Original Research Article

Sliding Mode Control-Based Modeling and Simulation of a Quadcopter

Abstract:

This article discusses the use of Sliding Mode Control (SMC) for the control of a four-rotor vertical take-off and landing (VTOL) unmanned aerial vehicle (UAV). The Newton-Euler method is utilized to build the quadcopter's dynamic model. The model is divided into under-actuated and fully actuated subsystems. Even though controlling UAVs is difficult owing to their extremely nonlinear characteristics, previous experimental trials and simulation studies have proved that the sliding mode controller yields satisfactory performance and disturbance tolerance. In this study, SMC was used to control the altitude and attitude of the quadcopter. MATLAB/Simulink was used to show the quadcopter dynamic model and controller model, and the result illustrating the controller's performance in different conditions was acquired.

Keywords: Quadcopter, Unmanned air vehicle, VTOL, MATLAB/Simulink, SMC.

I- Introduction:

In recent years, quadcopters, or unmanned aircraft with four rotors have been the focus of UAV research. Although quadcopter control is difficult because of its nonlinear, instability, and vulnerability to external disturbances, building control systems for quadcopters is a growing field of study. Researchers have examined a variety of approaches for manipulating the motion of quadcopters. Sliding mode control (SMC) is a robust and efficient nonlinear control system that may be used in uncertain environments.

SMC consists of two components: a discontinuous control law that drives the error vector toward a decision rule known as the sliding surface. Second, after the error vector is restricted to the sliding surface, a continuous component of the controller acts to follow the dynamics enforced by the equations characterizing the sliding surface.

SMC in quadcopters has been the subject of several investigations, including publications[1-8]. Because of its drawbacks, such as chattering and computational time, SMC is often used in conjunction with other controllers to control the quadcopters. The attitude and attitude of quadcopters have been controlled by researchers using a variety of control strategies that combine SMC with other methods including LQR, fuzzy logic, backstepping, etc. Khebbache[9] built a quadcopter with Fuzzy Backstepping Sliding Mode Controllers. The developed controller approach employed the Backstepping technique for control, where the error signals were calculated as the sliding manifold, and a fuzzy controller was included instead of a sign function in the control law. An SMC&LQR tilt-wing VLOT quadcopter controller was developed by Oner et al. [10]. The LQR controller was used to control the vehicle's flight mode for all possible yaw angles, and SMC was used to stabilize the vehicle's attitude. When it comes to rolling and pitching, the sliding mode controller did a great job of tracking inputs, but the controller reference angles are quite large, and it achieves the correct roll and pitch angles in about one second.

Even though the quadcopter is subject to external disturbances, little attention has been paid to this problem in designing an SMC controller for the vehicle in several studies. Matouk et al. [11]proposed a MATLAB/Simulink-based second-order sliding mode control (2-SMC) for controlling a quadcopter. Using this strategy, it was possible to develop very stable control laws for each position and attitude state. The design did not address the external disturbances, and the system was not tested experimentally. Noordin et al.[12] developed a sliding mode control to keep a quadrotor's altitude and orientation stable in the presence of external disturbances. External disturbances were mostly ignored in the research, which studied an "X-configuration quadcopter" and considered saturation function. Zhu[13] developed a robust adaptive sliding mode control technique for quadrotor unmanned aerial vehicle (UAV) attitude and altitude tracking under simultaneous parametric uncertainty and persistent external disturbance. The suggested controller increased the system's robustness against parametric uncertainties. Despite the disturbances acting on the system, the suggested ASMC has errors converging within 5 seconds. In the SMC controller design, parameter tuning is one of the most difficult aspects. Dikmen et al. [14] used Ant Colony Optimization (ACO), Invasive Weed Optimization (IWO), and Firefly Optimization (FO) methods to obtain optimal Sliding Mode Controller (SMC) parameters for attitude and position control of quadcopter. The improved SMC parameters yield good results. The significant disadvantage of this study is that the algorithm requires a lot of processing power, especially in the case of FA. Higher-Order Sliding Mode Control (HOSMC) based on a super-twisting algorithm for quadcopter control was developed by Mebaye[15]. Lagrange formalism was used to derive the mathematical model which includes aerodynamic effects and gyroscopic moments.

