Research article

Finite-time trajectory tracking control of quadrotor UAV via adaptive RBF neural network with lumped uncertainties

Rui Ma*, **Jinjin Han and Li Ding**

College of Mechanical Engineering, Jiangsu University of Technology, Changzhou 213001, China

* **Correspondence:** Email: nuaamarui@163.com.

Abstract: The trajectory tracking control of the quadrotor with model uncertainty and time-varying interference is studied. The RBF neural network is combined with the global fast terminal sliding mode (GFTSM) control method to converge tracking errors in finite time. To ensure the stability of the system, an adaptive law is designed to adjust the weight of the neural network by the Lyapunov method. The overall novelty of this paper is threefold, 1) Owing to the use of a global fast sliding mode surface, the proposed controller has no problem with slow convergence near the equilibrium point inherently existing in the terminal sliding mode control. 2) Benefiting from the novel equivalent control computation mechanism, the external disturbances and the upper bound of the disturbance are estimated by the proposed controller, and the unexpected chattering phenomenon is significantly attenuated. 3) The stability and finite-time convergence of the overall closed-loop system are strictly proven. The simulation results indicated that the proposed method achieves faster response speed and smoother control effect than traditional GFTSM.

Keywords: UAV; trajectory tracking control; global fast terminal sliding mode control; neural network-based method; adaptive control

1. Introduction

Nowadays, multi-rotor unmanned aerial vehicle (UAVs) plays an important role in many commercial applications, such as air pollution monitoring, rescue missions, precision agriculture, retail delivery, as well as academic research and military action [1]. The quadrotor is one of the most widely used classes of UAVs, which has the advantages of hovering, vertical take-off and landing, simple structure, and low cost. The quadrotor has four actuators and six degrees of freedom, and its dynamic model has the characteristics of strong coupling, under-actuated and susceptible to external disturbances [2]. Therefore, the design of high-performance flight controllers for quadrotors is a challenging task, which has received extensive attention in the academic circle.

In recent years, significant research has been done on the trajectory-tracking control of quadrotors. The conventional approach to deal with the nonlinearity is simplifying the model to linear equivalents by using dynamic inversion, feedback linearization, gain-scheduling, or Taylor's approximation, etc. However, these control methods are susceptible to model uncertainties and time-varying interference [3]. In order to achieve better robustness and anti-interference performance, many advanced control algorithms have been used to design flight controllers of quadrotors, for instance, neural network-based optimal mixed H_2/H_∞ control [4], terminal sliding mode control [5], backstepping control [6], adaptive fuzzy quantized control [7] and active disturbance rejection control [8].

Although these advanced control strategies improved the flight performance of the quadrotor, few results are concerned with the convergence rate of the system output. In practice, the real-time performance of the control system is vital to the flight stability of the quadrotor, and the convergence speed should be one of the key indexes for the flight control system [9]. The control methods concerned with the convergence rate of system output include Hybrid finite-time control [10], finite-time adaptive sliding mode tracking control [11], finite-time Lyapunov theory [12], global fast terminal sliding mode (GFTSM) control [13, 14], etc. The GFTSM control method can make the system state variables converge to the equilibrium point in a limited time. However, traditional GFTSM control still has some limitations and challenges. Firstly, accurate model information is required in the calculation of the equivalent control which restricts GFTSM available to applications. Especially, some parameters are difficult to obtain because of the complex dynamics of multi-rotor UAVs. Secondly, the GFTSM control needs the upper bound of the external disturbance, which is hard to estimate in real applications. Thirdly, the system chattering is inevitable because of the discontinuous item in the switch control. The chattering will affect the stability of multi-rotor UAVs, which is not suitable for practical application.

This article focuses on the GFTSM control strategy of quadrotor UAVs based on the RBF neural network (GFTSM-RBF). This strategy has three important features: 1) The chattering phenomenon of GFTSM control can be effectively reduced by using a neural network to learn the unknown dynamics model and the upper bound of disturbance of UAV online. It also reduces the work of dynamics modeling and parameter identification before controller design. 2) The weights of the neural network are adjusted online according to the designed adaptive law to ensure the stability of the closed-loop system. 3) The equivalent control that usually requires precise model information of the system is computed directly using the RBF neural network. Therefore, the structure of the controller is simplified and the online calculation is reduced, which makes it affordable for practical applications.

The rest of the paper is arranged as follows. In Section 2, the dynamics model of a quadrotor UAV is briefly described, and a novel GFTSM-RBF controller design for trajectory tracking of the UAV is proposed. Simulation results and performance analysis are shown in Section 3. The work of this paper is summarized in the last section.

2. Method

2.1. Model formulation

The coordinate system and schematic of the quadrotor are shown in Figure 1. The body frame of the quadrotor $\{B\}$ is attached to the center of mass and $\{E\}$ is the reference frame. According to the

direction of rotation, the rotors of the quadrotor can be divided into two groups, namely (1,3) and (2,4). These two sets of rotors produce lift force $F_1, \ldots, F_4$ and neutralize the counter-torque. The rotation of the body is controlled by the speed difference between rotors. The rolling motion of the quadrotor is accomplished by changing the speed of rotor 2 or 4. The pitching motion of the quadrotor is achieved by adjusting the speed of rotor 1 or 3. The yaw rotation of the quadrotor is accomplished by the reverse moment difference between the two pairs of rotors (1,3) and (2,4). The vertical motion is performed by increasing or decreasing the total speed of the rotors [15].
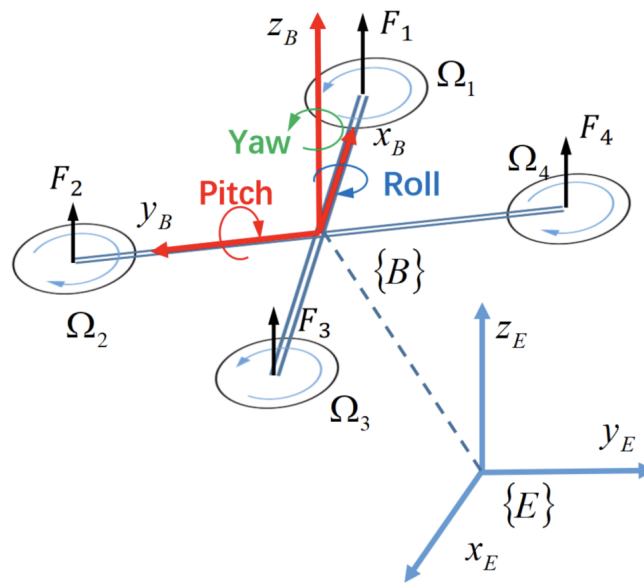

Figure 1. The schematic diagram of the quadrotor.

