3D CADMODEL FOR A QUADROTOR SYSTEM MODELING AND CONTROL

Fatima A. RAHEEMAH *, Mustafa T. HUSSEIN*

Department of Mechanical Engineering, University of Babylon, Iraq
* Corresponding author, e-mail: sth.fatima.ali@student.uobabylon.edu.iq

Abstract

In this paper, the work focus on the advantages of several engineering software to present a case study for the quadrotor system by a used 3D CAD model. The CAD model initially generated using one of the CAD software and it can be accessed from MATLAB software and converted into a virtual physical model. Quadrotors are unmanned aerial vehicles capable of vertical takeoff, hovering, and landing. The quadrotor is distinguished by its small size, flexibility, and maneuverability. The small sensors and actuators used in these systems are effective enough in comparison with the larger systems. The CAD model for the quadrotor system in this study is used to show the capability of the 3D model implementation in modeling and control. These models echo on the real-time models to improve the behavior in the real world. The actual quadrotor device was converted from the real system to a 3D model. Also, the model is converted to SimMechanics for the sake of simulation. Two different control methods are used in this work to stable the motion of the quadrotor system. First the adaptive sliding mode controller with the adaptive controller. Second the adaptive sliding mode controller with the PID controller. The simulation results show the model works fine with the controllers and it preserves the desired position and attitude along the desired predefined path. The results shown when a comparison was made that the ratio of error for PID controller is better.

Keywords: Quadrotor, SimMechanics/MATLAB, CAD model, Unmanned aerial vehicles

1. INTRODUCTION

Quadrotor is a four-rotor aerial vehicle capable of vertical takeoff and landing. The quadrotor is defined as an uncontrollable aerial aircraft whose direction is controlled by changing the speed of four motors. In order to attain stability, proper quadrotor control methods are required. Instead of one huge propeller, the quadrotor uses four little propellers. Small propellers reduce torque in the system, allowing the blades to be pushed quicker without producing additional mechanical vibrations and enhancing the engine's efficiency [1]. Quadrotors now include a variety of equipment on board, including global positioning system, which enables the quadrotor to correctly determine its location. Application development has also been done to make it easier to use quadrotors, such as setting up waypoints, hovering with position control, and flight trajectory following, among other things. Quadrotor has advantages that differ from helicopters, such as the low cost of production, where it is used in dangerous areas or hard to reach areas, in the military fields for espionage purposes. Also, it is used in many fields, such as mine detection, radar system, and aerial photography since the quadrotor has many characteristics. While the quadrotor offers the advantages in easy mechanical construction against the conventional helicopter, it is still not commonly used in many of the areas and applications suggested. For instance, quadrotor stabilization is a very difficult task due to the nonlinear dynamic equations [2].

In recent years, accelerated developments in the field of engineering yielded a new direction in the modeling, design, and control process of engineering systems. The developments of electronics and software open new ways to simplify even the way of thinking in engineering studies and the works. The main signs of progress in the field of engineering software made the modeling process less difficult and more effective. Nowadays, the CAD programs which produce accurate 3D models than ever had become the language of engineers. These models can describe simple or complex engineering systems from a single link mechanism to hybrid car systems. The issues of modeling and controlling the quadrupole system using 3D CAD models have been addressed by a variety of studies. This software has the advantage of producing reliable results that replicate any dynamic system while taking geometry, mass, and inertia into account. It comes with a large library of features, like as sensors and electromechanical actuators, that may be utilized to enhance the model's usefulness. Such solutions necessitate a high level of theoretical understanding

and experience in dynamic system modeling, design, and control [3]. Quadrotor UAV modeling and testing programs were employed, as well as automated navigation systems and the trajectory of the UAV. Simscape multibody, depicted in figure 1, was used to create a simulation model using SolidWorks as the CAD program, and then the CAD model was executed and exported to MATLAB/Simulink [4].

Laboratory truck cranes and forestry cranes are examples of models that can be made with CAD software. CAD designs may substantially speed up the construction process and update the plant model used in simulation studies because they are frequently generated in combination with simulations. Because it is conducted automatically rather than manually, this technique can assist reduce the number of errors when design modifications occur. Quadrupole drives are researched using CAD data. The model is created by translating CAD data to XML, which can then be read by a physical modeling program. When the dynamics of a quadrilateral system are determined using CAD rather than differential equation derivation, the process design is simplified [5].

In [6] a linear control model is proposed to create a global model that operates in all operating ranges and uses a linear time constant controller due to the system's non-linearity. The parameters of these linear models are subjected to CAD procedures. The dynamic parameters are derived from the engineering and physical properties of the components. The generated linear model has sufficient "quality" and can be utilized to build a better four-motor controller in the future.

