A Fractional-Order Sliding Mode Controller Design for Trajectory Tracking Control of an Unmanned Aerial Vehicle

Kaan Can[1], Kamil Orman[2], Abdullah Basci[1], Adnan Derdiyok[3]
[1]*Department of Electrical & Electronics Engineering, Ataturk University, 25240, Erzurum, Turkey*
[2]*Department of Computer Engineering, Erzincan Binali Yildirim University, 24100, Erzincan, Turkey*
[3]*Faculty of Technology, Sakarya University, Electrical & Electronics Engineering, 54187, Sakarya, Turkey*
abasci@atauni.edu.tr

***Abstract*—In this paper, a fractional-order sliding mode controller (FOSMC) is designed and applied to a four-rotor unmanned aerial vehicle (Quadrotor) to perform trajectory tracking for different time-varying references. Because quadrotor's nonlinear system dynamics are effected by external disturbances and parameter variations easily, the FOSMC is used as a nonlinear controller to combine the flexibility of the fractional-order approach and the disturbance rejection characteristics of the integer-order sliding mode controller (IOSMC) to keep quadrotor on desired trajectory, as well as overcome parameter variations. In order to indicate the priority of the FOSMC, the IOSMC is also applied to quadrotor for the same references. The experimental results show that FOSMC is better than IOSMC in terms of error elimination and is good at dealing with parameter variations occurred while tracking the desired trajectory.**

***Index Terms*—Fractional-order control; Sliding mode control; Unmanned systems; Quadrotor.**

I. INTRODUCTION

The control of a quadrotor is quite difficult due to the complexity of the structure and the presence of four control inputs while having six degrees of freedom. Thus, this kind of systems are called "under-actuated systems". In addition, since the quadrotor has a nonlinear structure, it needs to be analysed accurately to keep it in the defined mission or designed trajectory. In literature, some solutions and control algorithms are proposed to design and control a quadrotor. Cowling *et al.* [1] have dealt with trajectory planning and path following of a quadrotor by using a simple linear quadratic regulator (LQR) path following controller. The simulation results show that quadrotor that modelled in simulation tracks the optimal trajectories generated in real-time. Basci [2] designed a fractional-order proportional-integral (FOPI) controller to realize trajectory-tracking control of a quadrotor in real-time experiment. Also, the proposed controller is compared with the PI controller. The experimental results show that FOPI controller has better performance than PI controller in terms of reference tracking and error elimination capability. Santos *et al.* [3] have designed a fuzzy logic controller to control a quadrotor. The simulation studies have indicated that the performance of the proposed controller is quite satisfactory in terms of stability, control, and reference tracking performance. Mian *et al.* [4] applied feedback linearization control strategy to a quadrotor for translational and rotational subsystems with a proportional-derivative (PD) controller and back stepping-based proportional–integral-derivative (PID) nonlinear controller, respectively. The simulation results show that the proposed control algorithms are able to deal with nonlinearity of the system. Bouadi *et al.* [5] presented a hybrid control approach, which is a combination of sliding mode controller (SMC) and backstepping control methods. They controlled the Quadrotor for a desired trajectory and different physical events. The simulation results indicate that they obtained satisfactory performance from the proposed hybrid controller. Can and Basci [6] presented a study on a quadrotor using a nonlinear backstepping controller. The experimental results show that the proposed controller gives satisfactory performance in terms of providing a stable flight. Basci *et al*. [7] applied a continuous SMC controller to a quadrotor for trajectory tracking. The experimental results show that SMC is better in terms of disturbances, parameter variations, and error elimination capability compared to the classical PI controller.

In this paper, a fractional-order sliding mode controller (FOSMC) is designed and applied to a quadrotor for trajectory tracking control. The experimental results show that the FOSMC is more robust to parameter uncertainty and better than SMC in terms of reference tracking, error percentages and gives fast response to the changes in the references.

II. MATHEMATICAL MODEL OF QUADROTOR

The quadrotor has six degrees of freedom in space [8]. From Fig. 1, which is used as an experimental setup for this paper, the inertia axis can be expressed as $I=\left[e_x,\ e_y,\ e_z\right]$

Manuscript received 28 February, 2020; accepted 16 June, 2020.

and the body axis can be expressed as $B=[e_1,\ e_2,\ e_3]$. The position of quadrotor is denoted by $\xi=(x,\ y,\ z)^{\mathrm{T}}$, linear velocity is denoted by $v=(\dot{x},\ \dot{y},\ \dot{z})^{\mathrm{T}}$, and angles are denoted by $\eta=(\varphi,\ \theta,\ \psi)^{\mathrm{T}}$ [7]. The relationship between the vehicle body axis and the axis of inertia are represented by the rotation matrix $R:I\rightarrow B$ and the angular velocity is $\Omega=(p,\ q,\ r)^{\mathrm{T}}$.

