Finite Time Back Stepping Supper Twisting Controller Design for a Quadrotor

N. Seyed Gogani[1], V. Behnamgol[2], S. M. Hakimi[3], G. Derakhshan[4]

Abstract— **The purpose of this paper is to control a quadrotor in a finite-time. Here the nonlinear controller is designed by the back-stepping method. For each quadrotor subsystem, a proposed finite-time nonlinear controller is designed in several steps. In addition, in the last step of the controller design algorithm, the super-twisting method is used, which is robust to uncertainties. This method produces a smooth control signal because it is a second-order sliding mode algorithm. This control system ensures finite-time convergence of quadrotor attitude and position states. The stability of the proposed method has been analytically proven and computer simulations have been performed to evaluate the results of the control system.**

Index Terms— **Quadrotor, Finite-time convergence, Back-stepping method, Super-twisting controller**

I. Introduction

UNMANNED aerial vehicles have received significant attention in recent decades. Especially, the quadrotor, as an example of small scale UAVs, is the focus of much attention in the research community, owing to its simple structure and low cost. It has been successful in many commercial applications, such as aerial photography, agricultural service, search and rescue, industrial inspection and so on [1]. Given advantages (managed remotely or autonomously, fly over areas of difficult access, etc.), these robots are ideal tools for inspection [2].

Control of these flying robots has been done using various methods including linear controllers. For example in [3] the classical PID controller is designed to control a quadrotor. The design of the LQR controller to control a quadrotor is performed in [4]. One of the disadvantages of linear controllers is that they do not have the ability to effectively stabilize quadrotor when complex airflow*s* are generated by the reaction of the rotors and also they cannot stabilize it in the presence of disturbances. Because these items cause the quadrotor to deviate from the operating point for which the controller is designed.

The dynamic quadrotor model is nonlinear [5], so it is better controlled with nonlinear control methods. So far, many nonlinear methods have been used to control it. The back stepping control method is very useful when some states are controlled by other states. In quadrotor, it is usually advantageous to use the back stepping method for transferring positional movements. In [6-8] the back stepping controller is designed to control quadrotors. The sliding mode control (SMC) method is a nonlinear method for uncertain systems. This control method has also been used for quadrotor in [9-11]. Different types of this control method have also been used for various applications, including dynamic terminal SMC [9], integral back stepping SMC [10], Adaptive SMC [11] and chaos synchronization via integral SMC [12].

The convergence of the output to the desired value in a finite time is very important. Finite time stability, in addition to the error convergence speed, also brings robustness to uncertainties and disturbances [13]. In some methods, such as the sliding mode method, to ensure finite time convergence, the control input equation includes a discontinuous sign function that causes high frequency oscillation in the control signal. Due to the oscillation control signal, it is not possible for this type of controllers to perform. A very popular control algorithm is the super-twisting (ST) [14] algorithm. It was proposed to stabilize disturbed systems of relative degree one by means of continuous control. The ST enforces second-order sliding modes, i.e. the output (sliding variable) and its time-derivative*s* are driven to zero in finite-time. Theoretically, the STA is able to compensate Lipschitz disturbances exactly [15]. Applications of ST algorithm in various systems are presented in [16-19].

The main innovation of this paper is to control a quadrotor robot in a finite time. This goal will be achieved by proposing a hybrid finite time backstepping algorithm. The hybrid controller must be used for four different quadrotor subsystems separately. Due to the nonlinearity of the proposed controller, it will be used in the whole quadrotor operating range. In the first stages of controller design, the finite time stability theory is used to obtain virtual controllers. The super twisting algorithm is used in final step of each four controllers which causes robustness against model and parametric uncertainties.

In the second section, modeling will be done. In the third

Manuscript received Oct 28, 2021; revised Apr 04, 2022.

Nazanin Seyed Gogani is a Ph.D. Candidate of the Department of Electrical Engineering and Renewable Energy Research Centre, Damavand Branch, Islamic Azad University, Damavand, Iran. (e-mail: seyedgogani.nazanin@gmail.com)

Vahid Behnamgol is an assistant professor of Malek Ashtar University of Technology, Tehran, Iran. (corresponding author; e-mail: vahid_behnamgol@mut.ac.ir).

Seyed Mahdi Hakimi is an assistant Professor of Department of Electrical Engineering and Renewable Energy Research Centre, Damavand Branch, Islamic Azad University, Damavand, Iran. (sm_hakimi83@yahoo.com).

Ghasem Derakhshan is an assistant professor of Department of Electrical Engineering and Renewable Energy Research Centre, Damavand Branch, Islamic Azad University, Damavand, Iran (e-mail: gh.derakhshan@yahoo.com).

part, the proposed control theory will be described and the proposed method will be used to control this flying object. In the fourth section, the simulation results will be presented. Finally, the conclusion will be made.

