrix} e_x \\ e_y \\ e_z \end{bmatrix} = \begin{bmatrix} x - x_d \\ y - y_d \\ z - z_d \end{bmatrix} \tag{16}$$

$$e_v = \dot{e}_p = \begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_z \end{bmatrix} = \begin{bmatrix} v_x - \dot{x}_d \\ v_y - \dot{y}_d \\ v_z - \dot{z}_d \end{bmatrix} \tag{17}$$

Now, define the sliding surface for the sliding mode control of uncertain nonlinear systems as [19]

$$s_p = \begin{bmatrix} s_x \\ s_y \\ s_z \end{bmatrix} = e_v + \lambda_p e_p \tag{18}$$

where λ_p is a design positive diagonal matrix defined later. Calculate the dynamics of the sliding surface using the

nonlinear system defined in (11)-(12) yields

$$\begin{aligned} \dot{s}_p &= (\dot{v} - \dot{v}_d) + \lambda_p e_p \\ \dot{s}_p &= (F_v + \theta_v G_v u_v + D_v + -\dot{v}_d) + \lambda_p e_p \end{aligned} \tag{19}$$

Using the relation between the rotation matrix and sliding surface dynamics, one can utilize the sliding error to construct the direction of the third axis of the command attitude signal as follows

$$q_p = \dot{v}_d - \hat{D}_v - F_v - \lambda_p e_v - K_p s_p \tag{20}$$

$$R_c e_3 = b_{3c} = q_p / \|q_p\| \tag{21}$$

where K_p is a positive diagonal matrix and selected later with λ_p to represent a stable Hurwitz polynomial. $\hat{D}_v \in \mathbb{R}^3$ denote the estimate of the disturbance signal D_v defined using the RBF-NN as

$$\hat{D}_v = \hat{f}(e_v, \dot{e}_v) = \hat{W}_v^T h_v(s_v) \tag{22}$$

where \hat{W}_v is the estimated disturbance weights.

The component of unit vector b_{3c} is constructed to resemble the weight of the correction term q_{p_i} of each translation axis. The direction of the first axis of the command attitude signal can be chosen to specify the desired heading direction of the quadrotor in the horizontal plane. This is done using

the command heading angle (ψ_c) as follows

$$b_{1ideal} = \begin{bmatrix} \cos(\psi_c) & -\sin(\psi_c) & 0 \\ \sin(\psi_c) & \cos(\psi_c) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad (23)$$

However, we need to ensure that the attitude command signal R_c is defined properly, that is $R_c \in SO(3)$. Practically, the direction of its first column need to be orthogonal to the third column. This could be done by find the projection of the of the ideal first axis on the third axis as follows

$$b_{1c} = Proj_{b_{3c}}(b_{1ideal}) \quad (24)$$

$$b_{1c} = b_{1ideal} - \left(b_{1ideal}^T b_{3c} \right) b_{3c} / \|b_{3c}\|$$

It is worth to mention that, using the projection function, the first axis of the attitude command signal as $t \rightarrow \infty$ converges to the projection of b_{1c} not b_{1ideal} . Then the full construction of the attitude command signal is found as

$$R_c = [b_{1c}, b_{3c} \times b_{1c}, b_{3c}] \quad (25)$$

4.1.2 Altitude control

The purpose of this part is to develop a control system for the altitude motion in the z-axis. To design a proper tracking signal of the altitude desired command signal z_d , we need to calculate the value of the input signal u_v . We can reuse the chosen sliding surface defined in the previous step to build a proper input signal. The time derivative of the third element of the sliding surface defined in (19) is rewritten as follows

$$\dot{s}_{p_z} = \dot{v}_z - \ddot{z}_d + \lambda_{p_z} e_{p_z} \quad (26)$$

$$\dot{v}_z = F_{v_z} + \theta_v R_{3,3} u_v + D_z \quad (27)$$

where the z-subscript indicates the third element of a corresponding vector, $R_{3,3}$ is the third element of the third column of the current attitude rotation matrix. Define the estimation of the parameter θ_v and its error as

$$\tilde{\theta}_v = \theta_v - \hat{\theta}_v \quad (28)$$

where $\hat{\theta}_v \in \mathbb{R}$ is the estimation of the unknown parameter θ_v and $\tilde{\theta}_v$ is the estimation error. We can now define the input signal u_v using (19) as follows

$$u_v = \left(R_{3,3} \hat{\theta}_v \right)^{-1} q_{p_z} \quad (29)$$

By substituting the input signal into the dynamics of the sliding surface defined in (26) yields

$$\dot{s}_{p_z} = -K_{p_z} s_{p_z} - \tilde{W}_{v_z}^T h_{v_z} + \epsilon_z - \tilde{\theta}_v R_{3,3} q_{p_z} \quad (30)$$

$$\tilde{W}_{v_z} = W_{v_z} - \hat{W}_{v_z} \quad (31)$$

where \tilde{W}_{v_z} is the estimation error of the disturbance signal. With $K_{p_z} > 0$ and $\tilde{W}_{v_z}, \tilde{\theta}_v \rightarrow 0$ as $t \rightarrow \infty$, then $s_{p_z} \rightarrow \epsilon_z$. Proof: using the following Lyapunov function:

$$V_z = \frac{1}{2} s_{p_z}^2 + \frac{1}{2\gamma_v} \tilde{W}_{v_z}^2 + \frac{1}{2\eta_v} \hat{\theta}_v^2 \quad (32)$$

where γ_v and η_v are positive constants represent the learning rate of the estimated weights and parameters, respectively. The derivative of the Lyapunov function can be expressed as

$$\dot{V}_z = -K_{p_z} s_{p_z}^2 + \epsilon_z s_{p_z} + \tilde{W}_{v_z}^T \left(s_{p_z} h_z - \frac{1}{\gamma_{v_z}} \dot{\tilde{W}}_{v_z} \right) - \hat{\theta}_v \left(R_{3,3} s_{p_z} q_z + \frac{1}{\eta_{v_z}} \dot{\hat{\theta}}_v \right) \quad (33)$$

Let us choose the following adaption function to to avoid the singularity and guarantee the stability of the control signal

$$\dot{\hat{W}}_{v_z} = \gamma_v s_{v_z} h_{v_z} \quad (34)$$

$$\alpha_z = R_{3,3} s_{p_z} q_z \quad (35)$$

$$\dot{\hat{\theta}}_v = \begin{cases} -\eta_{v_z} \alpha_z & \alpha_z > 0 \\ -\eta_{v_z} \alpha_z & \alpha_z \leq 0 \ \& \ \hat{\theta}_v > \bar{\theta}_v \\ -\eta_{v_z} & \alpha_z \leq 0 \ \& \ \hat{\theta}_v \leq \bar{\theta}_v \end{cases} \quad (36)$$