This paper shows quadrotor attitude stabilization and attitude stabilization in MATLAB/Simulink using an SMC controller. Nonlinearity, parameter uncertainty, and external disturbances were considered in the development of the quadrotor control system. This study is noteworthy since it was able to control and maintain the attitude of a quadcopter while accounting for external disturbances caused by quadcopter components like batteries and sensors. This work has the following components: Section II of A Quadcopter Newton-Euler Formula in Dynamic. Section III covered sliding mode control. Section IV describes the development process of the SMC. Section V tests the whole model, including the SMC controller, using MATLAB/Simulink. The analysis and results are very certainly included in section VI of the research.

II- Quadcopter working principle and Dynamic Model:

Despite having 6 degrees of freedom (6-DOF), the quadcopter is an under-actuated aircraft. The vehicle features four propellers positioned orthogonally. Vehicle balance and motion are ensured by separate manipulation of propeller speed and direction. One set of rotors spins clockwise, while the other pair turns counterclockwise to keep the system stable. The quadcopter can go forward, backward, and sideways by varying the speed of the rotors. Altitude, Pitch, Roll, and Yaw are the 4 different types of quadcopter motion based on the relative motion of the four propellers.

Quadcopters can be maneuvered up and down by increasing or lowering the speed of all four propellers using the throttle. Quadcopters descend when all propellers are running at slow speeds. However, the vehicle hovers when the four propellers spin faster. The quadcopter's longitudinal axis rotation generates the roll angle. The vehicle accelerates to the left side direction while the two propellers on the right side are spinning at an increasing speed. When the two propellers on the left side of the vehicle are spinning fast, the vehicle moves in the right-side direction. Pitch is the term used to describe the quadcopter's ability to rotate around a side axis (either forward or backward). The rotation occurs when the vehicle's front propellers are spinning at a high speed. The quadcopter's vertical axis is rotated using the yaw control (either to the left or to the right). The quadcopter rotates counter-clockwise when two of its left-side propellers spin fast.

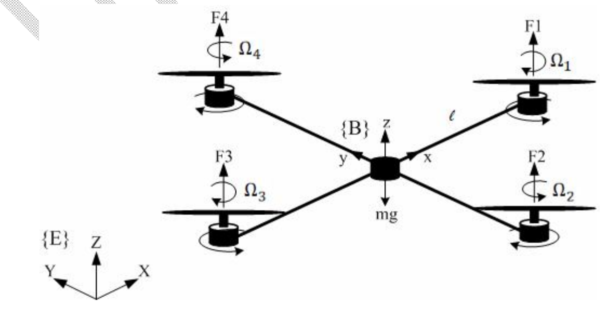

Fig.1. Quadcopter UAV configuration

The vehicle accomplishes a counterclockwise rotation when its right-side propellers spin fast. Roll, pitch, yaw, and angular velocities may be determined using an Earth-fixed Frame(E), while the linear acceleration is determined using a Body-fixed Frame(B). The X-configuration with B and E frames is shown in **Fig.1.** Three translational states (x, y, z), three rotational states (ϕ, θ, ψ), and their derivatives ($\dot{x}, \dot{y}, \dot{z}, \dot{\phi}, \dot{\theta}, \dot{\psi}$).

Where, x, y, and z are the position in the x, y, and z axes, $\dot{x}$, $\dot{y}$, and $\dot{z}$ are the angular velocities, ϕ, θ, ψ are the roll, pitch, and yaw angles, and the parameters, $\dot{\phi}$, $\dot{\theta}$, and $\dot{\psi}$ are the speed for roll, pitch, and yaw respectively. The quadrotor dynamics were expressed using Newton-Euler translational and rotational dynamics formulation Ferry [16]as:

$$\ddot{x} = \frac{1}{m}((\sin\psi \sin\phi - \cos\psi \sin\theta \sin\phi)U_1 - A_x\dot{x}) \quad (1)$$

$$\ddot{y} = \frac{1}{m}((\cos\psi \sin\,\phi + \sin\psi \sin\theta \cos\phi)U_1 - A_y\dot{y}) \quad (2)$$