Assuming that the fuselage structure and rotors of the quadrotor are rigid, the rotational and translation movement of the quadcopter can be obtained as follows [16]:

$$
\begin{cases}
\ddot{\phi} = \dot{\theta}\dot{\psi}a_1 + \dot{\theta}a_2\Omega_r + b_2 U_\phi \\
\ddot{\theta} = \dot{\phi}\dot{\psi}a_3 + \dot{\phi}a_4\Omega_r + b_4 U_\theta \\
\ddot{\psi} = \dot{\theta}\dot{\phi}a_5 + b_6 U_\psi \\
m\ddot{x} = \left(s_\phi s_\psi + c_\phi s_\theta c_\psi \right) U_t \\
m\ddot{y} = \left(-s_\phi c_\psi + c_\phi s_\theta s_\psi \right) U_t \\
m\ddot{z} = mg - \left(c_\psi c_\phi \right) U_t
\end{cases}
\tag{2.1}
$$

where ϕ, θ and ψ represent the Euler angles (i.e., roll, pitch, and yaw angles). x, y and z are the positions of the center of gravity of the quadrotor, g is the gravity, m is the total mass of the quadrotor, I_r is the inertia of the rotor. $\Omega_r = \Omega_4 + \Omega_3 - \Omega_2 - \Omega_1$. a_i and $b_i (i = 1, \ldots, 6)$ are constants, which given by, $a_1 = \left(I_y - I_z \right)/I_x$, $a_2 = I_r/I_x$, $a_3 = (I_z - I_x)/I_y$, $a_4 = I_r/I_y$, $a_5 = \left(I_x - I_y \right)/I_z$, $b_2 = 1/I_x$, $b_4 = 1/I_y$, $b_6 = 1/I_z$. U_t is the total thrust, U_ϕ, U_θ, U_ψ are the torques in the ϕ, θ and ψ direction of rotation,

respectively. The control allocation is:

$$\begin{cases} U_t = b\left(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2\right) \\ U_\phi = bl\left(\Omega_4^2 - \Omega_2^2\right) \\ U_\theta = bl\left(\Omega_1^2 - \Omega_3^2\right) \\ U_\psi = c\left(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2\right) \end{cases}$$

where l is the distance of the rotors on the diagonal. b is the lift coefficient. c is the drag coefficient. $\Omega_1, \ldots, \Omega_4$ is the rotational speed of each propeller.

Based on the model Eq (2.1) and taking the lumped disturbance into account, the quadrotor equations are formulated as

$$\dot{X} = f(X, U) + d(X, U) \tag{2.2}$$

where $X = [\phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}, z, \dot{z}, x, \dot{x}, y, \dot{y}]^T \in \mathbb{R}^{12}$ is the state vector, $U = \left[U_\phi, U_\theta, U_\psi, U_x, U_y, U_z\right]^T \in \mathbb{R}^6$ is the input, $f(X, U)$ is nonlinear functions, $d(X, U) = [d_1, d_2, \ldots, d_6]^T$ is the lumped disturbance vector on each degree of freedom of the quadrotor. Equation (2.2) can be expended as follows:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_4 x_6 a_1 + x_4 a_2 \Omega_r + b_2 U_1 + d_1 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = x_2 x_6 a_3 + x_2 a_4 \Omega_r + b_4 U_2 + d_2 \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = x_2 x_4 a_5 + b_6 U_3 + d_3 \\ \dot{x}_7 = x_8 \\ \dot{x}_8 = U_4 + d_4 \\ \dot{x}_9 = x_{10} \\ \dot{x}_{10} = U_5 + d_5 \\ \dot{x}_{11} = x_{12} \\ \dot{x}_{12} = U_6 + d_6 \end{cases} \tag{2.3}$$

where U_4, U_5, U_6 are the position's virtual control:

$$\begin{cases} U_4 = \left(s_{x_1} s_{x_5} + c_{x_1} s_{x_3} c_{x_5}\right) U_t / m \\ U_5 = \left(-s_{x_1} c_{x_5} + c_{x_1} s_{x_3} s_{x_5}\right) U_t / m \\ U_6 = g - \left(c_{x_5} c_{x_1}\right) U_t / m \end{cases} \tag{2.4}$$

To simplify the design of controller, it is assumed that the vehicle does not pass through singularities $(-\pi/2 < \phi < \pi/2, -\pi/2 < \theta < \pi/2 \text{ and } -\pi < \psi < \pi)$.

In order to track the position reference signal, the desired total thrust U_t and the attitude angles (x_{1d}, x_{3d}) can be achieved through Eq (2.4) as follows:

$$\begin{cases} x_{1d} = \arctan\left(\cos\theta_r \frac{U_4 \sin x_5 + U_5 \cos x_5}{U_6 - g}\right) \\ x_{3d} = \arctan\left(\frac{U_4 \cos x_5 + U_5 \sin x_5}{U_6 - g}\right) \\ U_t = m\sqrt{U_4^2 + U_5^2 + (U_6 + g)^2} \end{cases} \tag{2.5}$$