In this paper, we have used Parrot AR.Drone 2.0 quadrotor as a low-cost experimental setup for validating our control algorithm. Our quadrotor has ARM Cortex A8 1 GHz central processing unit (CPU), DDR2 128 MB memory, and also it has LINUX OS 2.6.32. The quadrotor has 4 brushless motor having 28500 rpm and 14.5 W. Its flight speed is 5 m/sec.

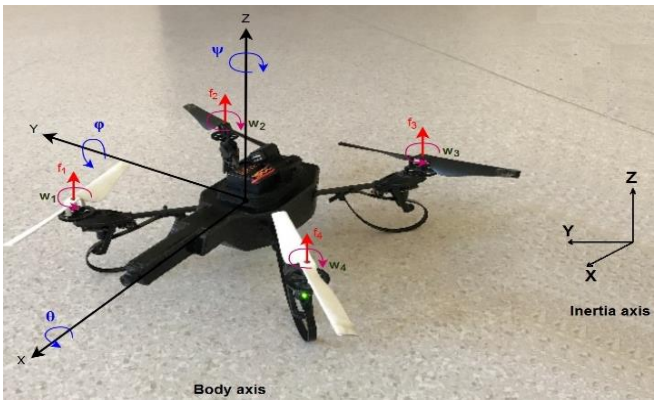


Fig. 1. The Parrot AR.Drone 2.0 and its axes [7], [9].

Besides, the whole dynamic equations of the vehicle can be written as given below [7], [10], [11]:

$$\begin{cases}\dot{x}=v_x,\\ \dot{y}=v_y,\\ \dot{z}=v_z,\\ \dot{v}_x=-(\sin(\psi)\sin(\theta)\cos(\phi)-\cos(\psi)\sin(\phi))\dfrac{T}{m},\\ \dot{v}_y=-(\cos(\psi)\sin(\theta)\cos(\phi)+\sin(\psi)\sin(\phi))\dfrac{T}{m},\\ \dot{v}_z=\cos(\theta)\cos(\phi)\dfrac{T}{m}-g,\\ \dot{\phi}=p+\sin(\phi)\tan(\theta)q+\cos(\phi)\tan(\theta)r,\\ \dot{\theta}=\cos(\phi)q-\sin(\phi)r,\\ \dot{\psi}=\sin(\phi)\sec(\theta)q+\cos(\phi)\sec(\theta)r,\\ \dot{p}=\left(\dfrac{I_y-I_z}{I_x}\right)qr+\left(\dfrac{I_r}{I_x}\right)q\Theta+\dfrac{\tau_1}{I_x},\\ \dot{q}=\left(\dfrac{I_z-I_x}{I_y}\right)pr+\left(\dfrac{I_r}{I_y}\right)q\Theta+\dfrac{\tau_2}{I_y},\\ \dot{r}=\left(\dfrac{I_x-I_y}{I_z}\right)qp+\dfrac{\tau_3}{I_z},\end{cases}\tag{1}$$

where "$I_{x,y,z}$" represents the diagonal inertia matrix of the quadrotor body frame in the X-Y-Z axis, $\Theta=-\omega_1+\omega_2-\omega_3+\omega_4$ represents the waste cycles per minute in a counter-clockwise direction, causing the gyroscopic effect on the rotor, "g" is gravity, "G" is gyroscopic torque, $\tau=[\tau_1\,\tau_2\,\tau_3]^T$ are torques, and "T" is the total thrust force generated by the rotor. Thus, the whole system model of a quadrotor with twelve state-space equations can be represented mathematically in (1) [11]. Also, the thrust force is given as follows

$$T=\sum_{i=1}^{4}f_i,\tag{2}$$

where "f_i" is generated lift force by each i. rotor and "w_i" is the rotor angular velocity and can be written as in (3)

$$f_i=b\omega_i^2.\tag{3}$$

The torque caused by gyroscopic effect can be given in (4)

$$G=\sum_{i=1}^{4}I(\Omega\times e_z)(-1)^{i+1}\omega_i.\tag{4}$$

Then, the expressions of the total torques can be written as follows:

$$\begin{bmatrix}\tau_1\\ \tau_2\\ \tau_3\end{bmatrix}=\begin{bmatrix}b(\omega_4^2-\omega_2^2)\\ b(\omega_3^2\text{-}\omega_1^2)\\ d(\text{-}\omega_1^2+\omega_2^2\text{-}\omega_3^2+\omega_4^2)\end{bmatrix},\tag{5}$$

where b is thrust factor and d is drift factor. Reference position and reference angles are defined as follows:

$$\begin{cases}\xi_r(t)=\left[x_r(t),y_r(t),z_r(t)\right]^T,\\ \eta(t)=\left[\varphi_r(t),\theta_r(t),\psi_r(t)\right]^T.\end{cases}\tag{6}$$

The error signals that occur while tracking the reference route can be defined as in (7)

$$E_\xi=\left[\xi-\xi_r,\dot{\xi}-\dot{\xi}_r\right]^T,\ \ E_\eta=\left[\eta-\eta_r,\ \dot{\eta}-\dot{\eta}_r\right]^T.\tag{7}$$