Here, $\bar{\theta}_v$ is the lower limit of the parameter θ_v where the initial value of estimated parameter must follow the condition of $\hat{\theta}_v(0) > \bar{\theta}_v$. Thus, the derivative can be represented as:

$$\dot{V}_z = -K_{p_z} s_{p_z}^2 + \epsilon_z s_{p_z} \quad (37)$$

The adaptive algorithm RBF-NN guarantees that the approximation error ϵ is adequately small enough and limited that makes ($\dot{V}_z < 0$). As a result, the filter error s_{p_z} has an exponential convergence rate and, in return, guarantees the convergence of tracking error e_p . Using the Barbalat's extension so does \dot{e}_p . Hence the altitude tracking is achieved. Similarly, we can generalize the adaption law of the disturbance weights \hat{W}_v for all position dynamics as defined in (34)

4.2 Attitude flight control

The inner-loop subsystem represents the UAV's attitude dynamics. It considered a fully actuated system as it has 3DOF and 3 input signals, i.e. $u_\omega \in \mathbb{R}^3$. Here, the controller system's main objective is to guarantee the exponential convergence of the attitude, represented by a rotation matrix R , toward the rotational command signal R_c . Furthermore, the control system has to guarantee the system's stability and performance in the presence of external disturbances and model

uncertainties. This will be accomplished using Dynamic Sliding Control (DSC) technique. The DSC method is a recursive procedure which basically composed of multiple sliding surfaces and first-order low-pass filters. In particular, the employment of the filters manifests to overcome the mathematical difficulties of an “explosion of terms”, which is a major drawback of similar techniques, such as integrator backstepping technique.

4.2.1 Attitude tracking control

To ensure the smoothness of the attitude command tracking signal R_c generated by the position control, we will introduce the following first-order low-pass filter and its augmented error as

$$\tau_R \dot{R}_d + R_d = R_c \tag{38}$$

$$\zeta_R = R_d - R_c \tag{39}$$

where τ_R represent the time constant of the filter and will be determined later, $R_d(3)$ is the filtered signal of the command attitude. For a given desired attitude tracking R_d and the current attitude (R), we define first error surface vector for attitude dynamics as follows

$$s_R = \left(R_d^T R - I_{3 \times 3} \right)^\vee \tag{40}$$

With its time derivative is calculated using (12) as

$$\dot{s}_R = \left(\dot{R}_d^T R + R_d^T R \omega_\times \right)^\vee \tag{41}$$

At this step, the objective is to ensure the stability of the first sliding surface, that is to drive $s_R \rightarrow 0$ as $t \rightarrow \infty$. This is done by designing a suitable forcing term for the angular velocity defined as:

$$\omega_{c \times} = \left(R_d^T R \right)^T \left(-\dot{R}_d^T R - q_R \right) q_R = \left(K_R S_R \right)_\times \tag{42}$$

where K_R is a positive diagonal matrix defined later. When the angular velocity converges to the forcing term, that is $\omega \rightarrow \omega_c$, the dynamics of the error surface becomes

$$\dot{s}_R = -K_R s_R \tag{43}$$

Thus, guarantee the exponential convergence of the first error surface s_R . Angular Velocity Tracking Control Similarly, we can choose the second first-order low-pass filter for the command angular velocity and its augmented angular velocity error as follows:

$$\tau_\omega \dot{\omega}_d + \omega_d = \omega_c \tag{44}$$

$$\zeta_\omega = \omega_c - \omega_d \tag{45}$$

where τ_ω represent the time constant of the filter and will be determined later, and $\omega_d \in \mathbb{R}^3$ is the desired angular velocity. For a given desired angular velocity ω_d and the current angular velocity ω , we define the second error surface vector as follows

$$s_\omega = \omega - \omega_d \tag{46}$$

With its time derivative is calculated using (13) as:

$$\dot{s}_\omega = \dot{\omega} - \dot{\omega}_d \tag{47}$$

$$\dot{s}_\omega = F_\omega + \theta_\omega u_\omega + D_\omega - \dot{\omega}_d \tag{48}$$

At this step, the objective is to ensure the stability of the second error surface, that is drive $s_\omega \rightarrow 0$ as $t \rightarrow \infty$. This is done by designing a suitable input signal u_ω . First, let us define the estimation error $\tilde{\theta}_\omega$ as the difference between the unknown parameter θ_ω and its estimation $\hat{\theta}_\omega$ and write it as

$$\tilde{\theta}_\omega = \theta_\omega - \hat{\theta}_\omega \tag{49}$$

Using this relation, we can now define the control signal as

$$u_\omega = \hat{\theta}_\omega^{-1} (q_\omega) \tag{50}$$

$$q_\omega = \dot{\omega}_d - F_\omega - \hat{D}_\omega - K_\omega s_\omega \tag{51}$$

where K_ω is a positive diagonal matrix defined later, $\hat{D}_\omega \in \mathbb{R}^3$ is the disturbance estimation defined using the RBF-NN as

$$\hat{D}_\omega = \hat{f}(s_\omega) = \hat{W}_\omega^T h_\omega(s_\omega) \tag{52}$$

where \hat{W}_ω is the estimated disturbance weights. Substituting the input signal u_ω into the dynamics of the second error surface defined in (48) yields

$$\dot{s}_\omega = -K_\omega s_\omega + \tilde{W}_\omega^T h_\omega + \epsilon_\omega + \tilde{\theta}_\omega u_\omega \tag{53}$$

$$\tilde{D}_\omega = D_\omega - \hat{D}_\omega = \tilde{W}_\omega^T h_\omega + \epsilon_\omega \tag{54}$$

where \tilde{W}_ω represents the difference between the estimated weights and their desired values. One can easily prof that the approximation error ϵ_ω is limited and sufficiently small. As the estimation error signal of \tilde{W}_ω and $\tilde{\theta}_\omega$ goes to zero, the angular velocity error surface will go to zero, i.e. $s_\omega \rightarrow 0$.

4.2.2 Neural network-based adaptive controller

Similar to the work in [29], this paper approximates the uncertainty in the nonlinear functions using neural network technique, namely radial basis function neural network

(RBF-NN). In brief, this technique is defined as:

$$h_j = e^{\|x-c_j\|^2/2bj^2} \tag{55}$$

$$f(x) = W^{*T} h(x) + \epsilon \tag{56}$$

where h function defines the output of the Gaussian function, signal x defined as the input signals of the neural network, parameter j represent the number of hidden layer nodes in the network, the function f defines the overall output value of the network, the neural network's ideal weights are denoted by W^* , finally, the parameter ϵ defined as the approximation error. To estimate the nonlinear function of the dynamic system $f(x)$, one must estimate the ideal weights W^* of the NN as:

$$\hat{f}(x) = \hat{W}^T h(x) \tag{57}$$

$$\tilde{W} = W - \hat{W} \tag{58}$$

where \hat{f} and \hat{W} represent the estimated disturbance and weights, respectively, \tilde{W} defined as the error between the estimated weights and their desired values, that is

$$f - \hat{f} = \tilde{W}^T h + \epsilon \tag{59}$$

4.3 Error dynamics

Once the design procedure is applied, the closed-loop error dynamics need to be derived for stability analysis. Using the definition of the error surfaces, forcing term and input signals, the nonlinear system equations for the rotation dynamics can be described as the error equations of DSC as follows:

$$\begin{aligned} \dot{s}_R &= (-\dot{R}_d^T R - R_d^T R (\omega_{c\times} - s_{\omega\times} + \zeta_{\omega\times}))^\vee \\ \dot{s}_R &= -K_R s_R + R_d^T R (s_\omega - \zeta_\omega) \end{aligned} \tag{60}$$

and for the angular velocity as:

$$\dot{s}_\omega = -K_\omega s_\omega - \tilde{W}_\omega^T h_\omega - \epsilon_\omega - \tilde{\theta}_\omega u_\omega \tag{61}$$