2.2. Controller design

The goal of this section is to propose the GFTSM-RBF control and use it to design a closed-loop controller for the quadrotor to track the desired trajectory $x_{1d}, x_{3d}, x_{5d}, x_{7d}, x_{9d}, x_{11d}$. The structure of the controller is presented in Figure 2. The altitude of the quadcopter is controlled by the total thrust U_t, and the rotation movement is controlled by U_1, U_2, U_3. The desired angles of roll x_{1d}, pitch x_{3d} and U_t is achieved through Eq (2.5). The desired yaw x_{5d} is used to control the heading through the yaw controller.
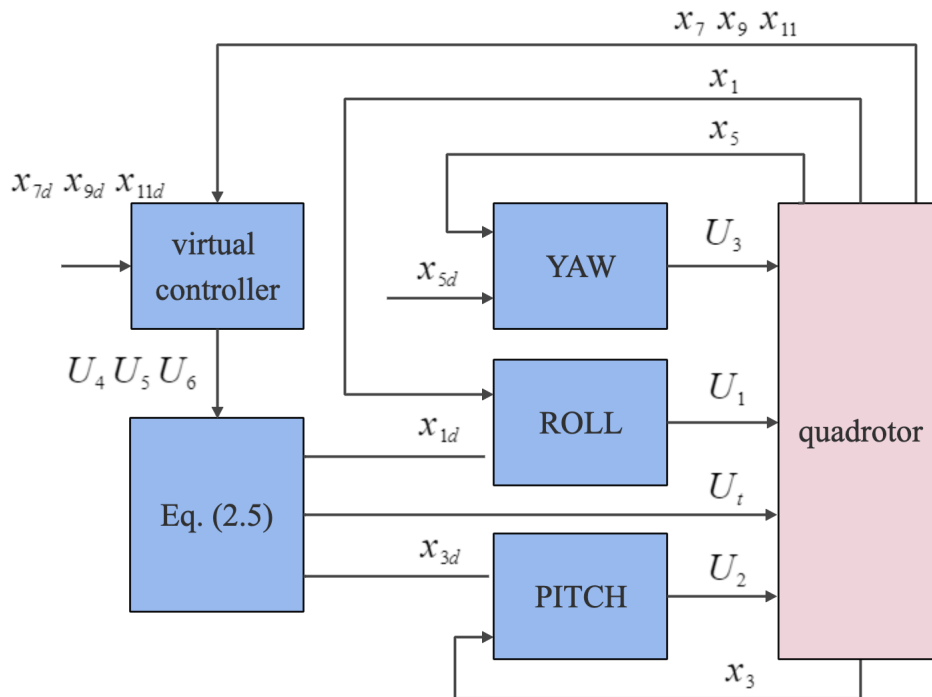
Figure 2. The structure of quadrotor flight controller designed by GFTSM-RBF control algorithm.

The controller integrates the GFTSM control and the adaptive RBF neural network in the frame of Lyapunov theory, as shown in Figure 3. There are four steps to compute the control as shown in Figure 4. The design steps are as follows:

Step 1. The tracking errors of the quadrotor are defined as

$$e_i = x_k - x_{kd}, (i = 1, \dots, 6, k = 2i - 1) \tag{2.6}$$

Thus, the derivative of the error are

$$\dot{e}_i = \dot{x}_k - \dot{x}_{kd} = x_{2i} - \dot{x}_{kd} \tag{2.7}$$

And the second time derivative of errors are as follows

$$\ddot{e}_i = \dot{x}_{2i} - \ddot{x}_{kd} \tag{2.8}$$

Step 2. Based on the GFTSM control theory [18], the sliding mode surfaces of the quadrotor position and attitude tracking control system is designed as:

$$s_i = \dot{e}_i + \alpha_i e_i + \beta_i e_i^{q_i/r_i} \tag{2.9}$$

where, α_i and β_i are positive constants. q_i and $r_i(q_i < r_i)$ are odd integers. The time derivative of the sliding surfaces are

$$\begin{aligned}
\dot{s}_i &= \ddot{e}_i + \alpha_i \dot{e}_i + \beta_i \frac{q_i}{r_i} e_i^{q_i/r_i-1} \dot{e}_i \\
&= \dot{x}_{2i} - \ddot{x}_{kd} + \alpha_i \dot{e}_i + \beta_i \frac{q_i}{r_i} e_i^{q_i/r_i-1} \dot{e}_i
\end{aligned} \tag{2.10}$$

Step 3. The Lyapunov candidate functions are designed as follows

$$\dot{V}_i^s = s_i \dot{s}_i$$

The derivative of Lyapunov candidate functions are obtained

$$\dot{V}_i^s = s_i \dot{s}_i \tag{2.11}$$

According to Eq (2.11), the control law of the quadrotor can be designed as

$$U_i = U_i^{eq} + U_i^{sw}, (i = 1, \ldots, 6) \tag{2.12}$$

where,

$$U_i^{sw} = -\lambda_i s_i - \delta_i s_i^{q_i/r_i}, (i = 1, \ldots, 6)$$

and,

$$\begin{aligned}
U_1^{eq} &= \frac{1}{b_2}\left(-d_1 - x_4 x_6 a_1 - x_4 a_2 \Omega_r + \ddot{x}_{1d} - \alpha_1 \dot{e}_1 - \beta_1 \frac{q_1}{r_1} e_1^{q_1/r_1-1} \dot{e}_1\right) \\
U_2^{eq} &= \frac{1}{b_4}\left(-d_2 - x_2 x_6 a_3 - x_2 a_4 \Omega_r + \ddot{x}_{3d} - \alpha_2 \dot{e}_2 - \beta_2 \frac{q_2}{r_2} e_2^{q_2/r_2-1} \dot{e}_2\right) \\
U_3^{eq} &= \frac{1}{b_6}\left(-d_3 - x_2 x_4 a_5 + \ddot{x}_{5d} - \alpha_3 \dot{e}_3 - \beta_3 \frac{q_3}{r_3} e_3^{q_3/r_3-1} \dot{e}_3\right) \\
U_4^{eq} &= -d_4 + \ddot{x}_{7d} - a_4 \dot{e}_4 - \beta_4 \frac{q_4}{r_4} e_4^{q_4/r_4-1} \dot{e}_4 \\
U_5^{eq} &= -d_5 + \ddot{x}_{9d} - a_5 \dot{e}_5 - \beta_5 \frac{q_5}{r_5} e_5^{q_5/r_5-1} \dot{e}_5 \\
U_6^{eq} &= -d_6 + \ddot{x}_{11d} - a_6 \dot{e}_6 - \beta_6 \frac{q_6}{r_6} e_6^{q_6/r_6-1} \dot{e}_6
\end{aligned}$$

where, U_i^{eq} is equivalent control and U_i^{sw} is the switch control. λ_i and δ_i are positive parameters. Take Eq (2.12) into Eq (2.10), the derivative of s_i are

$$\begin{aligned}
\dot{s}_1 &= b_2\left(-\lambda_1 s_1 - \delta_1 s_1^{q_1/r_1}\right) \\
\dot{s}_2 &= b_4\left(-\lambda_2 s_2 - \delta_2 s_2^{q_2/r_2}\right) \\
\dot{s}_3 &= b_6\left(-\lambda_3 s_3 - \delta_3 s_3^{q_3/r_3}\right) \\
\dot{s}_4 &= -\lambda_4 s_4 - \delta_4 s_4^{q_4/r_4} \\
\dot{s}_5 &= -\lambda_5 s_5 - \delta_5 s_5^{q_5/r_5} \\
\dot{s}_6 &= -\lambda_6 s_6 - \delta_6 s_6^{q_6/r_6}
\end{aligned}$$