Taking derivative of the both error signals, the following equations can be written:

$$\dot{E}_\xi=A_1E_\xi+B_1(\ddot{\xi}-\ddot{\xi}_r),\ \ \dot{E}_\eta=A_1E_\eta+B_1(\ddot{\eta}-\ddot{\eta}_r),\tag{8}$$

where $A_1\in\Re^{6\times6}$ and $B_1\in\Re^{6\times3}$ are defined as given below

$$A_1=\begin{bmatrix}0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\\0&0&0&0&0&0\\0&0&0&0&0&0\\0&0&0&0&0&0\end{bmatrix},\ \ B_1=\begin{bmatrix}0&0&0\\0&0&0\\0&0&0\\1&0&0\\0&1&0\\0&0&1\end{bmatrix}.\tag{9}$$

In (10), the dynamic equations are derived depending on $\ddot{\xi}$ and $\ddot{\eta}$, while the outer loop and the inner loop are coupled via a cascaded structure [12]:

$$\begin{cases} m\dot{v} = TRe_z - mge_z, \\ I\dot{\Omega} = -\Omega \times I\Omega - G + \tau, \\ \dot{\xi} = v,\ \dot{\eta} = \Omega, \\ m\ddot{\xi} = TRe_z - mge_z, \\ I\ddot{\eta} = -\dot{\eta} \times I\dot{\eta} - G + \tau, \\ \dot{R} = R \times sk(\Omega). \end{cases} \tag{10}$$

Therefore, we may consider the quadrotor's structure as a coupled two linear subsystems. These two linear subsystems are connected to each other by means of a vector, $\gamma = [\gamma_1, \gamma_2, \gamma_3]^T \in \Re^3$ called as a virtual control vector to control the system in the cascade structure. After this definition, the cascaded system can be obtained by (11) in terms of $\ddot{\xi}$ and $\ddot{\eta}$ as in (11):

$$\begin{cases} \dot{E}_\xi = A_1 E_\xi + B_1(\gamma - \ddot{\xi}_r) + B_1(\frac{T}{m} Re_z - ge_z - \gamma), \\ \dot{E}_\eta = A_1 E_\eta + B_1(I^{-1}(\tau - \dot{\eta} \times I\dot{\eta} - G) - \ddot{\eta}_r). \end{cases} \tag{11}$$

$f_\Delta(T, \gamma, e_z) = B_1(\frac{T}{m} Re_z - ge_z - \gamma)$ is unifying term that provides the cascaded connection between the outer loop and inner loop subsystems [13]. Analysing the studies given in [11], [14], the virtual control signal $\gamma = [\gamma_1, \gamma_2, \gamma_3]^T$ can be written as in (12):

$$\begin{cases} \gamma_1 = \frac{T}{m}(\cos(\psi_r)\sin(\theta_r)\cos(\varphi_r) + \sin(\psi_r)\sin(\varphi_r)), \\ \gamma_2 = \frac{T}{m}(\cos(\psi_r)\sin(\theta_r)\cos(\varphi_r) - \sin(\psi_r)\sin(\varphi_r)), \\ \gamma_3 = \frac{T}{m}(\cos(\theta_r)\cos(\varphi_r) - g). \end{cases} \tag{12}$$

III. CONTROLLER DESIGN

The FOSMC algorithm of the system is presented in this section. The fractional-order calculation (FOC) is very attractive method to obtain more sensitive control achievement. The FOC can be represented with a ${}_aD_t^p$ differentiator for a general fundamental operator. Here, the limits of the calculation are denoted by a and t, respectively. The FOC integral and differentiator are expressed as in (13) [15]

$${}_aD_t^p = \begin{cases} \frac{d^p}{dt^p}, & p \rangle 0, \\ 1, & p = 0, \\ \int_a^t (dt)^{-p}, & p \langle 0, \end{cases} \tag{13}$$

where p represents the fractional order. In this paper, to determine the a, t, and p parameters, trial and error method is used. To obtain the FOSMC, a second order system is defined as in (14) to obtain control algorithm, and its parameter error can be expressed as in (15) [16], [17]:

$$m\ddot{x}(t) + b\dot{x}(t) + kx(t) = u, \tag{14}$$

$$\varepsilon(t) = x_{ref}(t) - x(t). \tag{15}$$

The errors of the system along the x, y, and h axes can be written as given below

$$\begin{cases} \varepsilon_1(t) = x_r(t) - x_m(t), \\ \varepsilon_2(t) = y_r(t) - y_m(t), \\ \varepsilon_3(t) = h_r(t) - h_m(t). \end{cases} \tag{16}$$

Then, using $\dot{x} = v$ transformation, the system can be rewritten as a first-order system as given below [17]:

$$\ddot{x} = \dot{v} = \frac{1}{m}(u - bv - kx) \tag{17}$$

$$\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{-k}{m} & \frac{-b}{m} \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} u. \tag{18}$$