The dynamics of the augmented errors defined in (39) and (45) affect the overall error dynamics of the closed-loop system. By assuming fast responses of the first-order low-pass filters, that is faster than the error surfaces and system dynamics, one can simply ignore their effects when designing the control system. Thus, the dynamics of the error surfaces becomes

$$\dot{s}_R \approx -K_R s_R + R_d^T R s_\omega \tag{62}$$

$$\dot{s}_\omega = -K_\omega s_\omega - \tilde{W}_\omega^T h_\omega - \epsilon_\omega - \tilde{\theta}_\omega u_\omega \tag{63}$$

Now, Let us define a Lyapunov function as:

$$V = \frac{1}{2} \left(s_R^T s_R + s_\omega^T s_\omega + \tilde{\theta}_\omega^T \Gamma_\omega^{-1} \tilde{\theta}_\omega + \tilde{W}_\omega^T \gamma_\omega^{-1} \tilde{W}_\omega \right) \tag{64}$$

where Γ_ω and γ_ω are positive matrices. Driving its derivative yields:

$$\begin{aligned} \dot{V} &= -K_R \|s_R\|^2 - K_\omega \|s_\omega\|^2 + s_R^T R_d^T R s_\omega - s_\omega^T \epsilon_\omega \\ &+ \tilde{W}_\omega^T (h_\omega s_\omega^T - \gamma_\omega^{-1} \dot{\tilde{W}}_\omega) + \tilde{\theta}_\omega (u_\omega s_\omega^T - \Gamma_\omega^{-1} \dot{\tilde{W}}_\omega) \end{aligned} \tag{65}$$

Similarly, an adaptation function can be used to guarantee the control signal's stability and avoid singularity as follows:

$$\dot{\hat{W}}_\omega = \gamma_\omega h_\omega s_\omega^T \tag{66}$$

$$\alpha_\omega = \Gamma_\omega u_\omega s_\omega^T \tag{67}$$

$$\alpha_\omega = \Gamma_\omega u_\omega s_\omega^T \dot{\hat{\theta}}_\omega = \begin{cases} -\eta_\omega \alpha_\omega & a_\omega > 0 \\ -\eta_\omega \alpha_\omega & \alpha_\omega \leq 0 \ \& \ \hat{\theta}_\omega > \bar{\theta}_\omega \\ -\eta_\omega & \alpha_\omega \leq 0 \ \& \ \hat{\theta}_\omega \leq \bar{\theta}_\omega \end{cases} \tag{68}$$

where the estimated parameter θ_ω has a lower limit represented by $\bar{\theta}_\omega$. The initial values of θ_ω must be selected as $\hat{\theta}_\omega(0) > (\bar{\theta}_\omega)$. The derivative of the Lyapunov function can be rewritten as

$$\begin{aligned} \dot{V} &= -K_R \|s_R\|^2 - K_\omega \|s_\omega\|^2 + s_R^T R_d^T R s_\omega - s_\omega^t \epsilon_\omega \\ &\leq K_R \|s_R\|^2 - K_\omega \|s_\omega\|^2 - s_\omega^t \epsilon_\omega \end{aligned} \tag{69}$$

RBF-NN guarantees that the approximation error represented by ϵ_ω is adequately small enough and limited i.e. ($\dot{V} < 0$). This gives the surface error s_R and s_ω an exponential convergence rate which results in tracking the desired signal, i.e. $R \rightarrow R_d$

5 Results

In this section, we demonstrate the results of the proposed controller through simulation software and experimentally using a testbed. The purposes of the simulations are to show the overall performance of the proposed controller in an ideal environment, while the experimental experiments show the feasibility of the proposed control in real-life implementation.

5.1 Numerical simulation

The simulations are carried out through MATLAB SIMULINK[®] software. The simulation demonstrates the performance of the proposed global robust DSC controller in terms of accuracy and convergence rate under the presence of uncertainties and disturbances. The quadrotor system parameters carried out through the simulation tests are shown in Table 2 while the used control parameters are shown in Table 3.

Here, the simulation illustrates the behavior of the proposed controller in the presence of unmodeled dynamics, parameter uncertainties, and external disturbance. The

Table 2 Plant parameters and inputs saturation

Parameter	Value	Unit	Parameter	Value	Unit
$I_x x, I_y y$	0.01	N m s ²	m	1	kg
$I_z z$	0.1	N m s ²	K_b	5×10^{-4}	N s ² rad ⁻²
l	0.5	m	K_d	0.1	m

Table 3 Control parameters in each domain

Parameter	Value	Parameter	Value
τ_R	40	γ	0.1
τ_ω	50	Γ	0.01
K_R	$diag(13, 13, 19)$	λ_p	$diag(5, 5, 10)$
K_ω	$diag(15, 15, 19)$	K_p	$diag(4, 4, 8)$

desired trajectory consists of four flight behaviors: ascending (take-off), stand still (hovering), path following and descending (landing). At the vertical take-off behavior, the system is commanded to start raising until it reaches a certain altitude with a slope of 1 m/s while the system is subjected to uncertainties in its parameters and dynamics. The required performance of the system for this stage includes showing a good error convergence with proper estimations and handling of the system uncertainties. At the hovering and path following stages, the system will be subjected to external disturbances, i.e. wind gust, on all of its axes. At both stages, the performance of the system is required to show a good disturbance rejection while following the specified trajectory for each stage. This includes demonstrating a good error convergence and an adequate estimation of the external disturbances. Finally, at the fourth stage, the landing test, the quadrotor is commanded to start descending from a specific