$$\ddot{z} = g - \frac{1}{m}(\cos\psi \cos\phi\, U_1 - A_y\dot{y}) \quad (3)$$

$$\ddot{\phi} = \left(\frac{I_{yy}-I_{zz}}{I_{xx}}\right)\dot{\psi}\,\dot{\theta} + \left(\frac{J_r\Omega_r}{I_{xx}}\right)\dot{\theta} + \left(\frac{1}{I_{xx}}\right)U_2 - \frac{A_\phi\dot{\phi}}{I_{xx}} \quad (4)$$

$$\ddot{\theta} = \left(\frac{I_{zz}-I_{xx}}{I_{yy}}\right)\dot{\psi}\,\dot{\phi} - \left(\frac{J_r\Omega_r}{I_{yy}}\right)\dot{\phi} + \left(\frac{1}{I_{yy}}\right)U_3 - \frac{A_\theta\dot{\theta}}{I_{yy}} \quad (5)$$

$$\ddot{\psi} = \left(\frac{I_{xx}-I_{yy}}{I_{zz}}\right)\dot{\theta}\,\dot{\phi} + \left(\frac{1}{I_{zz}}\right)U_4 - \frac{A_\psi\dot{\psi}}{I_{zz}} \quad (6)$$

The input signal U_1 is the total thrust of the 4 rotors. And U_2, U_3 *and* U_4 are the moments for pitch, roll, and yaw respectively. Where m represents the mass of the quadrotor, J_r is the inertia of the rotor, and I_{xx}, I_{yy}, and I_{zz} are the inertia of the quadrotor in x, y, and z respectively. The inputs can be given by:

$$U_1 = b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \quad (7)$$

$$U_2 = b\sin\left(\frac{pi}{4}\right)(\Omega_1^2 - \Omega_2^2 - \Omega_3^2 + \Omega_4^2) \quad (8)$$

$$U_3 = b\sin\left(\frac{pi}{4}\right)(\Omega_1^2 + \Omega_2^2 - \Omega_3^2 - \Omega_4^2) \quad (9)$$

$$U_4 = d(\Omega_1^2 + \Omega_3^2 - \Omega_2^2 - \Omega_4^2) \quad (10)$$

$$\Omega_r = \Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2 \quad (11)$$

Where b is thrust coefficient, and d is drag coefficient, the angular speed for each rotor is $\Omega_1, \Omega_2, \Omega_3, \Omega_4$. Ω_r is the general angular speed.

III- Sliding Mode Controller:

Sliding Mode Control is a procedure that is evolved from Variable Structure Control (VSC). Slide mode control aims to limit the amount of error in the system. **Fig.3**. illustrates the SMC law U_X **Eqn.23.,** which has two key components: a continuous part (equivalent control U_{eqX}) and a discontinuous part (switching control $\hat{U}_X$). The switching control rule directs the system to the user-specified sliding surface, S, and maintains the system's trajectory on this surface throughout switching control rule. The equivalent control is used to guarantee that the system state goes toward the sliding surface. In attempt to develop a sliding surface or decision rule, the error must be defined. To put it another way, an error is a difference between a value's actual value and the value desired.

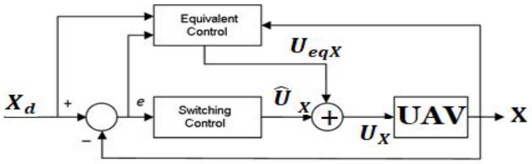

Fig.3. sliding-mode control for a quadcopter

$$e_X = X_d - X \tag{12}$$

By calculating the error's first and second derivatives:

$$\dot{e}_X = \dot{X}_d - \dot{X} \Longrightarrow \ddot{e}_X = \ddot{X}_d - \ddot{X} \tag{13}$$

The sliding surface is given according to the following equation:

$$S_X=\dot{e}_X+\lambda_X e_X \tag{14}$$

Also, the first derivative can be computing as:

$$\dot{S}_X=\ddot{e}_X + \lambda_X\dot{e}_X \tag{15}$$

Where variable e_X is the tracking error, and variable λ_X is the tuning parameter must satisfy the condition ($\lambda_X>0$), S_X is sliding surface $S_X = 0$, X is state space.
To design the control law, a Lyapunov function $V(S_X)$ is defined. This function must be positive-definite.
A Lyapunov function is defined as:

$$V(S_X)=\frac{1}{2}S_X^2 \tag{16}$$

then the desired sliding condition is verified and
Lyapunov stability is guaranteed. The chosen law for the attractive surface must satisfy

$$S_X\dot{S}_X = -k_{X1}S_X - k_{X2}\,\mathrm{sign}(S_X) < 0 \tag{17}$$

As mentioned, the control law, U_X, consists of two parts: a continuous part, U_{eqX} and a discontinuous part, $\widehat{U}_X$.