Several researchers addressed the problems of modeling and control of the quadrotor system based on 3D CAD models. The authors in [7] he has created a CAD model in SolidWorks that estimates the moment of mass of inertia and all of the missing geometric characteristics. Control tuning, the current quadrupole model, and applications are also explored, as are chassis parameter definition, rotary assembly determination, and unmetered machine state estimate algorithms for inertial measurement unit use. The link between pulse width modulation signal, angular velocity, force, and momentary drag was investigated using three basic test pads. The stability of the quadrupole position was investigated using a control system based on linear feedback technology, which was employed to transition the system from a nonlinear to a linear state. The quad's individual properties are checked and the position is calculated using two independent testers. The experiment's findings suggest that the methods for determining the structure's parameters are accurate enough.

In [8] employed a positioning system such as global positioning system in conjunction with a CAD model to verify that the excavator can be controlled with six degrees of freedom and simulation using the Msc. Adams and MATLAB/Simulink programs. By using a 2 DOF Rototilt, the excavator was given two additional degrees of flexibility in comparison to the typical 4 DOF machine. MSc. Adams was used as a simulation platform to investigate this issue. The bucket route is calculated by using the CAD reference on the target plane.

In order to create and test control devices that will coordinate and interact with unmanned vehicles and the environment simulation technology has been developed, as shown in figure 2. Because of their large range of capabilities, modules, and cross-platform interoperability V-REP and MATLAB/ Simulink are utilized in simulation strategy. V-REP is based on the Gazebo system and offers a number of implementation benefits. It can import CAD models, grid processing, and editing tools, all of which can be easily converted into usable and efficient robotic models. The findings show that this technique greatly decreases development and delivery timeframes by providing a ready-to-use simulation environment [9] .

Researchers in [10] has been designing and modeling a quadrotor using an easy interface. The software uses SolidWorks application program interface and MATLAB Simscape toolbox for model validation, simulation, and control generation. Developers can use SolidWorks API to create a standalone application that takes advantage of their technical expertise and integrates CAD to improve the simulation's relevance to the real world. The tool creates MATLAB simulation environment files based on the criteria provided after a 3D model has been created and its attributes are defined.

For correct dynamic modeling, determining model parameters utilizing simulation control systems and MATLAB variables is critical [11]. Simulink is a block diagramming environment that may be used to model-driven design and simulate different domains. Embedded system design, simulation, and automatic code generation, as well as continuous testing and verification, are made easier. Simulink is a dynamic system modeling and simulation program that includes a customizable block library, graphic editor, and solution tools. It is linked to MATLAB, allowing for more thorough testing of MATLAB procedures as well as the export of simulation data to MATLAB [12]. And Simscape multibody is a program that creates and solves motion equations for a mechanical system. Complete CAD assemblies, including blocks, splicing, joints, constraints, and 3D geometry can be imported. Simscape Multibody helps with the creation of system-wide performance control and testing systems [11].

SolidWorks and MATLAB/Simulink software are used in [13] to test theories and simulate robot motion by comparing two robots with the same trajectory and time length and generating computer code to identify dynamic and kinetic parameters. Create a CAD model for two purposes in [14], namely, to develop a mathematical model for the

design of a specific controller and to create a model that can be exported to MATLAB/Simulink (Simscape/SimMechanics), where the modeling process was aided by utilizing the blend properties of MATLAB and SolidWorks.

The Simscape Multibody environment was used to create a simplified model of the quadrotor in [15] .The simplified quadrotor model is a nonlinear system that is underacting and tightly linked. The drag effect, the push caused by propeller spinning, and external wind fields are all factored into the dynamics. For feedback control, linear PID controllers are built in the presence of wind fields.

In this paper, the ability of modeling and control for complex engineering systems is studied. The quadrotor system is used as a case study. The modeling process is based on a 3D CAD model using SolidWorks software and SimMechanics/ MATLAB. The real quadrotor system is transferred from the real system into a 3D model on the computer. Also, the control process is considered in this work. Two different approaches have been tested to see the effectiveness of the 3D model-based control. Several simulation results are illustrated for a designed specific path. The aim of the present work is to diagnose the stability condition of quadrotor through using two controllers is adaptive control and PID control. The rest of the paper is organized as follows: in the next section the quadrotor modeling is detailed, after that the control process is presented, later the simulations carried out in this work are demonstrated, and finally, the conclusions of this paper are addressed.
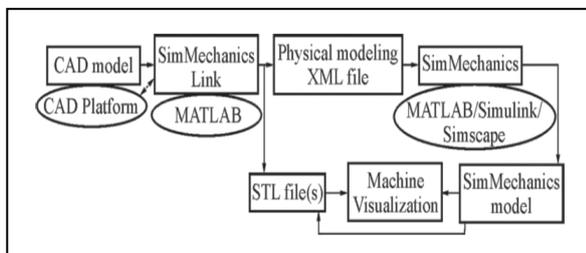