Fig. 3 Desired and actual trajectories

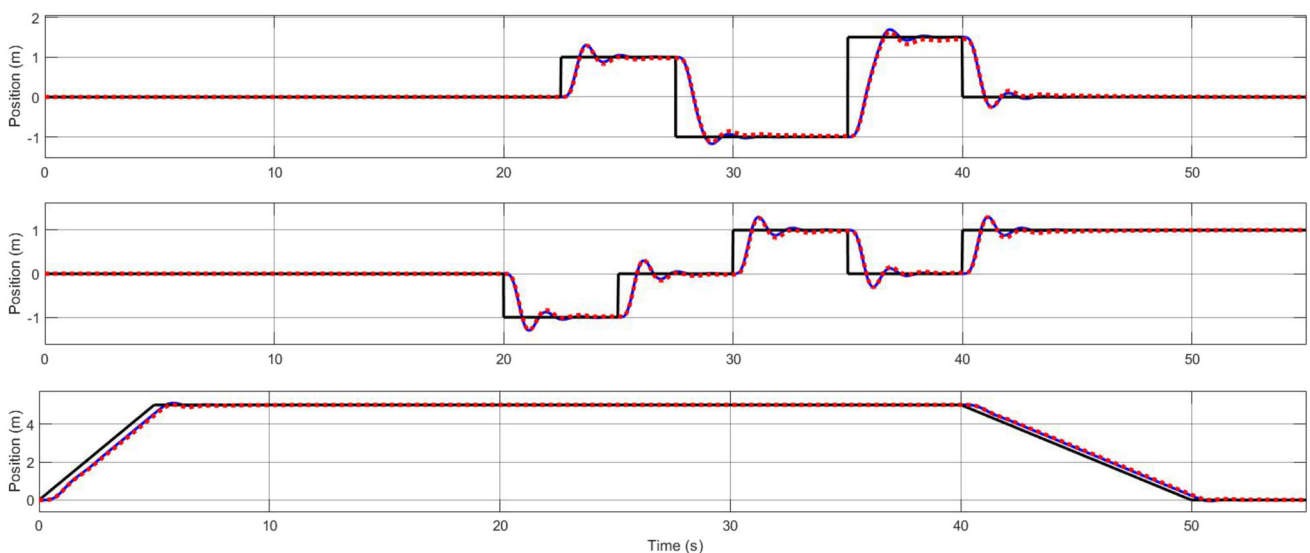
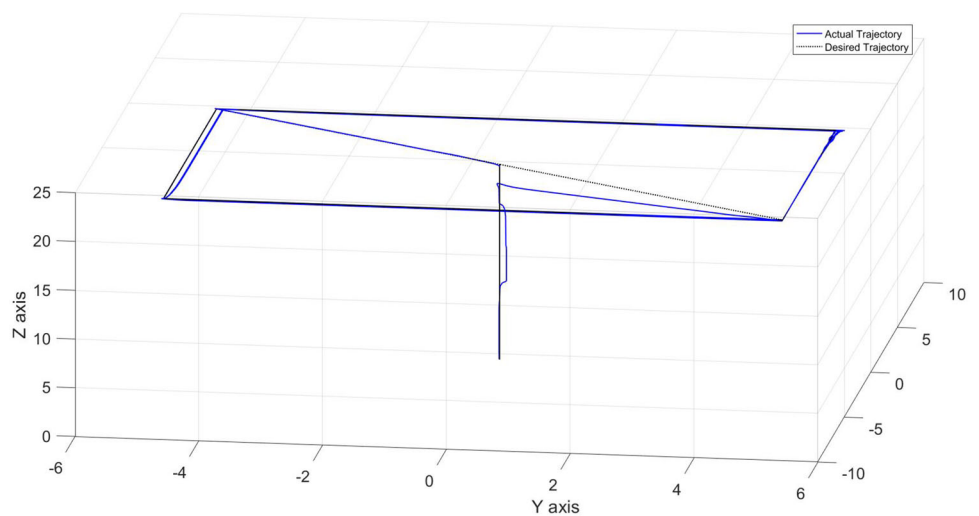


Fig. 4 Position command and response in x-, y-, z-axis; command in black, desired in red and actual in blue