Taking into account of (18), the system can be rewritten as a state-space form

$$[\dot{X}] = f(x,t) + Bu, \tag{19}$$

where $f(x,t) = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{b}{m} \end{bmatrix} \begin{bmatrix} x \\ v \end{bmatrix}$ and $B = \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}$. Besides, the sliding surface can be denoted as in (20) [16]–[19]

$$s = \{x : \sigma(x,t) = 0\}. \tag{20}$$

From (20), the sliding function can be expressed as in (21)

$$\sigma = C(x_{ref} - x) = C\varepsilon = \beta(t) - \delta(x), \tag{21}$$

where C is the design parameter of the system. Taking time derivative of (21) and the help of (19), (22) can be obtained

$${}_\alpha D_t^p \sigma = \alpha D_t^p \beta(t) - C(f(x,t) + Bu). \tag{22}$$

Then, $\dot{\sigma} = 0$, the equivalent control $(u = u_{eq})$ is found:

$${}_\alpha D_t^p \sigma|_{u=u_{eq}} = \alpha D_t^p \beta(t) - C(f(x,t) + Bu_{eq}) = 0, \tag{23}$$

$$CBu_{eq} = {}_\alpha D_t^p \beta(t) - C(f(x,t)). \tag{24}$$

While the system on the sliding surface, the equivalent control is used. However, to reach this phase, the system must be controlled by another control phase to drive the

system to the sliding surface. Thus, the control signal named as reaching signal can be derived by means of a positive definite Lyapunov function that can be selected as in (25) [19]

$$V=\frac{1}{2}\sigma^T\sigma>0. \tag{25}$$

In (26), the differentiation of (25) with respect to time is given

$$\dot{V}=\sigma^T\,{}_{\alpha}D_t^p\sigma. \tag{26}$$

A solution $\sigma(x,t)=0$ will be stable if time derivative of the Lyapunov function is expressed as [18], [19]

$$\dot{V}=-\sigma^T Q\sigma<0, \tag{27}$$

where Q is a positive definite matrix. Therefore, as seen from (27), the derivative of (25) will be negative with respect to time and the stability of the system is ensured [18]. Moreover, (26) and (27) are equalled to each other, then lead to

$$\sigma^T\,{}_{\alpha}D_t^p\sigma=-\sigma^T Q\sigma<0. \tag{28}$$

For (28), a solution can be written as in (29)

$$\sigma^T({}_{\alpha}D_t^p\sigma+Q\sigma)=0. \tag{29}$$

Moreover, the time derivative of the σ is given below

$${}_{\alpha}D_t^p\beta(t)-C(f(x,t)+Bu)+Q\sigma=0. \tag{30}$$

Using (24), (30) can be rewritten as

$$CBu_{eq}-CBu+Q\sigma=0. \tag{31}$$

From (31) and the result of the short algebra, the control signal can be obtained as in (32)

$$u=u_{eq}+K\sigma, \tag{32}$$

where $K=(CB)^{-1}Q$. While calculating the u_{eq}, due to a well-known disaffect called "chattering", we prefer to design (32) with a low-pass filter that is the average of the sum of the all equivalent control signals in the system to overcome this unwanted effects. With this method, the control signal can be rewritten as given below

$$u=u_{eq}+\Delta u+K\sigma, \tag{33}$$

where Δu represents the sum of the high frequency part of the u [16]–[19].

Choosing $u_{eq}+\Delta u=\frac{1}{1+\tau s}u$, and then (33) can be rewritten as in (34)

$$u=\frac{1}{1+\tau s}u+K\sigma. \tag{34}$$

To obtain more robust control strategy, σ can be determined as in (35)

$$\sigma={}_{\alpha}D_t^p\varepsilon+C\varepsilon. \tag{35}$$

Finally, substituting (35) into (34), the general control signal form is achieved as in (36)

$$u=\frac{1}{1+\tau s}u+K({}_{\alpha}D_t^p\varepsilon+C\varepsilon). \tag{36}$$

When (36) is rewritten as a position control signal of the quadrotor, the control signal equations can be obtained as seen in (37), (38), and (39) to realize the outer loop control of the system along the *x*, *y*, and *h* axes:

$$\gamma_1(t)=\frac{1}{1+\tau s}\gamma_1(t)+K_1\left({}_{a}D_t^p\varepsilon_1+C\varepsilon_1\right), \tag{37}$$

$$\gamma_2(t)=\frac{1}{1+\tau s}\gamma_2(t)+K_2\left({}_{a}D_t^p\varepsilon_2+C\varepsilon_2\right), \tag{38}$$

$$\gamma_3(t)=\frac{1}{1+\tau s}\gamma_3(t)+K_3\left({}_{a}D_t^p\varepsilon_3+C\varepsilon_3\right), \tag{39}$$

where K_1, K_2, and K_3 are the multiplier coefficients of the control signals. Also, the control block diagram of the quadrotor is given in Fig. 2.