II. Quadrotor Dynamic Model

Several basic assumptions are made for quadrotor modeling. These assumptions simplify the modeling process and provide a simpler dynamic model. These assumptions are as follows [20]:

- The structure is supposed to be rigid. This assumption limits the degree of freedom of the quadrotor to 6.
- The structure is supposed to be symmetrical. Using this assumption causes the inertia matrix of the quadrotor body to be symmetrical.
- The center of mass and the body fixed frame origin are assumed to coincide.
- The propellers are supposed to be rigid.
- The propelling force of the rotors is proportional to the square of the angular speed of the rotors.

Using Newton's second law, the relationship between the equations of motion, forces and torques is expressed. Therefore, the equations of motion must be transferred from the body coordinate system to the inertia coordinate system. Using Newton's second law, the dynamic equations are as follows:

$$(1)\quad \begin{cases} \ddot{X} = -g.\begin{pmatrix}0\\0\\1\end{pmatrix} + R.\dfrac{b}{m}\sum_{i=1}^{4}\omega_i^2.\begin{pmatrix}0\\0\\1\end{pmatrix} \\ I\ddot{\Theta} = -\dot{\Theta}\times I\dot{\Theta} - \sum_{i=1}^{4} J_R\left(\dot{\Theta}\times\begin{pmatrix}0\\0\\1\end{pmatrix}\right)\omega_i + \tau \end{cases}$$

In this equation, I is the body inertia matrix of quadrotor, which is a diametric matrix due to the symmetry of the quadrotor [21].

$$(2)\quad I = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}$$

J_R is the inertia of the rotor and τ is the angular torque vector on the quadrotor.

$$(3)\quad \tau = \begin{pmatrix} lb(\omega_4^2-\omega_2^2) \\ lb(\omega_3^2-\omega_1^2) \\ ld(\omega_2^2+\omega_4^2-\omega_1^2-\omega_3^2) \end{pmatrix}$$

d is the coefficient of air resistance and l is the length of the connecting rod between the rotor and the center of the quadrotor. Using the angular speed of the rotors, four control signals are defined to simplify the equations of the quadrotor dynamic model in the form of Equation (4).

$$(4)\quad \begin{cases} u_1 = b(\omega_1^2+\omega_2^2+\omega_3^2+\omega_4^2) \\ u_2 = b(\omega_4^2-\omega_2^2) \\ u_3 = b(\omega_3^2-\omega_1^2) \\ u_4 = d(\omega_2^2+\omega_4^2-\omega_1^2-\omega_3^2) \end{cases}$$

$$(5)\quad \omega_d = \omega_2+\omega_4-\omega_1-\omega_3$$

The following dynamic are obtained using differential equations for angular accelerations and linear acceleration.

$$(6)\quad \begin{cases} \ddot{x} = (\cos\phi\sin\theta\cos\psi+\sin\phi\sin\psi)u_1/m \\ \ddot{y} = (\cos\phi\sin\theta\sin\psi-\sin\phi\cos\psi)u_1/m \\ \ddot{z} = -g+(\cos\phi\,\cos\theta)u_1/m \\ \ddot{\phi} = \dot{\psi}\dot{\theta}\left(\dfrac{I_y-I_z}{I_x}\right) - \dfrac{J_R}{I_x}\dot{\theta}\omega_d + \dfrac{l}{I_y}u_2 \\ \ddot{\theta} = \dot{\phi}\dot{\psi}\left(\dfrac{I_z-I_x}{I_y}\right) + \dfrac{J_R}{I_y}\dot{\phi}\omega_d + \dfrac{l}{I_y}u_3 \\ \ddot{\psi} = \dot{\phi}\dot{\theta}\left(\dfrac{I_x-I_y}{I_z}\right) + \dfrac{l}{I_z}u_4 \end{cases}$$

It can be seen that the angles subsystem is completely independent of the position subsystem. But the position subsystem depends on the angles subsystem. This model can be written in the form of $\dot{X} = f(X,U)$ state space where $U^T = [u_1,u_2,u_3,u_4]$ is the vector of input. $X \in \Re^{12}$ is the vector of variables and is defined as follows [21]:

$$(7)\quad X^T = (x,\dot{x},y,\dot{y},z,\dot{z},\phi,\dot{\phi},\theta,\dot{\theta},\psi,\dot{\psi})$$

System state space:

$$(8)\quad \begin{aligned} \dot{X} &= F(x)+G(x)U \\ Y &= CX \end{aligned}$$

III. Control System Design

A. Hybrid back-stepping super-twisting control with finite-time convergence

Many studies have been conducted on the finite-time convergence of systems [22, 23]. Consider the following system:

$$(9)\quad \dot{x} = f(x), \qquad f: D \to \Re^n$$

Consider the continuous Lyapunov function $V:\mathbb{R}^n \to \mathbb{R}$ in the neighborhood of the equilibrium point, which is positive. So the origin will be finite time stable if:

$$(10)\quad \dot{V}(x) \le -cV^{\alpha}(x),\ \ 0<\alpha<1,\ \ c>0$$

and the convergence time depends on the initial conditions as follows [24, 25]:

$$(11)\quad t \le \frac{V(x(0))^{1-\alpha}}{c(1-\alpha)}$$

To describe a backstepping controller with finite time convergence capability, consider the following second order system:

$$(12)\quad \begin{cases} \dot{x} = f(x)+g(x)\xi \\ \dot{\xi} = u \end{cases}$$

we start with the first equation (12):

$$(13)\quad \dot{x} = f(x)+g(x)\xi$$

ξ is considered as a virtual input. Then the virtual control is as follows:

$$(14)\quad \xi = \phi(x)$$

to bring the state variable x to the desired value in a finite time. For this purpose, consider the following Lyapunov function:

(15) $$V(x)=\frac{1}{2}\left(x-x_d\right)^2$$

x_d is the desirable value of the x . The derivative of this function is as follows:

(16) $$\begin{aligned}\dot{V}(x)&=\frac{\partial V}{\partial x}\dot{x}\\&=\left(x-x_d\right)\left[f(x)+g(x)\phi(x)-\dot{x}_d\right]\end{aligned}$$

using

(17) $$\phi(x)=\frac{1}{g(x)}\left[-f(x)+\dot{x}_d-\frac{1}{\left(x-x_d\right)}\left(cV^{\alpha}\right)\right]$$

we have

(18) $$\dot{V}(x)\leq -cV^{\alpha}$$

Equation (18) guarantees that the x converges to x_d in a finite time and its convergence time will be obtained from:

(19) $$t_x\leq\frac{V\left(x(0)-x_d(0)\right)^{1-\alpha}}{c\left(1-\alpha\right)}$$

Now in the second step, a sliding variable is defined as follows:

(20) $$\begin{aligned}s&=\xi-\phi(x)\\\dot{s}&=\dot{\xi}-\dot{\phi}(x)=\mathrm{u}-\dot{\phi}(x)\end{aligned}$$

The sum of parametric and model uncertainties can be added to the sliding variable dynamics as follows:

(21) $$\dot{s}=u-\dot{\phi}(x)+d(t),\quad \left|d(t)\right|\leq L_d$$

control input to stabilize the sliding variable, by using super twisting algorithm is determined as [24-29]:

(22) $$\begin{aligned}u&=\dot{\phi}(x)-k_1\left|s\right|^{\rho}sign\left(s\right)+\xi\\\dot{\xi}&=-k_2 sign\left(s\right)\end{aligned}$$

(23) $$\begin{cases}k_1>\sqrt{\dfrac{4L_d\left(k_2+L_d\right)}{k_2-L_d}}\\k_2>L_d\end{cases}$$

B. Position subsystem controller

To control the position of the quadrotor on the z axis, the subsystem related to this section is as follows:

(24) $$\begin{aligned}\dot{x}_5&=x_6\\\dot{x}_6&=-g+\frac{\cos x_7\cos x_9}{m}u_1\\y_3&=x_5\end{aligned}$$

where $x_5=z$ and since the dynamic system has two state variable, so the back stepping controller will be designed in two steps. To control the z, we start with equation (25):

(25) $$\dot{x}_5=x_6$$

In this equation, x_6 is considered as a virtual input. Then this virtual control will be considered as follows:

(26) $$x_6=\phi_1(x)$$

this virtual control input brings the state variable x_5 to the desired value in a finite time. For this purpose, consider the Lyapunov function:

(27) $$V_{11}(x)=\frac{1}{2}\left(x_5-x_{5d}\right)^2$$

where x_{5d} is the desirable value of the state variable x_5 or the desirable position on the z axis. The derivative of this Lyapunov function is as follows:

(28) $$\dot{V}_{11}(x)=\frac{\partial V_{11}}{\partial x}\dot{x}_5=\left(\dot{x}_5-\dot{x}_{5d}\right)\left[\phi_1(x)-\dot{x}_{5d}\right]$$

(29) $$\phi_1(x)=\dot{x}_{5d}-\frac{1}{\left(x_5-x_{5d}\right)}\left(c_{11}V_{11}^{\alpha_{11}}\right)$$

(30) $$\dot{V}_{11}(x)\leq -c_{11}V_{11}^{\alpha_{11}}$$

Equation (30) guarantees that the state variable x_5 will converge to x_{5d} in a finite time and its convergence time will be obtained from the following equation:

(31) $$t_{x_5}\leq\frac{V_{11}\left(0\right)^{1-\alpha_{11}}}{c_{11}\left(1-\alpha_{11}\right)}$$

Now in the second step, a sliding variable is defined as follows:

(32) $$s_1=x_6-\phi_1(x)$$

(33) $$\begin{aligned}\dot{s}_1&=\dot{x}_6-\dot{\phi}_1(x)\\&=-g+\frac{\cos x_7\cos x_9}{m}u_1-\dot{\phi}_1(x)+d_1(t)\end{aligned}$$

where $d_1(t)$ is sum of parametric and model uncertainties. Control input by using super twisting algorithm is determined as:

(34) $$\begin{aligned}u_1&=\frac{m}{\cos x_7\cos x_9}[g+\dot{\phi}_1(x)\\&\quad-k_{11}\left|s_1\right|^{\rho_1}sign\left(s_1\right)+\xi_1]\\\dot{\xi}_1&=-k_{12}sign\left(s_1\right)\end{aligned}$$

By using this controller, the finite time stability of the system will be guaranteed in the second step.