$$U_X = \widehat{U}_X + U_{eqX} \tag{18}$$

$$\widehat{U}_X = -k_{X1}S_X - k_{X2}\,\mathrm{sign}(S_X) \tag{19}$$

Where k_{X1}, k_{X2} are the tuning parameters, sign can be as:

$$\mathrm{sign}(S_X) = \begin{cases} +1 & \text{if } S_X > 0 \\ -1 & \text{if } S_X < 0 \end{cases} \tag{20}$$

IV- Quadcopter SMC Controller:

A sliding mode control (SMC) is used in this investigation to obtain the desired altitude and attitude. The proposed control law is developed by dividing the system model into two subsystems, **a fully actuated subsystem,** and **a under-actuated subsystem,** as shown in **Fig.2.**.Unlike in the under-actuated subsystem, where the inputs $U_2$and U_3 are smaller than the number of outputs (x, y, ϕ, θ), in the fully actuated subsystem there are two outputs (z, ψ) for each of the inputs (U_1, U_4).

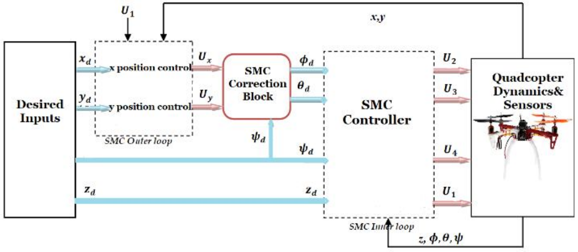

Fig.2. UAV control system block diagram.

1. Under-Actuated Subsystem Control:

It was decided to separate the control of the under-actuated subsystem into two separate blocks in the design, an inner-loop, and an outer loop. Using the correction control block (the outer loop), the desired altitude can be obtained by converging the error and extracting the desired attitude angles θ_d and $\phi_d$from x and y. In order to achieve roll and pitch control an SMC controller is employed as the inner-loop controller to drive the attitude angles and from the outer-loop controller.

Using **Eqn.1.,2.** the following x, y position equation may be derived:

$$U_x = (\cos\phi_d \sin\theta_d \cos\psi - \sin\phi_d \sin\psi) \tag{21}$$

$$U_y = (\cos\phi_d \sin\theta_d \sin\psi - \sin\phi_d \cos\psi) \tag{22}$$

$$\begin{bmatrix} U_x \\ U_y \end{bmatrix} = \begin{bmatrix} \sin\psi & -\cos\psi \\ \cos\psi & \sin\psi \end{bmatrix} \begin{bmatrix} \cos\phi_d \sin\theta_d \\ \sin\theta_d \end{bmatrix} \tag{23}$$

$$\begin{bmatrix} \cos\phi_d \sin\theta_d \\ \sin\theta_d \end{bmatrix} = \begin{bmatrix} \sin\psi & -\cos\psi \\ \cos\psi & \sin\psi \end{bmatrix}^{-1} \begin{bmatrix} U_x \\ U_y \end{bmatrix} \tag{24}$$

By starting from the second row ϕ_d and then $\theta_d$can be calculated as follows:

$$\begin{bmatrix} \phi_d \\ \theta_d \end{bmatrix} = \begin{bmatrix} \arcsin(U_x \sin\psi - U_y \cos\psi) \\ \arcsin((U_x \cos\psi - U_y \sin\psi)/\cos\phi_d) \end{bmatrix} \tag{25}$$

2. Fully Actuated Subsystem Control:

The objective of this fully actuated subsystem controller is to minimize the error in the altitude and yaw angle e_z and e_ψ to satisfy the following conditions:

Deriving **SMC controller for altitude:**
By applying **Eqn.17. , 18. , 19. , 20.:**

$$e_z = z_d - z = 0 \Longrightarrow \dot{e}_z = \dot{z}_d - \dot{z} \Longrightarrow \ddot{e}_z = \ddot{z}_d - \ddot{z} \tag{25}$$

$$S_z = \dot{e}_z + \lambda_z e_z \Longrightarrow \dot{S}_z = \ddot{e}_z + \lambda_z \dot{e}_z \tag{26}$$

by substituting **Eqn.25, 26.** sliding condition $\dot{S}_z=0$ result is obtained.