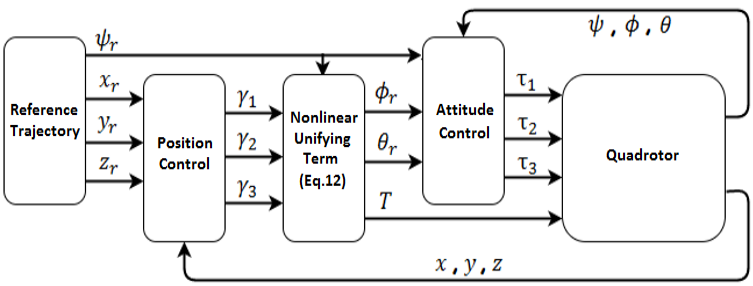


Fig. 2. The control block diagram of the system [7].

IV. THE EXPERIMENTAL RESULTS

In this part of the paper, the experimental results of both controllers are given for two different reference routes. First, the inclined circle route has been applied to the system to indicate the tracking performances in the *x*, *y*, and *h* axes for the FOSMC and the IOSMC as seen in Fig. 3 and Fig. 4,

respectively.

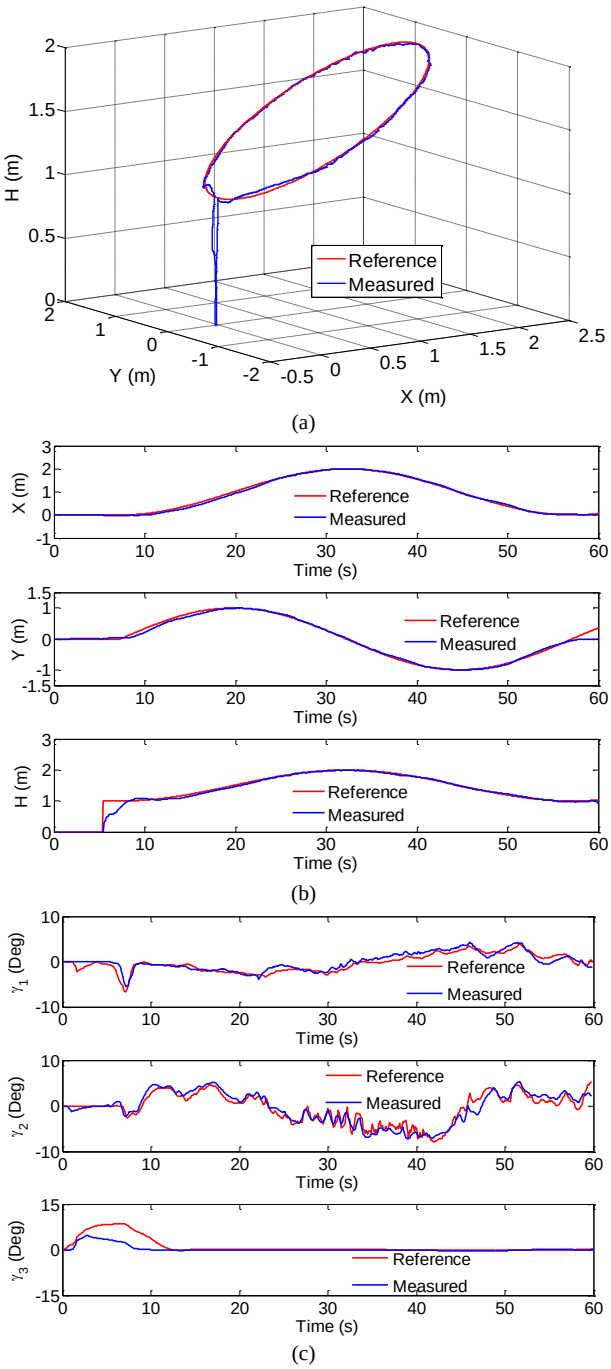
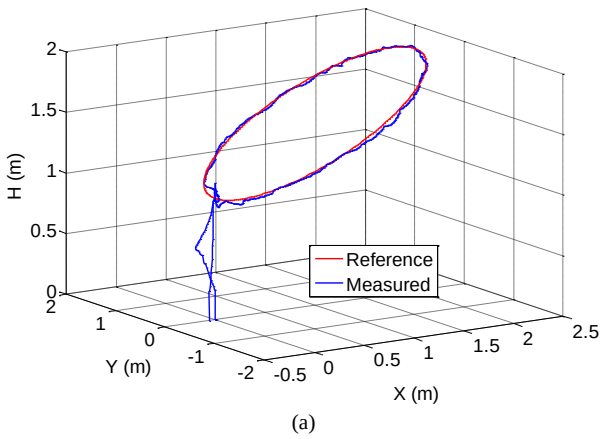


Fig. 3. The 3D oblique circle route tracking result for FOSMC (a), the oblique circle route tracking results throughout *x*, *y* and *h* axes for FOSMC (b), and the generated virtual control signals by FOSMC under oblique circle route (c).