Step 4. Adaptive RBF design: The control law U_i is difficult to calculate because U_i^{eq} contains the dynamic model and the disturbance term of the system. To obtain the control law, the RBF neural network is used to estimate the U_i^{eq}. Suppose a neural network with hidden layers [17] is adopted:

$$\hat{U}_i^{eq} = \hat{W}_i^T h_i(x_i), (i = 1, \ldots, 6)$$

where $\boldsymbol{x}_i = [s_i, x_{kd}]^T$, $(k = 2j - 1)$ is the input of the neural network, $\hat{\boldsymbol{W}}_i = [w_{i1}, w_{i2}, \ldots, w_{in}]^T$ is the weights of the neural networks, $\boldsymbol{h}_i(\boldsymbol{x}_i) = [h_{i1}, h_{i2}, \cdots, h_{in}]^T$ is the radial basis function and given by:

$$h_{ij} = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{c}_{ij}\|^2}{\sigma_{ij}^2}\right), (j = 1, 2, \ldots, n)$$

where σ_i is j th standard deviation. $\boldsymbol{c}_{ij}$ is the j th center vector.

In addition, the adaptive law of $\hat{\boldsymbol{W}}_i$ is

$$\dot{\hat{\boldsymbol{W}}}_i = -\frac{\eta_1}{m} s_i h_j \tag{2.13}$$

The final design of control law is

$$\hat{U}_i = \hat{U}_i^{eq} + U_i^{sw}, (i = 1, \ldots, 6) \tag{2.14}$$

Substitute Eq (2.14) into Eq (2.10), then the derivative of the sliding surface variable is

$$
\begin{aligned}
\dot{s}_1 &= b_2\left(-\lambda_1 s_1 - \delta_1 s_1^{q_1/r_1} + \hat{U}_1^{eq} - U_1^{eq}\right) \\
\dot{s}_2 &= b_4\left(-\lambda_2 s_2 - \delta_2 s_2^{q_2/r_2} + \hat{U}_2^{eq} - U_2^{eq}\right) \\
\dot{s}_3 &= b_6\left(-\lambda_3 s_3 - \delta_3 s_3^{q_3/r_3} + \hat{U}_3^{eq} - U_3^{eq}\right) \\
\dot{s}_4 &= -\lambda_4 s_4 - \delta_4 s_4^{q_4/r_4} + \hat{U}_4^{eq} - U_4^{eq} \\
\dot{s}_5 &= -\lambda_5 s_5 - \delta_5 s_5^{q_5/r_5} + \hat{U}_5^{eq} - U_5^{eq} \\
\dot{s}_6 &= -\lambda_6 s_6 - \delta_6 s_6^{q_6/r_6} + \hat{U}_6^{eq} - U_6^{eq}
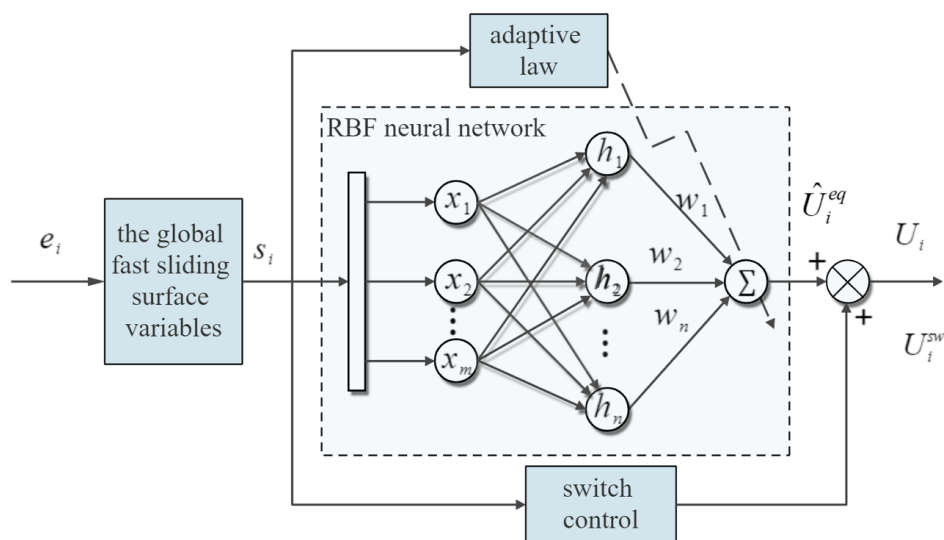\end{aligned}
\tag{2.15}
$$

Figure 3. The structure of the global fast terminal sliding mode (GFTSM) control based on the adaptive RBF neural network.
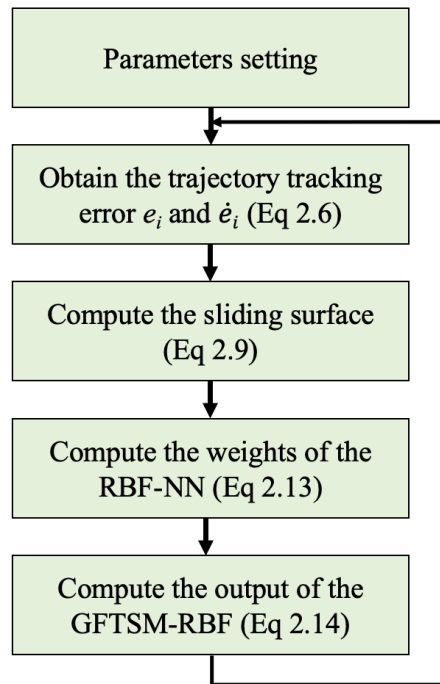
Figure 4. The brief flow chart of the GFTSM-RBF control strategy.