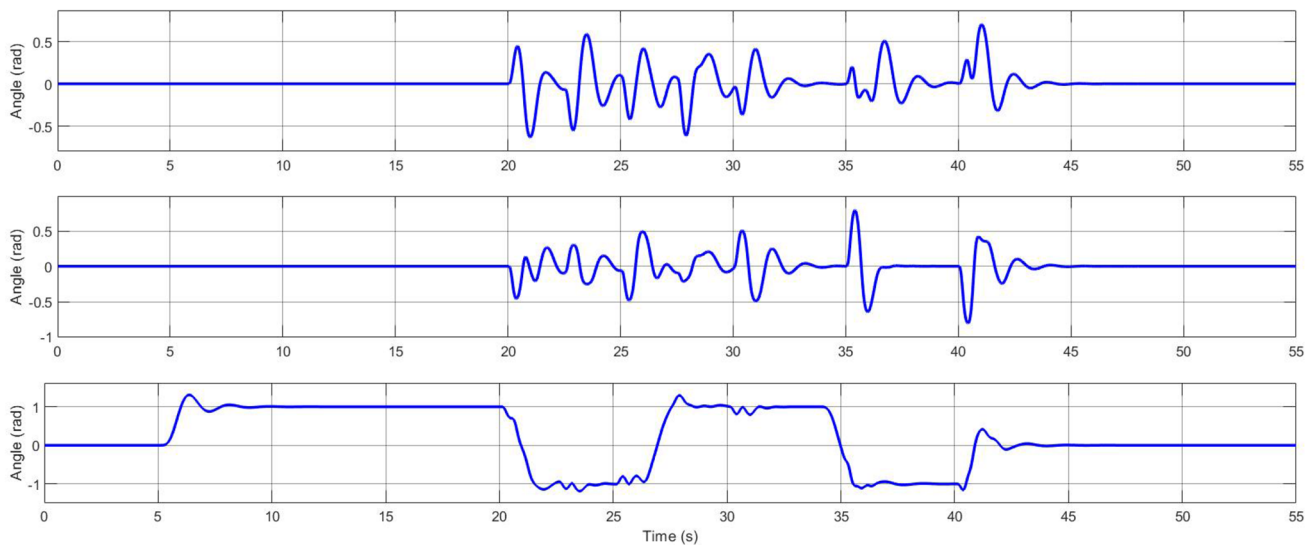


Fig. 5 Euler-angle response in roll, pitch and yaw domain

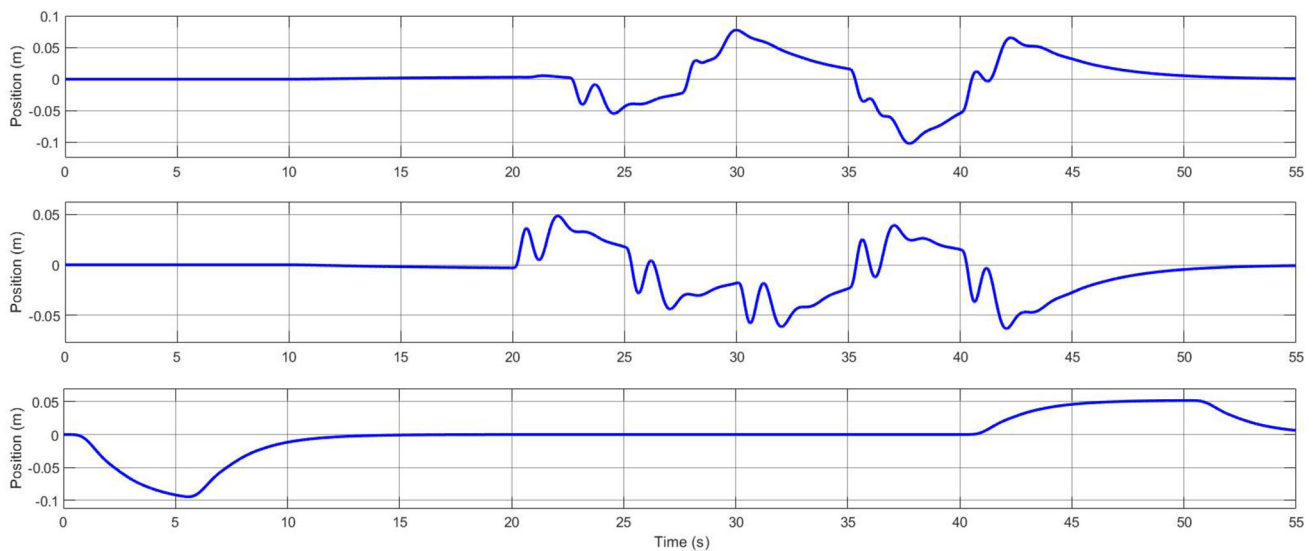


Fig. 6 Tracking error between in x -, y - and z -axis

altitude until it reaches the ground with a slower sloop, 0.5 m/s. The quadrotor needs to land safely and precisely. Figure 3 shows the path of the desired and actual trajectories of the carried tests.

Figure 4 shows the position signal of the command, desired and actual signals. Figure 5 shows the Euler-angle responses. The results show a perfect tracking of the command signals. Due to the uncertainties in the system model, the designed adaptation mechanism enhances the system behavior, continuously. The quadrotor was able to perform all required behaviors despite the considered challenges.

Detailed navigational of the position and Euler angles are shown in Figs. 6 and 7, respectively. It is clearly seen that the proposed control system achieved an adequate tracking of

the command signals, i.e. all tracking error have converged to zero, despite the external disturbances and model uncertainties.

To show the efficiency of the proposed control algorithm, several comparison tests have been carried out. Two algorithms have been selected for the comparison; a PID-based technique, similar to the one typically used in the off-the-shelf controller, and the control algorithm proposed by T. Lee in [16]. The comparison tests are carried out by generating a train of pulses as a command signal and then calculating the integral of absolute error (IAE) for each algorithm. Two sets of command signals are generated; one with a range between $\pm \pi/3$ and the other between $\pm 2\pi/3$. For each set, the tests were carried out with and without applying external

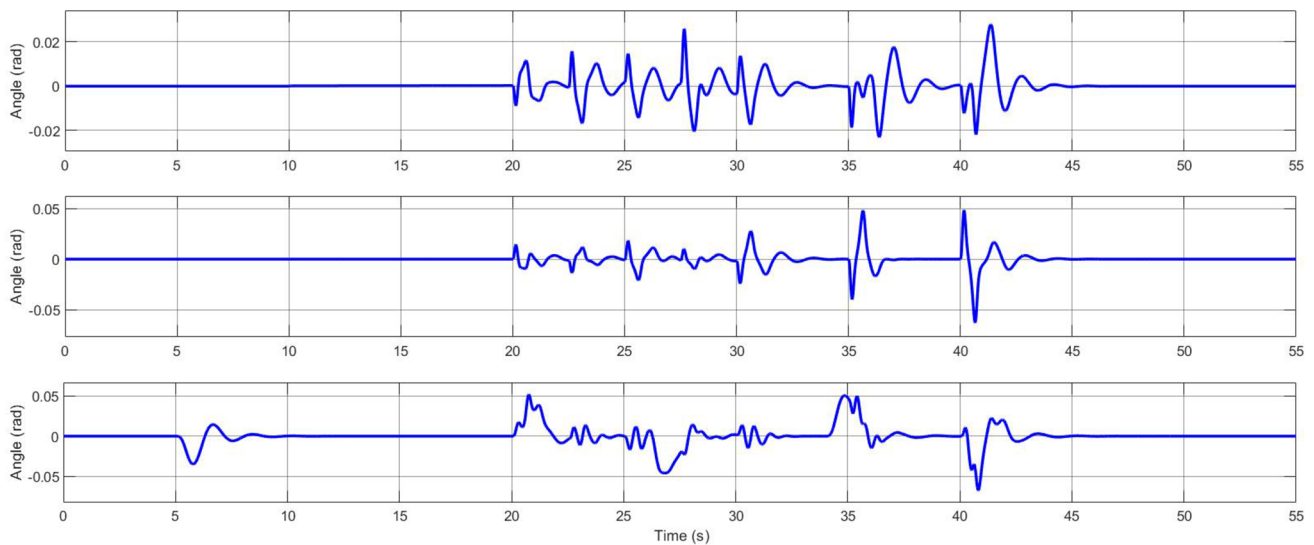


Fig. 7 Tracking error in roll, pitch and yaw domain

Table 4 Attitude comparison tests

Method	Command range [$-\pi/3$ $\pi/3$]		Command range [$-2\pi/3$ $2\pi/3$]	
	Without dist.	With dist.	Without dist.	With dist.
PID	5400	6900	12,323	18,230
T. Lee	2320	2711	2497	2912
Proposed	482	650	645	946

disturbances. The results of the comparison tests are shown in Table 4. In this table, lower numbers of IAE indicate better performance and smaller tracking error.

5.2 Experimental test

To experimentally validate the proposed control system, we have developed a testbed system. The testing system uses Robot Operating System (ROS) environment to smoothly integrate required sensors, estimation and control algorithms, and communication with the system. The testbed system consists mainly of five parts, a mechanical skeleton, sensors, actuators, a processor and a ground station. Figure 8 shows a graphical representation of the overall architecture of the testbed system. The testbed is developed to resemble the rotation dynamics of a quadrotor vehicle. The body frame of the quadrotor is mounted on a 3-DOF pivot joint, which allows the quadrotor to rotate about its x -, y - and z -axes. The mechanism allows the frame to have a maximum of 35 degrees in roll and pitch angles and 360 degrees in yaw angle. The platform is supported by a thick aluminum base with a 40 cm long and 5 cm thickness pipe. The system is mounted on the ceiling to reduce the ground effects generated by the

quadrotor vortex. The testbed consists of four arms with 30 cm in length. At the end of each arm, a brushless DC motor is attached with a propeller and controlled using an Electrical Speed Controller (ESC) unit. To measure the vehicle states, the testbed is supported with various on-board sensors. In our setup, we use two Inertial Measurement Unit (IMU) sensors and three joint encoders. Each IMU sensor consists of 3-axis accelerometers, 3-axis gyroscopes, and 3-axis magnetometers. The reason for using two IMU sensors is mainly for improving the measurement accuracy and the system redundancy. The joint encoders can accurately measure the platform's rotation angles which are used to obtain the orientation of the testbed with a relatively low noise level and micro degree accuracy at up to 1 kHz. In our experiments, the encoder readings are used for the purpose of estimation comparisons as they serve as the ground truth measurements. In terms of onboard computing power, the testbed contains a Raspberry pi 3@microprocessor board that runs up to 1.2 GHz and capable of running the closed-loop controller algorithm. The board is supported with an input-output extension that allows it to control 16 PWM channels with 12bit resolution used to control the ESC. We use a PC, serves as the ground station, running the ROS environment on Ubuntu 16, which generates the command orientation signals and displays the testbed states. We have connected the on-board microprocessor to the ground station through WiFi. The communication gives the user the ability to control the testbed's orientation through a graphical user interface (GUI) and visualize the vehicle states using GAZEBO software. During this experiment, the data interchange was running at 100 Hz. Figure 9 shows the testbed fixed to the ceiling.