The x axis is controlled by θ angle. Therefore, we use the following dynamics to control the position on this axis:

(35) $$\begin{aligned}\dot{x}_1&=x_2\\\dot{x}_2&=\frac{(\cos x_7\sin x_9\cos x_{11}+\sin x_7\sin x_{11})u_1}{m}\\\dot{x}_9&=x_{10}\\\dot{x}_{10}&=x_{12}x_8I_2+\frac{J_R}{I_y}x_8w_d+\frac{l}{I_y}u_3\\y_1&=x_1\end{aligned}$$

In which $x_1=x$ and $x_9=\theta$. The above dynamic system has four state variables, so the finite time back stepping controller will be designed in four steps. With the same process as in the height axis, the controller in this axis is designed as follows:

$$
\begin{aligned}
&u_3 = I_y[-x_{12}x_8 I_2 - \frac{J_R}{I_y}x_8 w_d \\
&\quad + \dot{\phi}_{33}(x) - k_{31}|s_3|^{\rho_3} sign(s_3) + \xi_3] \\
&\dot{\xi}_3 = -k_{32} sign(s_3) \\
&V_{31}(x) = \frac{1}{2}(x_1 - x_{1d})^2 \\
&\phi_3(x) = \dot{x}_{1d} - \frac{1}{(x_1 - x_{1d})}(c_{31}V_{31}{}^{\alpha_{31}}) \\
&z_3 = x_2 - \phi_{31}(x) \\
&\phi_{32}(x) = \sin^{-1}(\frac{m}{\cos x_7 \cos x_{11}} \\
&\qquad (\frac{-\sin x_7 \sin x_{11} u_1}{m} + \dot{\phi}_{31}(x) \\
&\qquad - \frac{1}{z_3}(c_{32}V_{31}{}^{\alpha_{32}}))) \\
&V_{32} = \frac{1}{2}z_{31}{}^2 \\
&z_{32} = x_9 - \phi_{32}(x) \\
&V_{33}(x) = \frac{1}{2}(z_{32})^2 = \frac{1}{2}(x_9 - \phi_{32}(x))^2 \\
&\phi_{33}(x) = \dot{\phi}_{32}(x) - \frac{1}{(x_9 - \phi_{32}(x))}(c_{33}V_{33}{}^{\alpha_{33}}) \\
&s_3 = x_{10} - \phi_{33}(x)
\end{aligned}
\tag{36}
$$

The control of the y axis is done by the ϕ angle. Therefore, to control the position on this axis, we consider the following subsystem:

$$
\begin{aligned}
&\dot{x}_3 = x_4 \\
&\dot{x}_4 = \frac{(\cos x_7 \sin x_9 \sin x_{11} - \sin x_7 \cos x_{11})u_1}{m} \\
&\dot{x}_7 = x_8 \\
&\dot{x}_8 = x_{12}x_{10}I_1 - \frac{J_R}{I_x}x_{10}w_d + \frac{l}{I_x}u_2 \\
&y_2 = x_3
\end{aligned}
\tag{37}
$$

In which $x_3 = y$, $x_{11} = \phi$. The above dynamic system has four state variables, so the back stepping controller will be designed in four steps. Similar to the x axis controller, the controller in this axis is also designed as follows:

$$
\begin{aligned}
&u_2 = I_x[-x_{12}x_{10}I_1 + \frac{J_R}{I_x}x_{10}w_d \\
&\quad + \dot{\phi}_{23}(x) - k_{21}|s_2|^{\rho_2} sign(s_2) + \xi_2] \\
&\dot{\xi}_2 = -k_{22} sign(s_2) \\
&V_{21}(x) = \frac{1}{2}(x_3 - x_{3d})^2 \\
&\phi_{21}(x) = \dot{x}_{3d} - \frac{1}{(x_3 - x_{3d})}(c_{21}V_{21}{}^{\alpha_{21}}) \\
&z_{21} = x_4 - \phi_{21}(x) \\
&V_{22} = \frac{1}{2}z_{21}{}^2 \\
&\phi_{22}(x) = \cos^{-1}(\frac{m}{\sin x_9 \sin x_{11}} \\
&\qquad (\frac{\sin x_7 \cos x_{11} u_1}{m} + \dot{\phi}_{21}(x) \\
&\qquad - \frac{1}{z_{21}}(c_{22}V_{22}{}^{\alpha_{22}}))) \\
&z_{22} = x_7 - \phi_{22}(x) \\
&V_{23}(x) = \frac{1}{2}(z_{22})^2 = \frac{1}{2}(x_7 - \phi_{22}(x))^2 \\
&\phi_{23}(x) = \dot{\phi}_2(x) - \frac{1}{(x_7 - \phi_{22}(x))}(c_{23}V_{23}{}^{\alpha_{23}}) \\
&s_2 = x_8 - \phi_{23}(x)
\end{aligned}
\tag{38}
$$