Theorem 2.1. *Considering the quadrotor UAV dynamics (Eq (2.3)) subject to lumped disturbance, there exists a set of control gains such that the proposed GFTSM (Eq (2.14)) for the quadrotor UAV can ensure that all signals in the closed-loop control system are bounded and the finite-time convergence of all the outputs to the designated trajectory can be guaranteed.*

Proof. Assume that the equivalent control U_i^{eq} can be estimated by the RBF neural network, and the minimum estimation error is $\varepsilon_i > 0$. The equivalent control U_i^{eq} becomes

$$U_i^{eq} = W_i h_i(x_i) + \varepsilon_i$$

where W_i is an ideal value of $\hat{W}_i$. Hence, the estimation error of U_i^{eq} is

$$\hat{U}_i^{eq} - U_i^{eq} = \hat{W}_i h_i(x_i) - W_i h_i(x_i) - \varepsilon_i = \tilde{W}_i h_i(x_i) - \varepsilon_i$$

where $\tilde{W}_i = \hat{W}_i - W_i$ is the error of weight vector, and submit it to Eq (2.15).

$$\begin{cases} \dot{s}_1 = b_2\left(-\lambda_1 s_1 - \delta_1 s_1^{q_1/r_1} + \tilde{W}_i h_i(x_i) - \varepsilon\right) \\ \dot{s}_2 = b_4\left(-\lambda_2 s_2 - \delta_2 s_2^{q_2/r_2} + \tilde{W}_i h_i(x_i) - \varepsilon\right) \\ \dot{s}_3 = b_6\left(-\lambda_3 s_3 - \delta_3 s_3^{q_3/r_3} + \tilde{W}_i h_i(x_i) - \varepsilon\right) \\ \dot{s}_4 = -\lambda_4 s_4 - \delta_4 s_4^{q_4/r_4} + \tilde{W}_i h_i(x_i) - \varepsilon \\ \dot{s}_5 = -\lambda_5 s_5 - \delta_5 s_5^{q_5/r_5} + \tilde{W}_i h_i(x_i) - \varepsilon \\ \dot{s}_6 = -\lambda_6 s_6 - \delta_6 s_6^{q_6/r_6} + \tilde{W}_i h_i(x_i) - \varepsilon \end{cases} \quad (2.16)$$

The following Lyapunov function is used

$$V_i^{ws} = \frac{1}{2}s_i^2 + \frac{1}{2\eta}\tilde{\boldsymbol{W}}_i^T \tilde{\boldsymbol{W}}_i$$

The time derivative of V_i^{ws} is

$$\dot{V}_i^{ws} = s_i\dot{s}_i + \frac{1}{\eta}\tilde{\boldsymbol{W}}_i^T \dot{\hat{\boldsymbol{W}}}_i \tag{2.17}$$

Take Eqs (2.13) and (2.16) into Eq (2.17), the derivative of the sliding surface variable can be written as (take $i = 1$ for example).

$$
\begin{aligned}
\dot{V}_1^{ws} &= s_1\dot{s}_1 + \frac{1}{\eta}\tilde{\boldsymbol{W}}_1^T \dot{\hat{\boldsymbol{W}}}_1 \\
&= s_1 b_2\left(-\lambda_1 s_1 - \delta_1 s_1^{q_1/r_1} + \tilde{\boldsymbol{W}}_i \boldsymbol{h}_i(\boldsymbol{x}_i) - \varepsilon_1\right) + \frac{1}{\eta}\tilde{\boldsymbol{W}}_1^T \dot{\hat{\boldsymbol{W}}}_1 \\
&= -b_2\lambda_1 s_1^2 - b_2\delta_1 s_1^{q_1/r_1+1} + s_1 b_2\tilde{\boldsymbol{W}}_1^T \boldsymbol{h}_i(\boldsymbol{x}_i) - s_1 b_2\varepsilon_1 + \frac{1}{\eta}\tilde{\boldsymbol{W}}_1^T \dot{\hat{\boldsymbol{W}}}_1 \\
&= -b_2\lambda_1 s_1^2 - b_2 s_1^{q_1/r_i+1}\left(\delta_1 + \frac{\varepsilon_1}{s_1^{q_1/r_1}}\right) + \tilde{\boldsymbol{W}}_1^T\left(s_1 b_2\boldsymbol{h}_i(\boldsymbol{x}_i) + \frac{1}{\eta}\dot{\hat{\boldsymbol{W}}}_1\right) \\
&= -b_2\lambda_1 s_1^2 - b_2 s_1^{q_1/r_1+1}\left(\delta_1 + \frac{\varepsilon_1}{s_1^{q_1/r_1}}\right)
\end{aligned}
$$

If $\delta_1 > \left|\varepsilon_1/s_1^{q_1/r_1}\right|$, then $\dot{V}_1^{ws} \leq 0$, which means s_1 and $\tilde{\boldsymbol{W}}_1$ converge to zero.

Define a Lyapunov candidate function

$$V_i^e = \frac{1}{2}e_i^2 \tag{2.18}$$

Submit $s_i = 0$ and Eqs (2.9)–(2.18), the derivative of Eq (2.18) are obtained.

$$\dot{V}_i^e = e_i\dot{e}_i = -a_i e_i^2 - \beta_i e_i^{q_i/r_i+1} \leq 0 \tag{2.19}$$

According to Eq (2.19), e_i converges to zero too. Equation (2.16) can be rewritten as (take $i = 1$ for example):

$$\dot{s}_1 = -\lambda_1' s_1 - \delta_1' s_1^{q_1}/r_1 \tag{2.20}$$

where, $\lambda_1' = \lambda_1 b_2$, $\delta_1' = \left(b_2\delta_1 + \frac{b_2\varepsilon_1}{s_1^{q_1/r_1}}\right)$.

By solving Eq (2.20), the convergence time of s_i from $s_1(0) \neq 0$ to $s_1(t_s) = 0$ is obtained:

$$
\begin{aligned}
t_s &= \frac{r_1}{\lambda_1'(r_1-q_1)} \ln \frac{\lambda_1' s_1(0)^{(r_1-q_1)/q_1}+\delta_1'}{\delta_1'} \\
&\leq \frac{r_1}{\lambda_1'(r_1-q_1)} \ln \frac{\lambda_1' s_1(0)^{(r_1-q_1)/q_1}+b_2\delta_1}{b_2\delta_1}
\end{aligned}
$$

□

Table 1. The main physical parameters of the quadrotor used in the simulation.