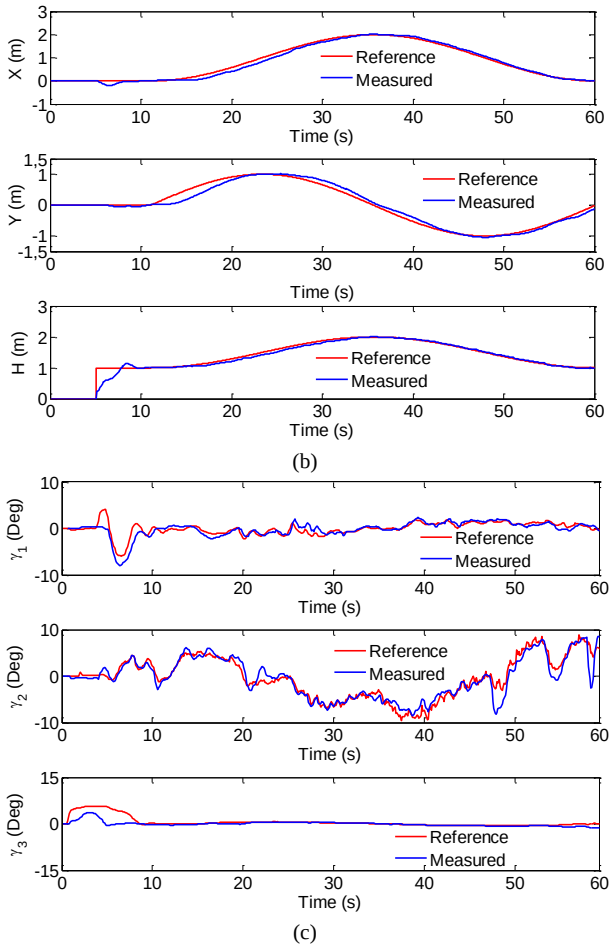
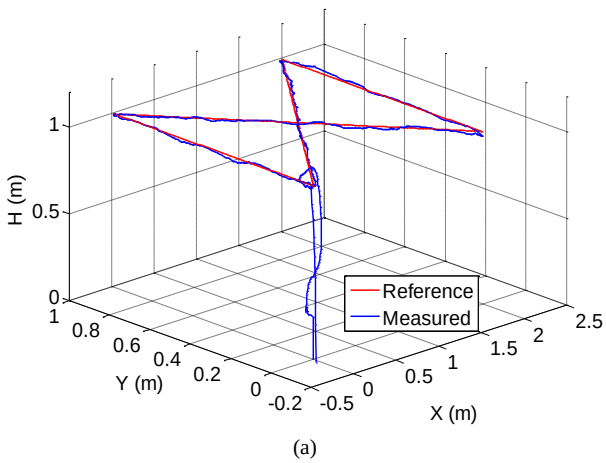
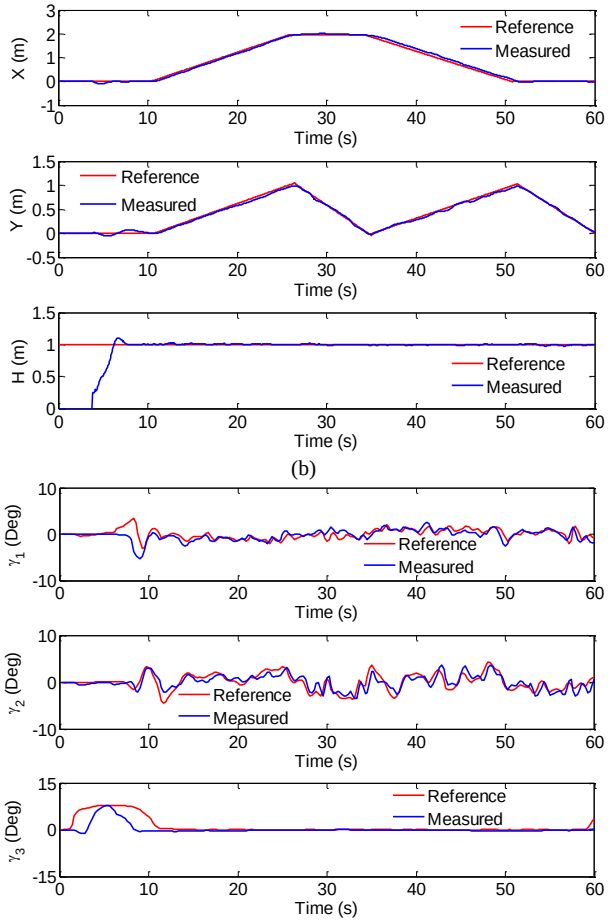


Fig. 4. The 3D oblique circle route tracking result for IOSMC (a), the oblique circle route tracking results throughout *x*, *y* and *h* axes for IOSMC (b), and the generated virtual control signals by IOSMC under oblique circle route (c).

The FOSMC has better rise and settling time performances with less overshoot for *h* axis when it is compared with the IOSMC as seen in Fig. 3(b). In addition, from the Fig. 4(b), the IOSMC has more deviations through the *x* and *y* axes, whereas the FOSMC has less delay time and better tracking performances. Although, the IOSMC has lower amplitude control signals that are needed for inner loop to generate suitable Euler angles, the FOSMC has shown better performance to eliminate the tracking errors and be able to keep the quadrotor on the desired way than the IOSMC.