$$\dot{S}_z = (\ddot{z}_d - \ddot{z}) + (\dot{z}_d - \dot{z})\lambda_z = 0 \tag{27}$$

$$\dot{S}_z = \left(\ddot{z}_d - g + \frac{1}{m}(\cos\psi\cos\phi U_1 + A_y \dot{y})\right) + (\dot{z}_d - \dot{z})\lambda_z \tag{28}$$

Since the system is in a sliding condition and $U_z=U_{eqz}$, $A_y\dot{y} = 0$ are considered, **Eqn.28** can be derived:

$$\dot{S}_z = \left(\ddot{z}_d - g + \frac{U_{eqz}}{m}(\cos\psi\cos\phi)\right) + \lambda_z(\dot{z}_d - \dot{z}) \tag{29}$$

The control laws for the altitude and yaw angle can be derived using classical SMC theory:

$$U_{eqz} = \left(\frac{m}{\cos\phi\cos\theta}\right)\left(\ddot{z}_d - g + \lambda_z(\dot{z}_d - \dot{z})\right) \tag{30}$$

From **Eqn.23, 24:**

$$U_1 = U_z = \hat{U}_z + U_{eqz}, \hat{U}_z = -k_{z1}S_z - k_{z2}\,\mathrm{sign}(S_z) \tag{31}$$

$$U_1 = \left(\frac{m}{\cos\phi\cos\theta}\right)(g - \ddot{z}_d - \lambda_z \dot{e}_z) + k_{z1}S_z + k_{z2}\,\mathrm{sign}(S_z) \tag{32}$$

The same approach used to derive altitude may also be used to derive SMC for roll, pitch, yaw, x, and y as illustrated below:

$$U_x = \frac{m}{U_1}(\ddot{x}_d - \lambda_x e_x) + k_{x1}S_x - k_{x2}\,\mathrm{sign}(S_x) \tag{33}$$

$$U_y = \frac{m}{U_1}(\ddot{y}_d - \lambda_y e_y) + k_{y1} S_y - k_{y2}\,\text{sign}\,(S_y) \quad (33)$$

$$U_2 = \frac{I_{xx}}{l}\left(\ddot{\phi}_d - \frac{I_{zz} - I_{xx}}{I_{yy}}\dot{\theta}\,\dot{\phi} + \lambda_\phi \dot{e}_\phi\right) + k_{\phi1} S_\phi + k_{\phi2}\,\text{sign}\,(S_\phi) \quad (34)$$

$$U_3 = \frac{I_{yy}}{l}\left(\ddot{\theta}_d - \frac{I_{zz} - I_{xx}}{I_{yy}}\dot{\theta}\,\dot{\phi} + \lambda_\theta \dot{e}_\theta\right) + k_{\theta1} S_\theta + k_{\theta2}\,\text{sign}\,(S_\theta) \quad (35)$$

$$U_4 = I_{zz}\left(\ddot{\psi}_d - \frac{I_{zz} - I_{xx}}{I_{yy}}\dot{\theta}\,\dot{\phi} + \lambda_\psi \dot{e}_\psi\right) + k_{\psi1} S_\psi + k_{\psi2}\,\text{sign}\,(S_\psi) \quad (36)$$

V- Simulation Results and Discussion:

To verify the performance of the proposed controller, a MATLAB/Simulink is presented in this section. The parameters of the quadrotor used in the simulation are selected in **Table.1.**

Table.1 shows the parameters of the quadcopter.

Parameter	Symbol	Value
Quad. mass	m	0.984 kg
Arm length	l	0.225m
Gravity	g	9.81 m/s^2
Rotor inertia	J_r	2.6e-06 $kg.m^2$
Inertia constants	I_{yy}, I_{xx}	9.5*10^{-3} $kg.m^2$
	I_{zz}	1.86*10^{-2} $kg.m^2$
Thrust Coeff.	b	1.4865e-07$N.s^2$
Drag coeffi.	d	2.925e-09 $N.m.s^2$
Aerodynamic coefficient	A_ϕ,A_θ, A_ψ	0 N/rad/s
Air Drag coeffi.	A_x,A_y,A_z	0 N/m/s

The suggested SMC approach's design parameters have been tuned manually in MATLAB/Simulink in order to track the trajectory smoothly. **Table.2.** list the parameters of the recommended controllers.