C. Control of ψ angle

To control ψ angle, consider the following dynamics:

$$
\begin{aligned}
&\dot{x}_{11} = x_{12} \\
&\dot{x}_{12} = x_{10}x_8 I_3 + \frac{l}{I_z}u_4 \\
&y_4 = x_{11}
\end{aligned}
\tag{39}
$$

where $x_{11} = \psi$ and the above dynamic system has two state variables, so the controller will be designed in two steps. The controller is designed similarly to the previous section as follows:

$$
\begin{aligned}
&u_4 = I_z[-x_{10}x_8 I_3 + \dot{\phi}_4(x) \\
&\quad - k_{41}|s_4|^{\rho_4} sign(s_4) + \xi_4] \\
&\dot{\xi}_4 = -k_{42} sign(s_4) \\
&V_4(x) = \frac{1}{2}(x_{11} - x_{11d})^2 \\
&\phi_4(x) = \dot{x}_{11d} - \frac{1}{(x_{11} - x_{11d})}(c_4 V_4{}^{\alpha_4}) \\
&s_4 = x_{12} - \phi_{14}(x)
\end{aligned}
\tag{40}
$$

IV. Simulation Results

Here the results of the proposed controller simulation will be presented. To perform the simulation, the values of the parameters are used as identified in the following table:

TABLE. 1- QUADROTOR MODEL PARAMETERS

Units	Values	Variable
kg	1.1	m_s
m	0.21	l
Ns2/rad	1.22	$l_x = l_y$
Ns2/rad	2.2	l_z
Ns2/rad	0.2	l_r
Ns/m	0.1	$k_i\,(i=1,2,3)$
Ns/m	0.12	$k_i\,(i=4,5,6)$
m/s2	9.81	g
Ns2	5	b
N/ms2	2	k
-	1	c

To check the performance of the proposed controller in first scenario, tracking a specific trajectory is considered. In Figures (1) and (2), by applying the controller to the quadrotor system, changes in position variables and errors in tracking the desired positions are shown. As can be seen, by applying the proposed control system, the position states follow the desired positions in a finite time, and the position tracking error converges to zero with high precision.

Figures (3) and (4) show the changes of angles and the angles tracking errors. It is evident that, by applying the proposed control system, the quadrotor angles follow the desired angles with high accuracy and in a finite time, also the angles tracking error converges to zero. It can be say that only state ψ is the main output of the system and states θ and ϕ are used as auxiliary variables to track the positions x and y.

Figure (5) shows the control input signals. It can be seen that the control signals generated using the proposed controllers have a suitable amplitude and smooth behavior that can be easily implemented. Figure (6) shows the trajectory of the quadrotor and the desired trajectory. It is evident that the quadrotor have followed the desired trajectory with good accuracy.