In Fig. 5 and Fig. 6, the zigzag reference route results of both controllers have been presented.

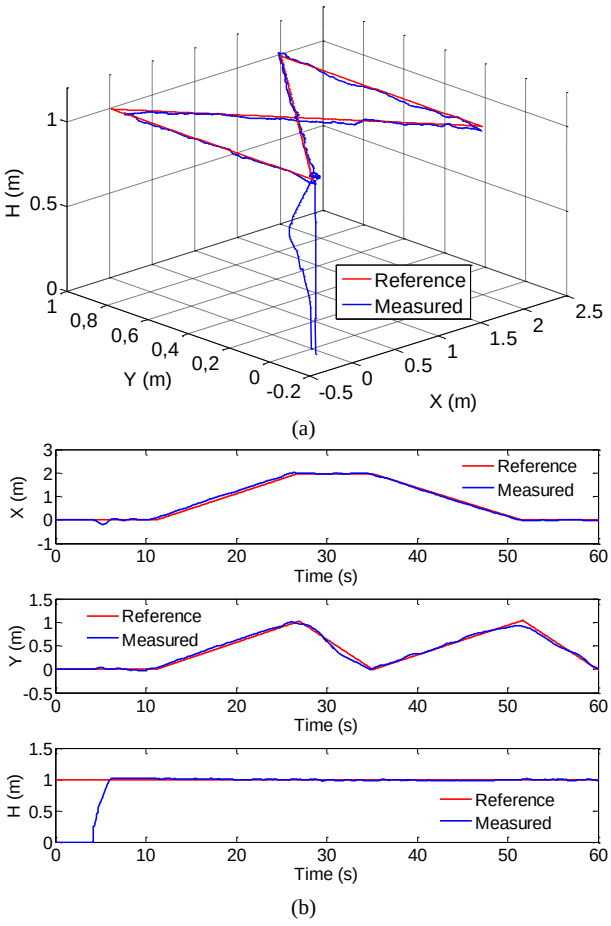




(b)

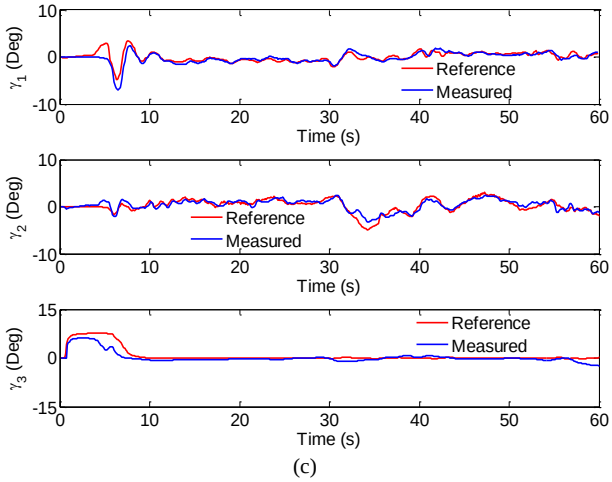
(c)

Fig. 5. The 3D zigzag route tracking result for FOSMC (a), the zigzag route tracking results throughout *x*, *y* and *h* axes for FOSMC (b), and the generated virtual control signals by FOSMC under zigzag route (c).



(a)

(b)



(c)

Fig. 6. The 3D zigzag route tracking result for IOSMC (a), the zigzag route tracking results throughout *x*, *y* and *h* axes for IOSMC (b), and the generated virtual control signals by IOSMC under zigzag route (c).

The zigzag route has sudden changes between *x* and *y*-axes, thus it is important to show the controllers' responses against to these changes. In Fig. 5(a), although the FOSMC has bigger overshoot for *h* axis than the IOSMC, the IOSMC has more deviations especially along the *y*-axes as seen in Fig 6(b). However, both controllers have missed some part of the reference route, the FOSMC has better performance in terms of error elimination, keeping the quadrotor on the route, and having less chattering level for generated control signals.

V. CONCLUSIONS

In this paper, the FOSMC has been designed and applied to the quadrotor to indicate its fractional behaviour against to disturbances, such as sensor noises, contrary wind effect, as well as robustness to parameter variations and trajectory tracking performance. Also, to compare the FOSMC, the IOSMC has been implemented on the quadrotor for the same routes to control of this unstable system. The experimental results show that the FOSMC has less trajectory tracking error with small deviations while tracking the inclined circle and zigzag routes, whereas the IOSMC has more tracking error, as well as having more overshoot/undershoot. Also, the FOSMC has less delay time and produced lower amplitude control signals that is very suitable for the inner loop to produce needed Euler angles for the system. Moreover, the control signals produced by the both controllers have nearly the same chattering level when compared their trajectory tracking performances.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] I. Cowling, "Towards autonomy of a quadrotor UAV", Ph.D. dissertation, Dept. School of Aerospace, Transport and Manufacturing, Cranfield Univ., U. K, 2008.