Parameter	Value	Parameter	Value
$g(m/s^2)$	9.81	$I_z(kg \cdot m^2)$	0.0013
$m(kg)$	0.468	$I_r(kg \cdot m^2)$	0.0028
$I_x(kg \cdot m^2)$	0.0075	$l(m)$	0.25
$I_y(kg \cdot m^2)$	0.0075	$\Omega_r(rad/s)$	1

3. Results

The simulation results given in this section verify the effectiveness and performance of the proposed controller. Table 1 lists the main physical parameters of the quadrotor used in the simulation. The proposed flight controllers parameters are set as: $\alpha_1 = 4.5, \beta_1 = 1.4, q_1 = 3, r_1 = 7, \sigma_1 = 0.2, \lambda_1 = 10, \alpha_2 = 8.2, \beta_2 = 2.5, q_2 = 3.5, r_2 = 7.5, \sigma_2 = 0.2, \lambda_2 = 20, \eta_1 = 10.5, \eta_2 = 14.5, n = 5, c = [-2.6, -1.2, 0, -1.2, -2.6]$

In the simulation flight, the quadrotor tracked 3d trajectory under the external interference and parameter uncertainties. The initial state values of the quadrotor are [0, 0, 0]rad and [0, 0, 0]m. The disturbance terms are set as $d_1 = d_2 = d_3 = 0.2sin(4t), d_4 = d_5 = d_6 = 0.06cos(4t)$. Seven seconds after the simulation started, the quadrotor weight was suddenly reduced by 30%. Comparative simulations with the conventional GFTSM control method proposed in [19] are also given.

The simulation results of the proposed controller are shown in Figures 5–10. As shown in Figure 5, the 3D flight trajectory demonstrated that the proposed controller has succeeded following the 3D flight trajectory in finite time, but the traditional GFTSM has steady-state error since the parameter perturbation. Figure 6 shows the time evolution of position variables (x, y and z). It can be seen that the abrupt change of z variable due to the mass variation was overcome by the GFTSM-RBF controller within 2 seconds, but GFTSM failed. Figure 7 shows the tracking errors of position variables. It can be observed that the proposed GFTSM-RBF achieves better position tracking than traditional GFTSM. Figures 8 and 9 shows the trajectory tracking of attitude angles(ϕ, θ and ψ). It can be observed that both of the two control systems can track the attitude references accurately, but the GFTSM-RBF controller has a faster tracking speed and smoother yaw angle ψ. Figure 10 presents the control inputs of the two control approaches. As expected, GFTSM exhibits chattering in the control input, and the GFTSM-RBF nearly eliminates the chattering. As a result, the GFTSM-RBF method presents a faster tracking speed and greater robustness against sustained time-varying disturbances and parametric uncertainties.
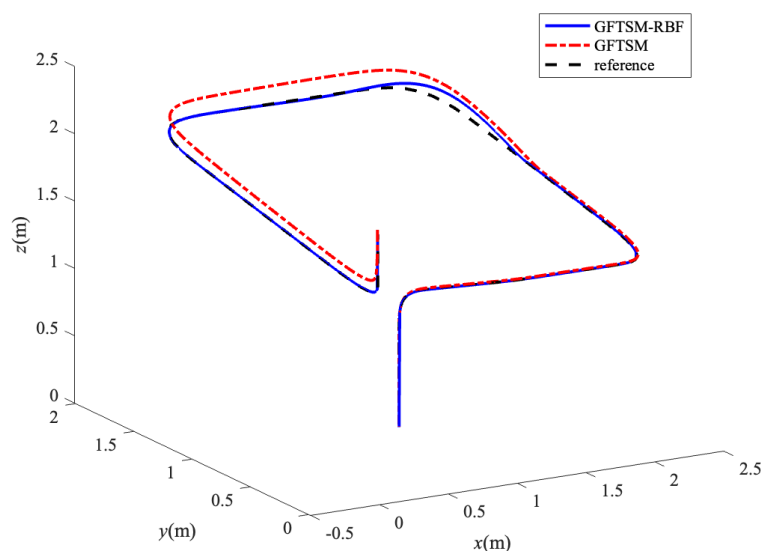

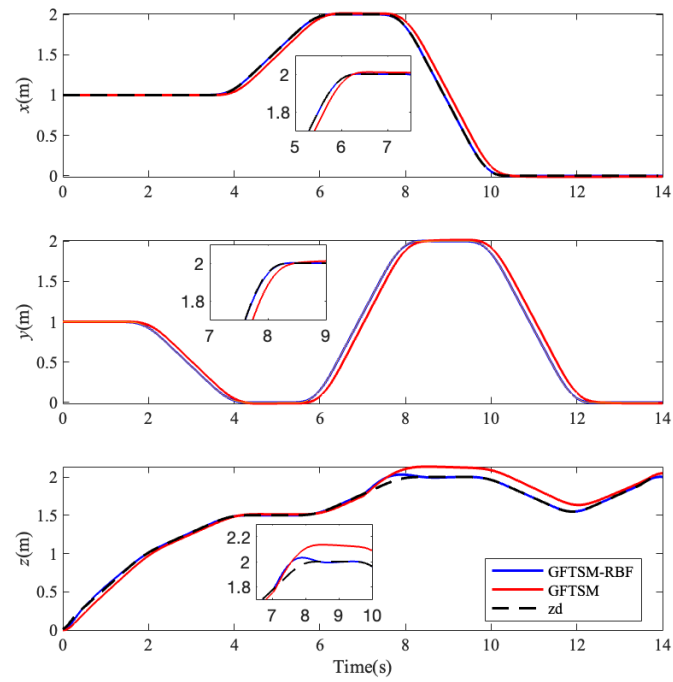**Figure 5.** The 3D trajectory tracking of the quadrotor.