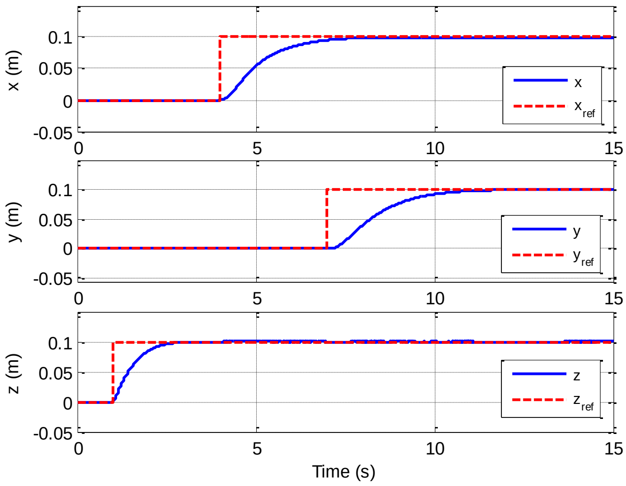


Fig. 1- position variables in first scenario

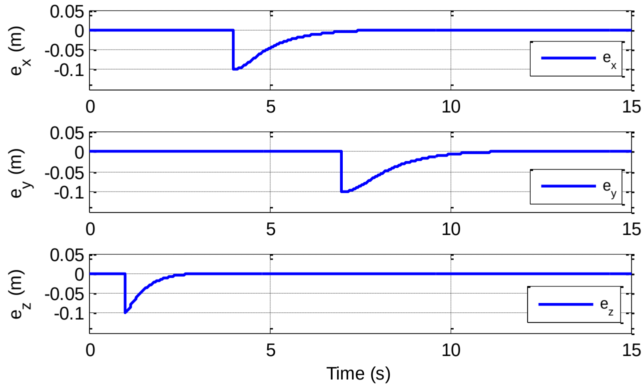


Fig. 2- Position tracking errors in first scenario

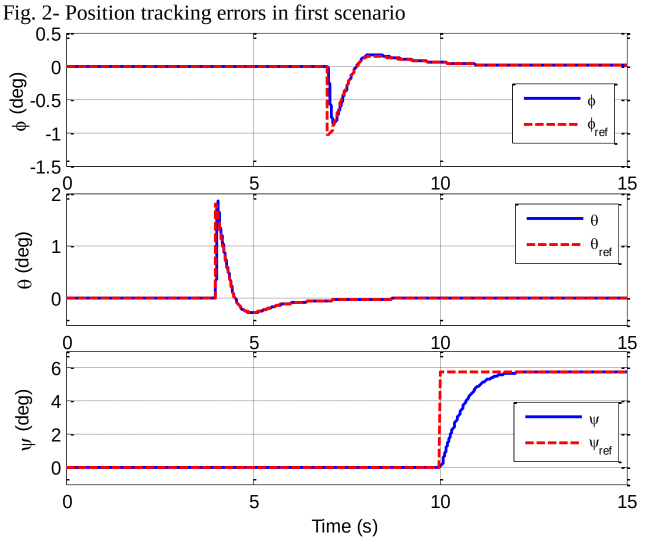
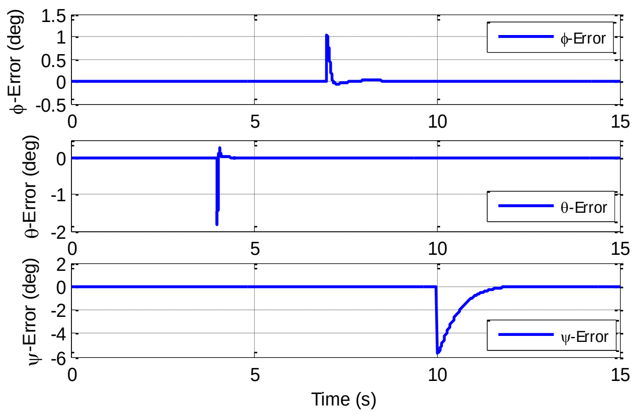


Fig. 3- Angle variables in first scenario



4- Angles tracking error in first scenario

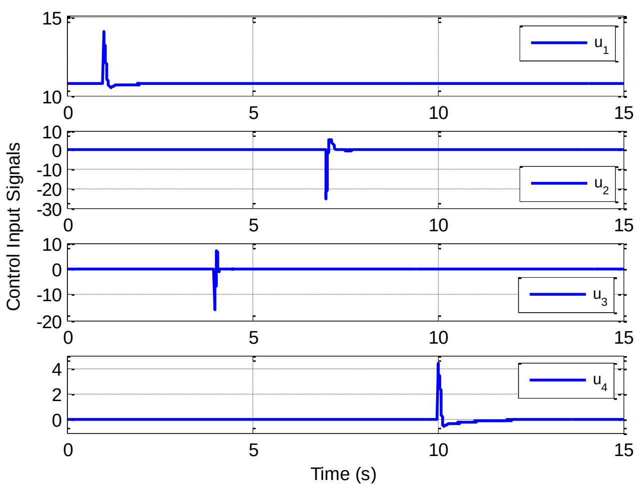


Fig. 5- Control Input Signals in first scenario

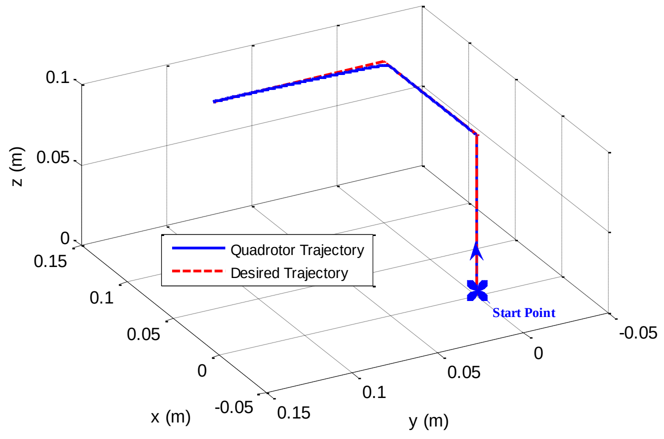


Fig. 6- Quadrotor Trajectory in first scenario

Now the performance of the control system in tracking a sinusoidal path is evaluated. For this purpose, from the beginning of the simulation, the desired sinusoidal output in the x and y channels and the desired slope output in the height axis are considered. By performing the simulation, changes in position and angle variables are plotted in Figures (7) and (8). As can be seen, by applying the proposed control system, the quadrotor position states follow the desired positions with high accuracy and the tracking errors are converged to zero. Also, by applying the proposed control system, the quadrotor angles follow the desired angles with high accuracy.

Figure (9) shows the changes in the control input signals in this scenario. It is observed that the control inputs have a smooth changes in the acceptable range to track the related outputs. Figure (10) also shows the quadrotor path and the desired path. It is observed that the desired sinusoidal path is tracked with high accuracy.