Table.2. shows SMC design parameters tuning.

Controller Tuning	SMC			
	Roll	Pitch	Yaw	Altitude
k_{X1}	1.97	1.97	1	16.33
k_{X2}	1.81	1.81	0	14.1
λ_X	3.68	3.68	18	2.5

Fig.4., 5., 6., and 7. show the actual and desired.

values of altitude, roll, pitch, and yaw.

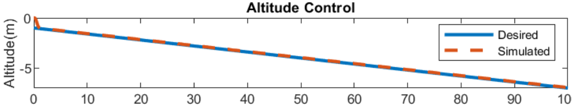

Fig.4. shows the actual and desired altitude values

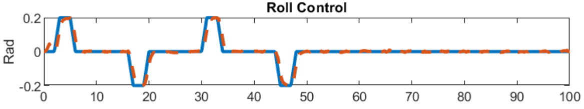

Fig.5. shows the actual and desired roll values

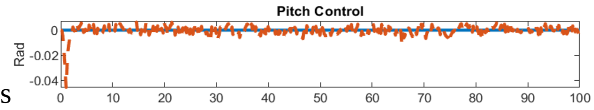

Fig.6. shows the actual and desired pitch values

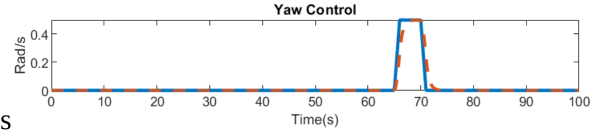

Fig.7. shows the actual and desired yaw values

Fig.4. illustrates the desired altitude and the actual values. Maintaining the desired altitude was not affected by any noticeable disturbance. And the disturbances were kept to a minimum. Roll actual and desired values are shown in **Fig.5.** With a rise time of 202.724 ms and an overshot percentage of 120.852 %, the controller was able to control the roll angle. **Fig.6.** shows the desired and actual pitch angle values. The pitch angle was achieved, although the controller recorded some disturbances. Yaw desired and actual value was shown in **Fig.7.** With a rise time of 792 ms and an overshot percentage of 0.505 %, the controller was able to follow the trajectory with neglectable disturbances. And the controller, on the other hand, has kept yaw angles stable throughout the flight. A 3D trajectory was provided in **Fig.8.** the figure depicts the overall system performance (the quadcopter with sensors and controls). The quadcopter's X, Y, and Z axes were exhibited. The vehicle's position was controlled by the controller, and a satisfactory trajectory was obtained. **Fig.9.** the gyroscope sensor's X and Y position readings are shown. According to the findings, the sensor was able to accurately measure x and y.