In the experimental test, we have implemented the control algorithm using the C++ language. All control processes

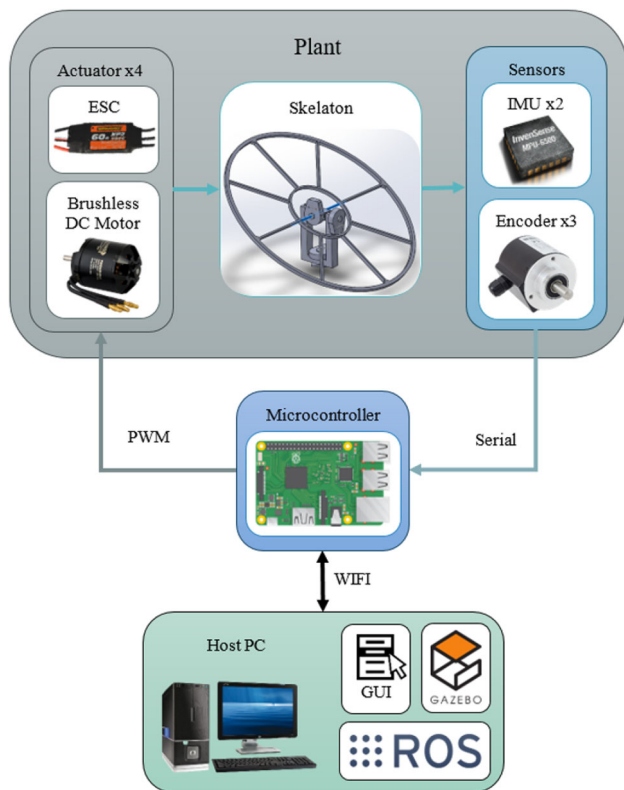


Fig. 8 A graphical representation of the overall architecture of the experimental setup



Fig. 9 Testbed while fixed to the ceiling

and measurements were performed on-board. In this experiment, we have generated pulse inputs as a desired command signal in roll and yaw domains. The initial weights of the Neural-Network parameters have been randomly selected, as the proposed RBF-NN does not require any prior training for its weights.

Figure 10 shows the output response of the setup for the Euler angles. The result shows a good convergence for the yaw angle and acceptable convergence for the roll and pitch angles. The output response of the roll and pitch angles are designed to be faster than the yaw angle, as can be seen from the selected control parameters defined in Table 3.

Figure 11 illustrates the tracking errors for the Euler angles. The error signal reached zero within an acceptable time. The response of the roll and pitch domains is less

smoother than the yaw domain that is due to the subsystem coupling effect, where the rotation of one domain generates a disturbance signal on the other one. Figure 12 shows the input signal generated by the proposed control system for each angle. The control signals are noisy due to the usage of embedded sensors in the feedback loop. The signal limitation, seen in the control signal of the yaw domain, is a software limitation designed so the relation in Eq. (5) is fully defined. This limitation is manually designed and depends on the system configuration and parameters.

6 Conclusions

A global robust control system with an adaptation mechanism is designed for quadrotors in the presence of underactuation, external disturbances and model uncertainties. The proposed attitude controller is based on dynamic surface control and developed directly on the special Euclidean group with a region of attraction covering the configuration space globally. Moreover, the proposed controller uses adaptive RBF-NN to overcome the model uncertainties and external disturbances. The RBFNN is based on a single hidden layer to reduce the computational complexity of the control system. This will allow the controller to maintain both dynamics stability and tracking performance while avoiding the singularity associated with orientation representations. The performance of the proposed controller was compared to two other approaches; a traditional PID control used typically in off-the-shelf solutions and a state-of-the-art solution proposed by T. Lee in [16]. The results show that the proposed controller outperforms those two approaches in terms of both accuracy and convergence rate. The stability of the control system is proven using Lyapunov functions, and its performance has been validated by both simulation and experiments.

The performance of the overall control system has been validated by both simulation and experiments using a 3-DOF testbed. The use of this testbed, instead of a regular UAV, has offered two very important advantages for this research. First, we have been able to use encoders to measure and record the angle and rate of yaw, pitch, and roll much more accurately than what we could have been measured using the IMU of a regular UAV, which is typically not an impressively accurate sensor in the price range of UAV applications. Second, and perhaps more importantly, we have been able to subject the testbed to much greater disturbances (for example a physical impact using a long stick) rather than what we could safely apply to a freely flying UAV. In other words, the use of the testbed made it possible for us to demonstrate the stability and robustness of the proposed dynamic surface control method in a much broader application domain without facing the risk of crashing the experimental setup over and over again.