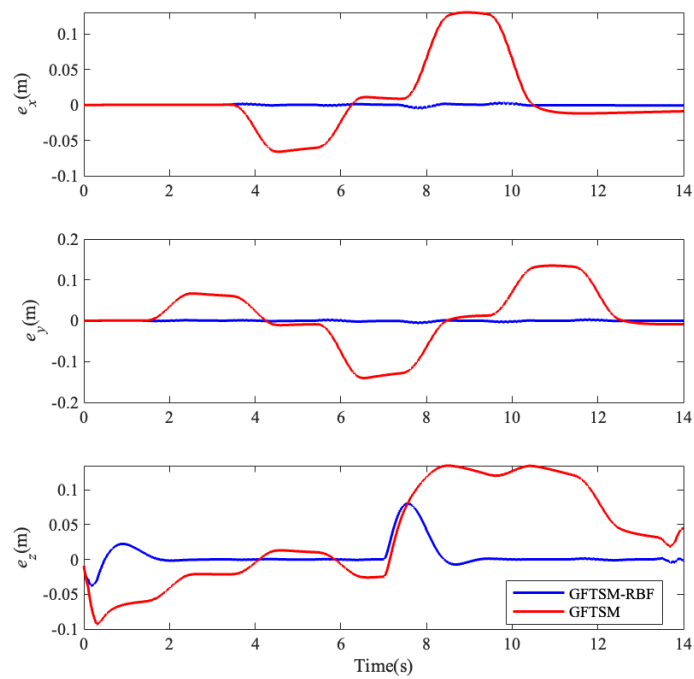**Figure 6.** The position components of the two obtained trajectories.

Figure 7. The position tracking errors of the two control method.
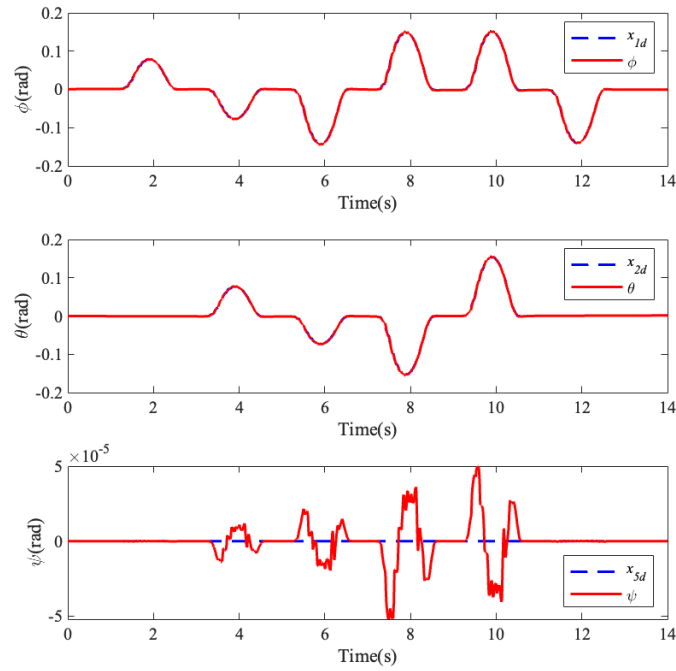
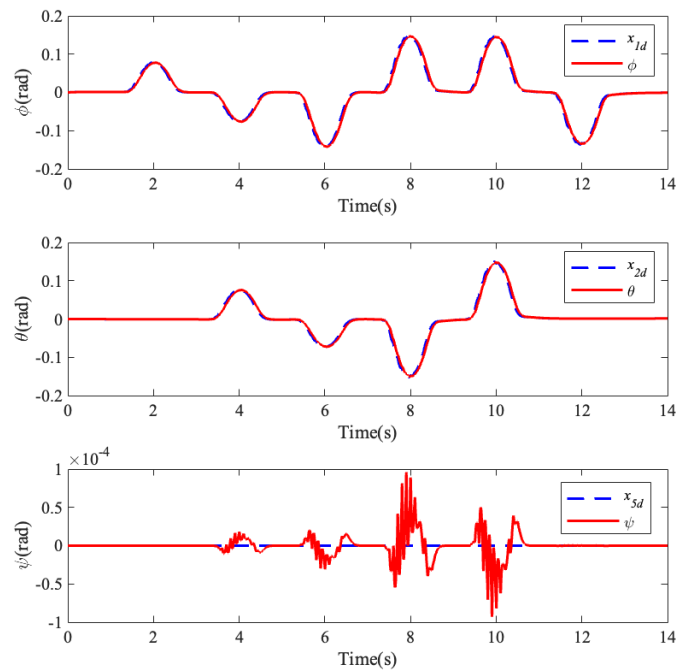**Figure 8.** The trajectory tracking of attitude angle with GFTSM-RBF.

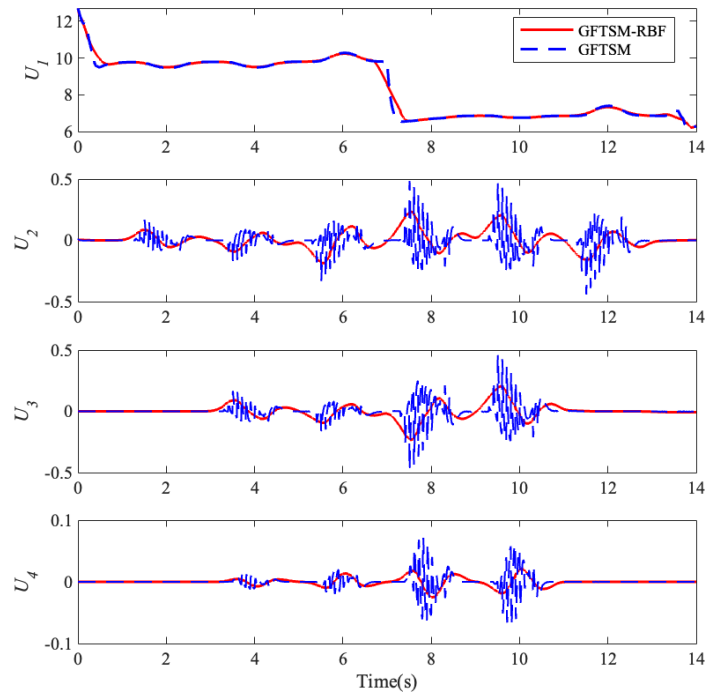**Figure 9.** The trajectory tracking of attitude angle with GFTSM.