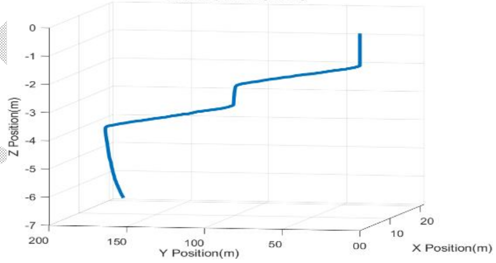

Fig.8. quadcopter trajectory

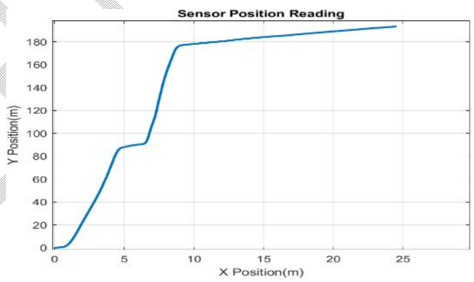

Fig.9. sensor position reading

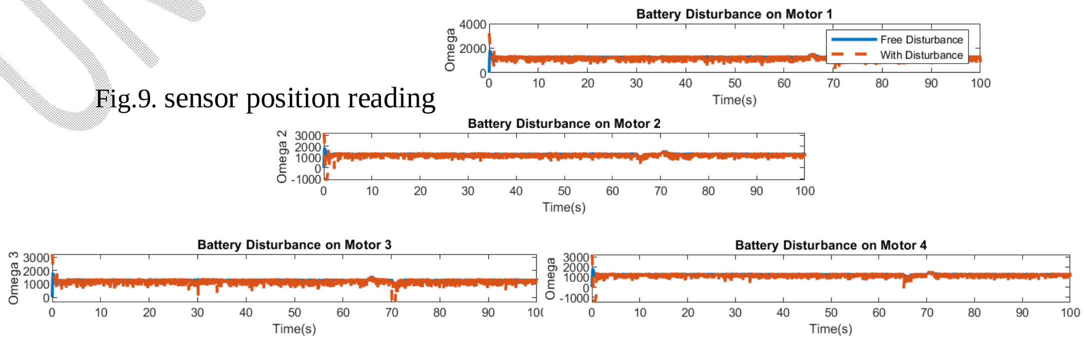

Fig.10. angular speed vs actual angular speed with battery disturbances included

Fig.11. depicts the motors' speeds. As shown, the motors spin at roughly 12000 rpm, which was successfully maintained and controlled.

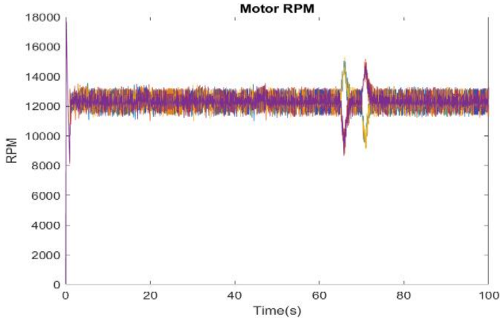

Fig.11. motors RPM

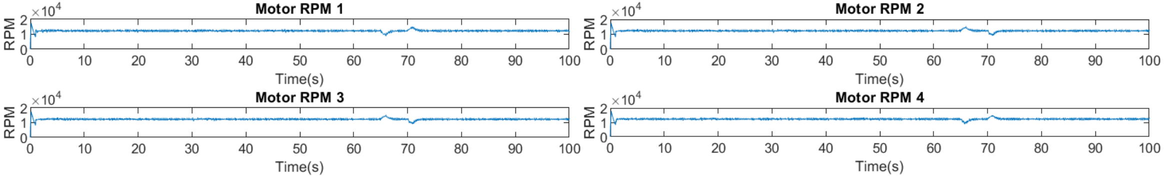

Fig.12. shows the speed of each motor (RPM)

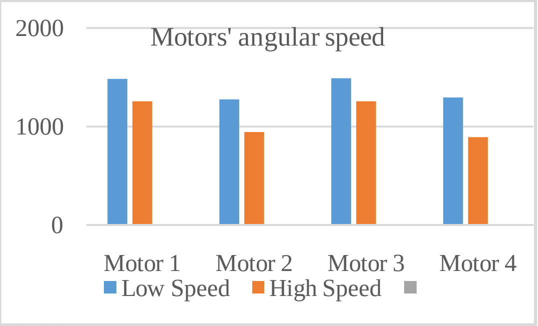

Fig.13. The motors' angular speed comparison

Simulating the spinning of the motors included considerations for battery and external disturbance. **Fig.12.** shows a speed of the motor (RPM) disturbances. Time periods 60, 66, and 70.2 s all have been effected by disturbances.

Fig.13. shows the angular speeds of all four motors. However, these impacts are insignificant in comparison to the controller as a whole, even when the angular speed is disturbed at multiple time periods.

The SMC controller rise time, and overshoot for altitude, roll, pitch, and yaw are shown in **Table.3.**

Table.3. Controllers rise and overshoot.

State Measures	Rise time(ms)	Overshoot(%)
Altitude	-	-
Roll	202.724	120.852
Pitch	-	-
Yaw	792	0.505

VI- Conclusion and Future works:

In this study, a Sliding Mode Control (SMC) for quadcopter control is discussed in detail. The Newton-Euler equations are used to build a mathematical model for the control design. Fully and under-actuated subsystems may be found in the controller. There are two basic loops in the under-actuated subsystem (inner and outer), which are needed to control the x, y position so that correction blocks may give ϕ_d and θ_d, which are then delivered to control the vehicle's attitude. In the outer loop, a correction block is employed, whereas in the inner loop, SMC is used. Altitude and heading control are controlled by an SMC in the full-actuated subsystem. The controller's robustness is increased while also considering the impacts of quadcopter-induced external disturbances. The suggested controller alleviates the chattering phenomenon resulting by SMC. A comprehensive simulation study was undertaken, and the results are explained. As a result, the designed quadcopter can be controlled effectively using the suggested SMC controller.