Fig. 10 Testbed response in roll, pitch and yaw domain

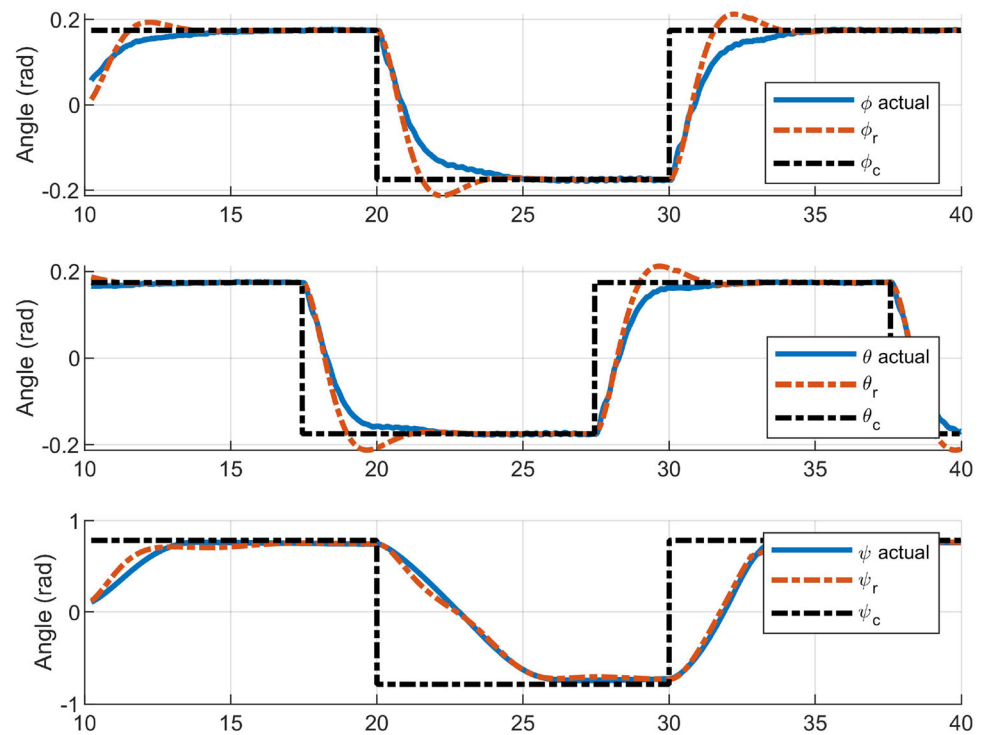


Fig. 11 Tracking error in Euler angles

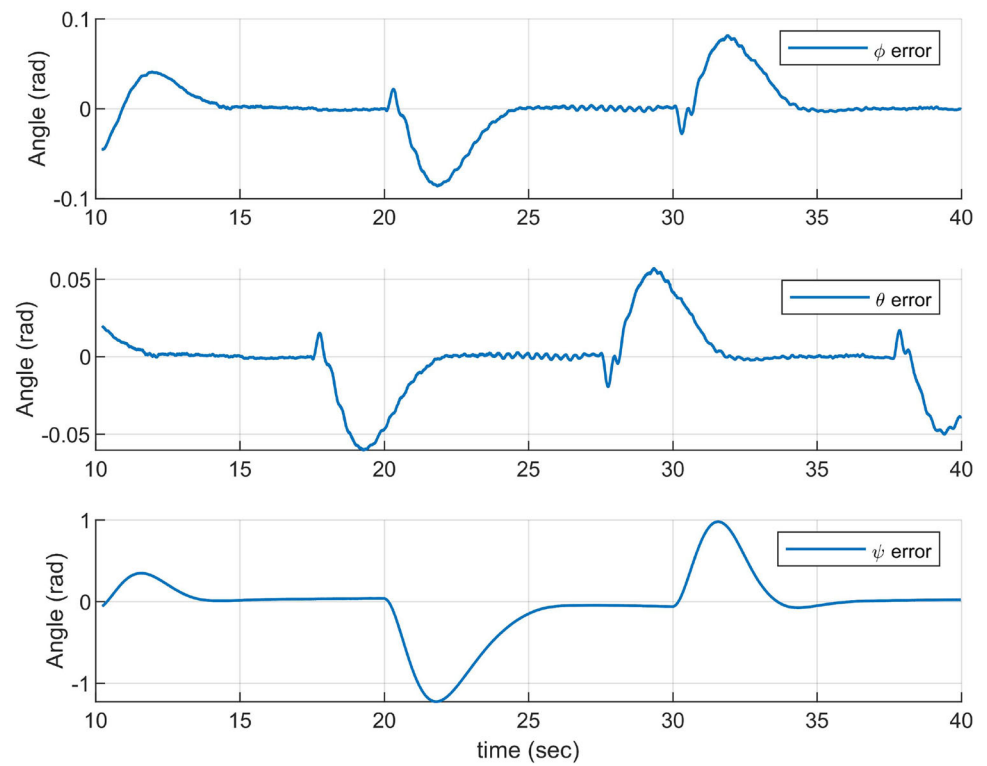
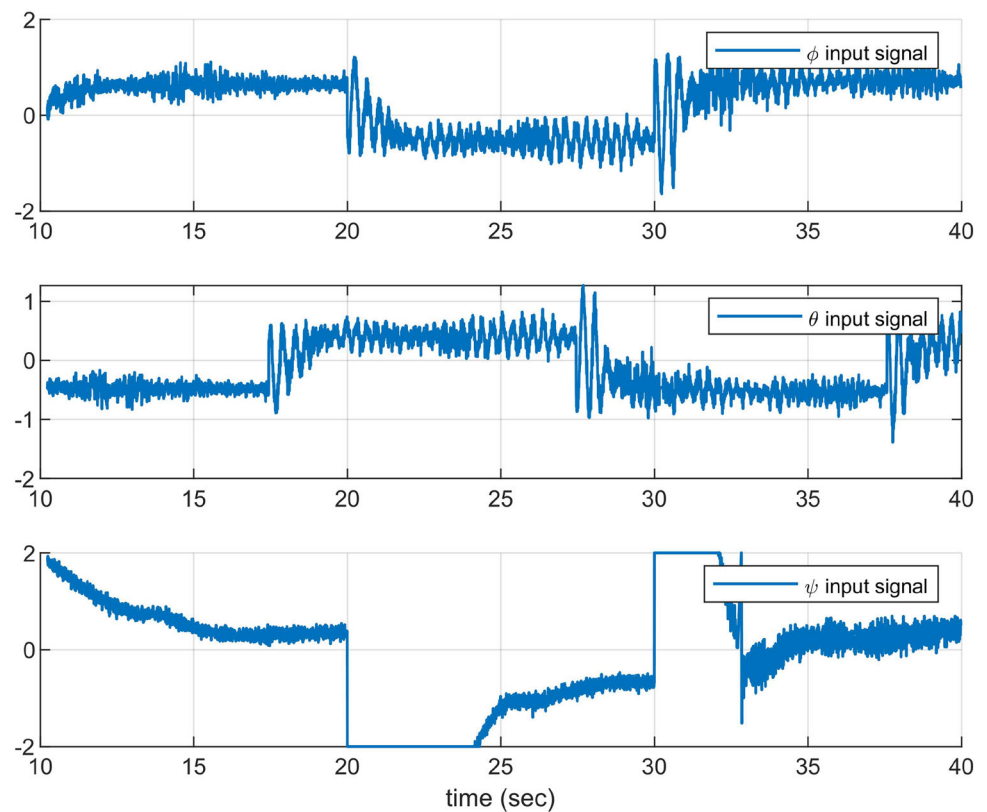


Fig. 12 Control signal



The study of the performance of the control system in real-flight experiments will be the topic of further research. In an uncontrolled environment, the system may be subjected to unexpected conditions. Thus, one future direction can be to validate the proposed algorithm against the unmodeled dynamics in 6 DOF. Even though the attitude dynamics have more influence on the stability of the system in comparison with the translation dynamics, real-flight validation tests are still required. Another future direction can be in terms of simplifying the procedure of identifying the control parameters. Although the proposed control system uses adaptive techniques to compensate for system uncertainties, several tests are still required in order to find suitable control parameters. One solution to that could be the use of the model reference adaptive control in conjunction with the dynamic surface control in the attitude controller. This will contribute to having a consistent system behavior regardless of the actual system parameters.