Figure 10. The control input of quadrotor.

4. Conclusions

This paper presents an adaptive GFTSM-NN controller to realize 3D trajectory tracking for the quadrotor with unknown disturbance and dynamic uncertainty. Global fast terminal sliding surfaces are designed for finite-time convergence of all the outputs of quadrotor. The equivalent control of the GFTSM controller is estimated by the RBF-NN. Adaptive laws are developed to compute the weights of RBF-NN. The quadrotor's closed-loop stability and finite-time convergence is guaranteed through the Lyapunov theory and subsequent analysis demonstrated in this paper. Finally, a comparison of the proposed control technique is presented with the conventional GFTSM. The results demonstrate that the GFTSM-NN achieves faster response speed, more robust to dynamic uncertainty, and lower chattering than the GFTSM. In future work, the proposed GFTSM-RBF approach will be validated by a real quadrotor UAV to perform the trajectory-tracking task. Also, control input constraints will be considered.

Acknowledgments

Conflict of interest

All authors disclosed no relevant relationships.

References

1. M. assanalian, A. Abdelkefi, Classifications, applications, and design challenges of drones: A review, *Prog. Aerosp. Sci.*, **91** (2017), 99–131. https://doi.org/10.1016/j.paerosci.2017.04.003

2. M. Labbadi, M. Cherkaoui, Adaptive fractional-order nonsingular fast terminal sliding mode based robust tracking control of quadrotor UAV with Gaussian random disturbances and uncertainties, *IEEE Trans. Aerosp. Electron. Syst.*, **57** (2021), 2265–2277. https://doi.org/10.1109/TAES.2021.3053109

3. A. Noordin, M. A. Mohd Basri, Z. Mohamed, Adaptive PID controller using sliding mode control approaches for quadrotor UAV attitude and position stabilization, *Arabian J. Sci. Eng.*, **46** (2021), 963–981. https://doi.org/10.1007/s13369-020-04742-w

4. F. W. Alsaade, H. Jahanshahi, Q. Yao, M. S. Al-zahranidAli, S. Alzahranie, A new neural network-based optimal mixed H2/H∞ control for a modified unmanned aerial vehicle subject to control input constraints, *Adv. Space Res.*, (2022), forthcoming. https://doi.org/10.1016/j.asr.2022.02.012

5. L. Wan, Y. Su, H. Zhang, Y. Tang, Global fast terminal sliding mode control based on radial basis function neural network for course keeping of unmanned surface vehicle, *Int. J. Adv. Rob. Syst.*, **16** (2019), 1729881419829961. https://doi.org/10.1177/1729881419829961

6. J. Zhang, D. Gu, C. Deng, B. Wen, Robust and adaptive backstepping control for hexacopter UAVs, *IEEE Access*, **7** (2019), 163502–163514. https://doi.org/10.1109/ACCESS.2019.2951282

7. X. Zhang, Y. Wang, G. Zhu, X. Chen, Z. Li, C. Wang, et al., Compound adaptive fuzzy quantized control for quadrotor and its experimental verification, *IEEE Trans. Cybern.*, **51** (2020), 1121–1133. https://doi.org/10.1109/TCYB.2020.2987811

8. Y. Zhang, Z. Chen, X. Zhang, Q. Sun, M. Sun, A novel control scheme for quadrotor UAV based upon active disturbance rejection control, *Aerosp. Sci. Technol.*, **79** (2018), 601–609. https://doi.org/10.1016/j.ast.2018.06.017

9. M. Labbadi, Y. Boukal, M. Cherkaoui, Control of the QUAV by a hybrid finite-time tracking technique, in *Advanced Robust Nonlinear Control Approaches for Quadrotor Unmanned Aerial Vehicle*, (2022), 81–101. https://doi.org/10.1007/978-3-030-81014-6_4

10. N. Wang, Q. Deng, G. Xie, X. Pan, Hybrid finite-time trajectory tracking control of a quadrotor, *ISA Trans.*, **90** (2019), 278–286. https://doi.org/10.1016/j.isatra.2018.12.042

11. Q. Yao, H. Jahanshahi, Novel finite-time adaptive sliding mode tracking control for disturbed mechanical systems, in *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, (2022), 09544062221091530. https://doi.org/10.1177/09544062221091530

12. Q. Xu, Z. Wang, Z. Zhen, Adaptive neural network finite time control for quadrotor UAV with unknown input saturation, *Nonlinear Dyn.*, **98** (2019), 1973–1998. https://doi.org/10.1007/s11071-019-05301-1

13. H. Chen, H. Chen, P. Xu, Global fast terminal sliding mode control law design of a quadrotor, in *2019 international conference on computer, network, communication and information systems (CNCI 2019), Atlantis Press*, (2019).

14. H. Hassani, A. Mansouri, A. Ahaitouf, Robust autonomous flight for quadrotor UAV based on adaptive nonsingular fast terminal sliding mode control, *Int. J. Dyn. Control*, **9** (2021), 619–635. https://doi.org/10.1007/s40435-020-00666-3

15. M. Labbadi, M. Cherkaoui, Robust adaptive backstepping fast terminal sliding mode controller for uncertain quadrotor UAV, *Aerosp. Sci. Technol.*, **93** (2019), 105306. https://doi.org/10.1016/j.ast.2019.105306

16. I. Eker, Second-order sliding mode control with experimental application, *ISA Trans.*, **49** (2010), 394–405. https://doi.org/10.1016/j.isatra.2010.03.010

17. I. Al-Darraji, D. Piromalis, A. A. Kakei, F. Q. Khan, M. Stojmenovic, G. Tsaramirsis, Adaptive robust controller design-based RBF neural network for aerial robot arm model, *Electronics*, **10** (2021), 831. https://doi.org/10.3390/electronics10070831

18. T. N. Truong, A. T. Vo, H. J. Kang, A backstepping global fast terminal sliding mode control for trajectory tracking control of industrial robotic manipulators, *IEEE Access*, **9** (2021), 31921–31931. https://doi.org/10.1109/ACCESS.2021.3060115

19. J. J. Xiong, G. B. Zhang, Global fast dynamic terminal sliding mode control for a quadrotor UAV, *ISA Trans.*, **66** (2017), 233–240. https://doi.org/10.1016/j.isatra.2016.09.019