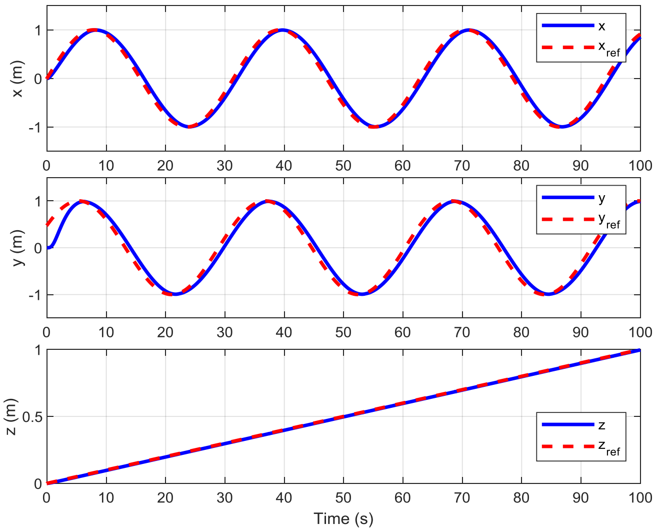


Fig. 7- Position variables in tracking the sinusoidal path

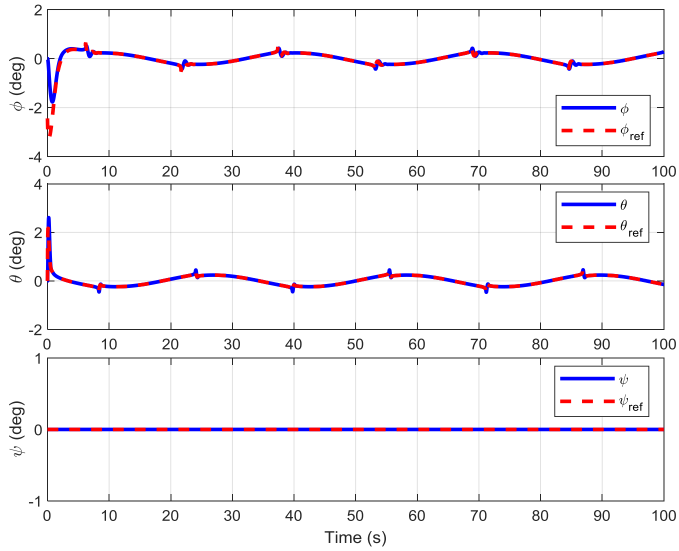


Fig. 8- Angle variables in tracking the sinusoidal path

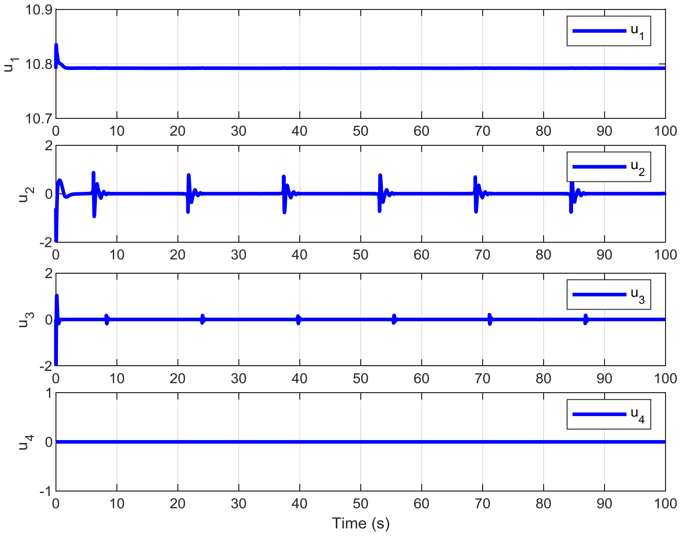


Fig. 9- Control inputs in tracking the sinusoidal path

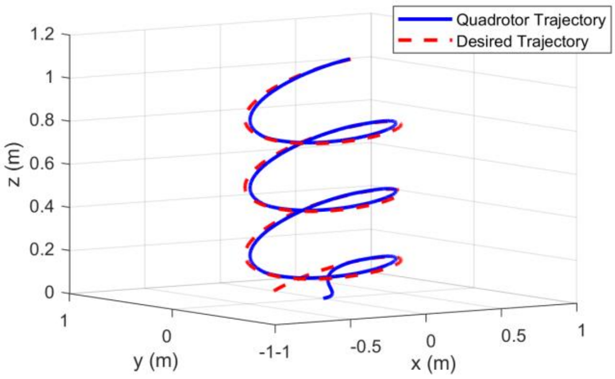


Fig. 10- Quadrotor path in tracking the sinusoidal path

V. CONCLUSION

In this paper, a finite time backstepping controller is designed to control a quadrotor. For this purpose, the dynamics of quadrotor were considered as four independent subsystems. These four sections include the z subsystem, the x position subsystem, the y position subsystem, and the ψ angle subsystem. In the x and y position subsystems, the angles θ and ϕ are used as auxiliary variables. The simulation results show that the proposed controller has a successful performance in controlling the position and angles of the quadrotor in a finite time.

REFERENCE

[1] Y. Wang, J. Sun, H. He and C. Sun, "Deterministic Policy Gradient With Integral Compensator for Robust Quadrotor Control," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 50, No. 10, pp. 3713-3725, 2020.

[2] J. M. Vazquez-Nicolas, E. Zamora, I. González-Hernández, et al, "PD+SMC Quadrotor Control for Altitude and Crack Recognition Using Deep Learning", *Int. J. Control Autom. Syst*, Vol.18, pp. 834–844, 2020.