References

- Máthé K, Buşoniu L (2015) Vision and control for UAVs: a survey of general methods and of inexpensive platforms for infrastructure inspection. *Sensors* 15(7):14887–14916. <https://doi.org/10.3390/s150714887>
- Heredia G, Jimenez-Cano AE, Sanchez I, Llorente D, Vega V, Braga J, Acosta JA, Ollero A (2014) Control of a multirotor outdoor aerial manipulator. In: IEEE/RSJ international conference on intelligent robots and systems, pp 3417–3422. <https://doi.org/10.1109/IROS.2014.6943038>
- Emran BJ, Dias J, Seneviratne L, Cai G (2015) Robust adaptive control design for quadcopter payload add and drop applications. In: Chinese control conference (CCC), vol 34, pp 3252–3257. <https://doi.org/10.1109/ChiCC.2015.7260141>
- Emran BJ, Tannant DD, Najjaran H (2017) Low-altitude aerial methane concentration mapping. *Remote Sens* 9(8):823. <https://doi.org/10.3390/rs9080823>
- Roza A, Maggiore M (2014) A class of position controllers for underactuated VTOL vehicles. *IEEE Trans Autom Control* 59(9):2580–2585. <https://doi.org/10.1109/TAC.2014.2308609>
- Emran BJ, Najjaran H (2017) Adaptive neural network control of quadrotor system under the presence of actuator constraints. In: IEEE international conference on systems, man, and cybernetics (SMC), pp 2619–2624
- Xiong J-J, Zhang G-B (2017) Global fast dynamic terminal sliding mode control for a quadrotor UAV. *ISA Trans* 66:233–240. <https://doi.org/10.1016/j.isatra.2016.09.019>
- Zheng E-H, Xiong J-J, Luo J-L (2014) Second order sliding mode control for a quadrotor UAV. *ISA Trans* 53(4):1350–1356. <https://doi.org/10.1016/j.isatra.2014.03.010>
- Dydek ZT, Annaswamy AM, Lavretsky E (2013) Adaptive control of quadrotor UAVs: a design trade study with flight evaluations. *IEEE Trans Control Syst Technol* 21(4):1400–1406. <https://doi.org/10.1109/TCST.2012.2200104>
- Liu H, Wang X, Zhong Y (2015) Quaternion-based robust attitude control for uncertain robotic quadrotors. In: IEEE transactions on industrial informatics, vol 11, pp 406–415. <https://doi.org/10.1109/TII.2015.2397878>
- Das A, Lewis F, Subbarao K (2009) Backstepping approach for controlling a quadrotor using Lagrange form dynamics. *J*

- Intell Robot Syst Theory Appl 56(1–2):127–151. <https://doi.org/10.1007/s10846-009-9331-0>
12. Das A, Lewis FL, Subbarao K (2011) Sliding mode approach to control quadrotor using dynamic inversion. In: Bartoszewicz A (ed) Challenges and paradigms in applied robust control, Chap. 1. InTech, London, pp 3–24. <https://doi.org/10.5772/16599>
 13. Choi YC, Ahn HS (2015) Nonlinear control of quadrotor for point tracking: actual implementation and experimental tests. IEEE/ASME Trans Mechatron 20(3):1179–1192. <https://doi.org/10.1109/TMECH.2014.2329945>
 14. Ha C, Zuo Z, Choi FB, Lee D (2014) Passivity-based adaptive backstepping control of quadrotor-type UAVs. Robot Auton Syst 62(9):1305–1315. <https://doi.org/10.1016/j.robot.2014.03.019>
 15. Emran BJ, Najjaran H (2017) Switching control of quadrotor with adaptation mechanism. In: IEEE International conference on systems, man, and cybernetics (SMC)—conference proceedings, Budapest, pp 4872–4877. <https://doi.org/10.1109/SMC.2016.7845000>
 16. Goodarzi F a, Lee D, Lee T (2015) Geometric adaptive tracking control of a quadrotor unmanned aerial vehicle on SE(3) for agile maneuvers. J Dyn Syst Meas Control 137(9):091007. <https://doi.org/10.1115/1.4030419>. arXiv:1411.2986v1
 17. Cabecinhas D, Naldi R, Silvestre C, Cunha R, Marconi L (2016) Robust landing and sliding maneuver hybrid controller for a quadrotor vehicle. IEEE Trans Control Syst Technol 24(2):400–412. <https://doi.org/10.1109/TCST.2015.2454445>
 18. Branicky MS (1998) Multiple Lyapunov functions and other analysis tools for switched and hybrid systems. IEEE Trans Autom Control 43(4):475–482. <https://doi.org/10.1109/9.664150>
 19. Goodarzi F A, Lee D, Lee T (2014) Geometric adaptive tracking control of a quadrotor UAV on SE(3) for agile maneuvers. J Dyn Syst Meas Control 137(9):091007. <https://doi.org/10.1115/1.4030419>. arXiv:1411.2986
 20. Min B-C, Hong J-H, Matsun ET (2011) Adaptive robust control (ARC) for an altitude control of a quadrotor type UAV carrying an unknown payloads. In: International conference on control, automation and systems, pp 1147–1151
 21. Zhao B, Xian B, Zhang Y, Zhang X (2015) Nonlinear robust adaptive tracking control of a quadrotor UAV via immersion and invariance methodology. IEEE Trans Ind Electron 62(5):2891–2902. <https://doi.org/10.1109/TIE.2014.2364982>
 22. Nicol C, MacNab CJB, Ramirez-Serrano A (2011) Robust adaptive control of a quadrotor helicopter. Mechatronics 21(6):927–938. <https://doi.org/10.1016/j.mechatronics.2011.02.007>
 23. Peng C, Bai Y, Gong X, Gao Q, Zhao C, Tian Y (2015) Modeling and robust backstepping sliding mode control with adaptive RBFNN for a novel. IEEE/CAA J Autom Sin 2(1):56–64. <https://doi.org/10.1109/JAS.2015.7032906>
 24. Emran BJ, Najjaran H (2018) A review of quadrotor: an underactuated mechanical system. Annu Rev Control 46:165–180. <https://doi.org/10.1016/J.ARCONTROL.2018.10.009>
 25. Song B, Hedrick JK (2011) Dynamic surface control of uncertain nonlinear systems, an LMI approach. Springer, Berlin
 26. Liu Z, Hedrick K (2016) Dynamic surface control techniques applied to horizontal position control of a quadrotor. In: International conference on system theory, control and computing (ICSTCC). IEEE, pp 138–144
 27. Hua M-D, Hamel T, Morin P, Samson C (2013) Introduction to feedback control of underactuated VTOL vehicles: a review of basic control design ideas and principles. IEEE Control Syst 33(1):61–75. <https://doi.org/10.1109/MCS.2012.2225931>
 28. Mahony R, Kumar V, Corke P (2012) Multirotor aerial vehicles: modeling, estimation, and control of quadrotor. IEEE Robot Autom Mag 19(3):20–32. <https://doi.org/10.1109/MRA.2012.2206474>
 29. Liu J (2013) Radial basis function (RBF) neural network control for mechanical systems design, analysis and Matlab simulation, vol.1, SpringerLink ebooks. arXiv:1011.1669v3, <https://doi.org/10.1017/CBO9781107415324.004>