[2] A. Basci, "Real time trajectory tracking control of a quadrotor using fractional pi control", *Iğdır Univ. J. Inst. Sci. & Tech.*, vol. 6, no. 3, pp. 49–56, 2016. DOI: 10.21597/jist.2016321839.

[3] M. Santos, V. López, and F. M. Telefónica, "Intelligent fuzzy controller of a quadrotor", in *Proc. of 2010 IEEE International Conference on Intelligent Systems and Knowledge Engineering*, 2010, pp. 141–146. DOI: 10.1109/ISKE.2010.5680812.

[4] A. A. Mian and W. Daobo, "Modeling and backstepping-based nonlinear control strategy for a 6 DOF quadrotor helicopter", *Chinese Journal of Aeronautics*, vol. 21, no. 3, pp. 261–268, 2008. DOI:

10.1016/S1000-9361(08)60034-5.

[5] H. Bouadi and M. Bouchoucha, "Modelling and stabilizing control laws design based on backstepping for an UAV type-quadrotor", *IFAC Proceedings Volumes*, vol. 40, no. 15, pp. 245–250, 2007. DOI: 10.3182/20070903-3-FR-2921.00043.

[6] K. Can and A. Basci, "Real time trajectory tracking control of a four rotor UAV with backstepping controller", *Iğdır Univ. J. Inst. Sci. & Tech.*, vol. 6, no. 3, pp. 77–85, 2016. DOI: 10.21597/jist.2016321842.

[7] A. Basci, K. Can, K. Orman, and A. Derdiyok, "Trajectory tracking control of a four rotor unmanned aerial vehicle using continuous sliding mode controller", *Elektronika ir Elektrotechnika*, vol. 23, no. 3, pp. 12–19, 2017. DOI: 10.5755/j01.eie.23.3.18325.

[8] A. Tayebi and S. McGilvray, "Attitude stabilization of a VTOL quadrotor aircraft", *IEEE Transactions on Control Systems Technology*, vol. 14, no. 3, pp. 562–571, 2006. DOI: 10.1109/TCST.2006.872519.

[9] ARDrone 2.0. by Parrot Inc.

[10] K. Can, "Dört rotorlu bir insansız hava aracının kayan kipli kontrolcü ile yörünge takibinin gerçekleştirilmesi", M.S. thesis, Dept. Elect. & Electron. Eng., Ataturk University, 2017.

[11] Q. Li, "Grey-box system identification of a quadrotor unmanned aerial vehicle", M.S. thesis, Faculty of Mechanical, Maritime and Materials Engineering, Delft University of Technology, 2014.

[12] B. Zhao, B. Xian, Y. Zhang, and X. Zhang, "Nonlinear robust adaptive tracking control of a quadrotor UAV via immersion and invariance methodology", *IEEE Transactions on Industrial Electronics*, vol. 62, no. 5, pp. 2891–2902, 2015. DOI: 10.1109/TIE.2014.2364982.

[13] F. Kendoul, Y. Zhenyu, and K. Nonami, "Guidance and nonlinear control system for autonomous flight of minirotorcraft unmanned aerial vehicles", *Journal of Field Robotics*, vol. 27, no. 3, pp. 311–334, 2010. DOI: 10.1002/rob.20327.

[14] K. Orman, A. Başçi, and A. Derdiyok "Speed and direction angle control of four wheel drive skid-steered mobile robot using fractional order PI controller", *Elektronika ir Elektrotechnika*, vol. 22, no. 5, pp. 14–19, 2016. DOI: 10.5755/j01.eie.22.5.16337.

[15] K. B. Oldham and J. Spainer, *The Fractional Calculus*. Academic Press, 1974.

[16] K. Orman, K. Can, A. Basci, and A. Derdiyok, "An adaptive-fuzzy fractional-order sliding mode controller design for an unmanned vehicle", *Elektronika ir Elektrotechnika*, vol. 24, no. 2, pp. 12–17, 2018. DOI: 10.5755/j01.eie.24.2.20630.

[17] K. Orman, "Adaptif bulanık kesir dereceli kayan kipli kontrol yapısının insansız araçların denetimine uygulanması", Ph.D. dissertation, Dept. Elect. & Electron. Eng., Ataturk University, 2018.

[18] A. Basci and A. Derdiyok, "Application of chattering-free sliding mode controller in coupled tank liquid level system", *Korean J. Chem. Eng.*, vol. 30, no. 3, pp. 540–545, 2013. DOI: 10.1007/s11814-012-0177-y.

[19] A. Sabanovic, K. Jezernik, and K. Wada, "Chattering-free sliding modes in robotic manipulators control", *Robotica*, vol. 14, no. 1, pp. 17–29, 1996. DOI: 10.1017/S0263574700018907.

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 (CC BY 4.0) license (http://creativecommons.org/licenses/by/4.0/).