References:

[1] Bouabdallah, S., P. Murrieri, and R. Siegwart, Design and control of an indoor micro quadrotor, in IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004. 2004. p. 4393-4398 Vol.5.

[2] Herrera, M., et al., Sliding Mode Control: An Approach to Control a Quadrotor, in 2015 Asia-Pacific Conference on Computer Aided System Engineering. 2015. p. 314-319.

[3] Huang, T., et al., Robust Tracking Control of a Quadrotor UAV Based on Adaptive Sliding Mode Controller. Complexity, 2019. **2019**: p. 1-15.

[4] Labbadi, M. and M. Cherkaoui, Robust Integral Terminal Sliding Mode Control for Quadrotor UAV with External Disturbances. International Journal of Aerospace Engineering, 2019. **2019**: p. 1-10.

[5] Ghamry, K.A., et al., Real-Time Autonomous Take-off, Tracking and Landing of UAV on a Moving UGV Platform. 2016 24th Mediterranean Conference on Control and Automation (Med), 2016: p. 1236-1241.

[6] Ghamry, K.A., M.A. Kamel, and Y.M. Zhang, Cooperative Forest Monitoring and Fire Detection Using a Team of UAVs-UGVs. 2016 International Conference on Unmanned Aircraft Systems (Icuas), 2016: p. 1206-1211.

[7] Ghamry, K.A., M.A. Kamel, and Y.M. Zhang, Multiple UAVs in Forest Fire Fighting Mission Using Particle Swarm Optimization. 2017 International Conference on Unmanned Aircraft Systems (Icuas'17), 2017: p. 1404-1409.

[8] Labbadi, M., et al., Modeling and Robust Integral Sliding Mode Control for a Quadrotor Unmanned Aerial Vehicle. 2018 6th International Renewable and Sustainable Energy Conference (Irsec), 2018: p. 613-618.

[9] Khebbache, H. and M. Tadjine, stRobust Fuzzy Backstepping Sliding Mode Controller For a Quadrotor Unmanned Aerial Vehicle. Control Engineering and Applied Informatics, 2013. **15**(2): p. 3-11.

[10] Kaan T. Oner, E.C., Efe Sirimoglu, Cevdet Hancer, Taylan Ayken, and Mustafa Unel, LQR and SMC Stabilization of a New Unmanned Aerial Vehicle. World Academy of Science, Engineering and Technology, 2009: p. 6.

[11] Matouk, D., et al., Second-Order Sliding Mode for Position and Attitude Tracking ,Control of Quadcopter Uav: Super-Twisting Algorithm. International Journal of Innovative Computing Information and Control, 2020. **16**(1): p. 29-43.

[12] Noordin, A., M.A.M. Basri, and Z. Mohamed, Sliding mode control for altitude and attitude stabilization of quadrotor UAV with external disturbance. Indonesian Journal of Electrical Engineering and Informatics (IJEEI), 2019. **7**(2).

[13] Wang, L., C.H. , and P.Z. , Adaptive Sliding Mode Control for Quadrotor Aerial Robot with I Type Configuration. International Journal of Automation and Control Engineering (IJACE) 2013. **3**(1).

[14] Dikmen İ. C. , K.T., Yeroğlu C., Multi-Parameter Optimization of Sliding Mode Controller for Quadcopter Application. Anatolian Journal of Computer Sciences, 2018. **3**(1): p. 14-28.

[15] Mamo, M.B., Regulation Control of Quadcopter by Designing Second Order Sliding Mode Controller. Journal of Electrical Engineering, Electronics, Control and Computer Science – JEEECCS, 2020. **6**(22): p. 9-16.

[16] Ferry, N., Quadcopter Plant Model and Control System Development With MATLAB/Simulink Implementation, in Electrical Engineering 2017, Rochester Institute Of Technology.