[3] A. Hamadi, A. Ismail, "Quadrotor Modeling and Control using PID Technique", *International Research Journal of Engineering and Technology*, Vol. 6, No. 3, 2019.

[4] E. Okyere, A. Bousbaine, G. T. Poyi, A. K. Joseph and J. M. Andrade, "LQR controller design for quad-rotor helicopters," *Journal of Engineering*, Vol. 2019, No. 17, 2019.

[5] N. Setiawan, G. Nugraha P. Pratama, H. A. Winarno, S. Herdjunanto, and A. I. Cahyadi, "Attitude Tracking Control on SO(3) Group with Linearization on Moving Operating Point for Transporting Quadrotor," *IAENG International Journal of Computer Science*, Vol. 48, No.4, pp. 940-951, 2021.

[6] G. Yu, D. Cabecinhas, R. Cunha and C. Silvestre, "Nonlinear Back stepping Control of a Quadrotor-Slung Load System," *IEEE/ASME Transactions on Mechatronics*, Vol. 24, No. 5, 2019.

[7] N. Koksal, H. An, B. Fidan, Back stepping -based adaptive control of a quadrotor UAV with guaranteed tracking performance, *ISA Transactions*, 2020,

[8] J. Ghommam, L. F. Luque-Vega, M. Saad "Back stepping –Based Nonlinear RISE Feedback Control for an Underactuated Quadrotor UAV without Linear Velocity Measurements". *Springer New Developments and Advances in Robot Control, Studies in Systems, Decision and Control*, Vol. 175, 2019.

[9] J. J. Xiong, G. B. Zhang, "Global fast dynamic terminal sliding mode control for a quadrotor UAV", *ISA Transactions*, Vol. 66, 2017.

[10] Z. Jia, J. Yu, Y. Mei, Y. Chen, Y. Shen, X. Ai, "Integral back stepping sliding mode control for quadrotor helicopter under external uncertain disturbances", *Aerospace Science and Technology*, Vol. 68, 2017.

[11] O. Mofid, S. Mobayen, "Adaptive sliding mode control for finite-time stability of quad-rotor UAVs with parametric uncertainties", *ISA Transactions*, Vol. 72, 2018.

[12] B. Subartini, S. Sundarapandian V. A. Sambas, and S. Zhang, "Multistability in the Finance Chaotic System, Its Bifurcation Analysis and Global Chaos Synchronization via Integral Sliding Mode Control," *IAENG International Journal of Applied Mathematics*, Vol. 51, No. 4, pp. 995-1002, 2021.

[13] D. Zhou, Sh. Sun, "Guidance Laws with Finite Time Convergence", *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 6, 2009.

[14] R. Seeber, M. Horn, "Stability proof for a well-established super-twisting parameter setting", *Automatica*, Vol. 84, 2017, pp. 241-243.

[15] U. P. Ventura, L. Fridman, "Design of super-twisting control gains: A describing function based methodology", *Automatica*, Vol. 99, 2019, pp. 175-180.

[16] J. Liu, L. Gao, J. Zhang, F. Yan, "Super-twisting algorithm second-order sliding mode control for collision avoidance system based on active front steering and direct yaw moment control", *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, Vol 235, No. 1, 2021, pp. 43-54.

[17] Q. Hou and S. Ding, "GPIO Based Super-Twisting Sliding Mode Control for PMSM," *IEEE Transactions on Circuits and Systems II: Express Briefs*, Vol. 68, No. 2, pp. 747-751, 2021.

[18] V. I. Utkin, A. S. Poznyak, "Adaptive sliding mode control with application to super-twist algorithm: Equivalent control method", *Automatica*, Vol. 49, 2013.

[19] R. Sadeghi, S. M. Madani, M. Ataei, M. R. Agha Kashkooli and S. Ademi, "Super-Twisting Sliding Mode Direct Power Control of a Brushless Doubly Fed Induction Generator," *IEEE Transactions on Industrial Electronics*, Vol. 65, No. 11, pp. 9147-9156, 2018.

[20] S. Bouabdallah and A. Noth, R. Siegwart, PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor, *Proceedings of the IEEE International Conference of Intelligent Robots Systems*, Sendai, 2004.

[21] F. S. Safayi. *Optimal sliding mode controller design for quadrotor*. Master Thesis, Islamic Azad University, Damavand, 2016.

[22] X. Huang, Global finite-time stabilization of a class of uncertain nonlinear systems, *Automatica*, 2005.

[23] Y. G. Hong, Finite Time Stabilization and Stability of a Class of Controllable Systems, *Systems and Control Letters*, Vol. 46, 2002.

[24] Y. Hong, J. Huang, and Y. Xu, On an Output Feedback Finite-Time Stabilization Problem, *IEEE Transactions on Automatic Control*, Vol. 46, 2001.

[25] S. P. Bhat and D.S. Bernstein. Finite time stability of continuous autonomous systems. *SIAM J. Control Optim*, Vol. 38, 2000.