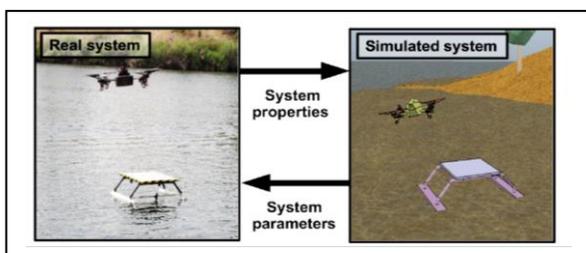Fig. 1. Matlab/Simulink implementation technique for the CAD model [4]

Fig. 2. In the real and simulated environment, the aerial-marine robot system is displayed both [9]

1.1. Quadrotor modelling

The dynamic model of mechanical systems traditionally derived using mathematical relations. The derivation process includes using physical laws of motion (i.e., newton's second law). Depending on the complexity of the system, the results of the derivation process yield normally to a partial differential equation or a set of partial differential equations. These equations can be simplified into a linear equation depending on the nature of the system using linearization methods. In this article, the physical system is a quadrotor system that has a highly nonlinear and coupled mathematical model using the traditional methods of modelling which complicated the modelling process and control design. In this work, the CAD model design is used instead of deriving mathematical equations. The CAD model can be designed using the real dimensions and properties of the system. This modelling method will give nearly exact real system parameters and simplifying the control design process. This type of model will be explained in the next sections and will also be used in controller design and simulations.

1.2. SolidWorks quadrotor CAD model

There are several different types of CAD, such as solid modeling and wireframe modeling, etc. Solid modeling is the process of constructing three-dimensional models of different components and structures using a computer-aided design (CAD) tool. Where solid modeling systems are used to build up expressions or variables and different surfaces for components and it plays a major role in manufacturing industries with separate parts. In many software modeling applications, wireframe meshes are widely used to display 3D artifacts. wireframe modeling is particularly suited for modeling forms such as frame structures similar to wireframe meshes in their overall appearance and wireframe modeling even with beams and columns dominant shapes [16].

Also, there are several different types of CAD software, such as Catia and SolidWorks, etc. Catia is a product analysis program used in the manufacture of automobiles, aircraft, and industrial machinery. Catia is a powerful engineering model designer, The program is ideal for creating, fabricating, and analyzing solids, surfaces, fixtures, and drawings. SolidWorks one of the industry's leading 3D CAD tools, which is used to produce parts and assembly drawings by utilizing parametric features [17]. SolidWorks provides powerful engineering tools for design and can simulate a model for analyzing forces and torque in mechanical links and acceleration and displacement mapping of every part of the system. The goal of SolidWorks is to create 3D models drawing from each part and this includes engineering data. The parts are connected correctly to form an assembly as shown in the figure3, where figure1 shows the 3D model designed using CAD of the SolidWorks system.

1.3. SimMechanics quadrotor model

SimMechanics forms part of the software Simscape/MATLAB simulation, which enables modeling of complex dynamic systems using the

method of programming block diagrams. SimMechanics can also interact with the other ingredients of the Simulink library, thus raising the probability of mechatronic and robotic scheme modeling [18]. SimMechanics provides two options for machine animation and visualization. One is the integrated Handle Graphics tool, which uses regular Handle Graphics equipment known from MATLAB with some special functions that are specific to Sim Mechanics. The machine bodies can be described as convex hulls. A second choice is to represent the bodies as similar ellipses[19].

Simscape Multibody used to export the CAD model from SolidWorks to SimMechanics and considered a multibody simulation environment. Using blocks representing bodies, joints, constraints, and force components, the multibody structure can be modeled. Then, based on the CAD model, SimMechanics formulates and solves the motion equations for the complete mechanical system [12]. CAD tools provide calculations for the center of gravity, as accurate calculations of the center of gravity are important to equalize motion and torque produced by motors.

2. QUADROTOR SIMULATION MODEL

The quadrotor system has three translational and three rotational degrees of freedom. It has also four actuators (rotors) which are modeled in a cross configuration as shown in figure 4. The quadrotor has four rotors and four propellers, where the thrust force is formed by the propellers, where an increase or decrease in the speed of the four propellers in the quadrotor will cause a change in the altitude in the velocity and position. Each two of these motors (M_1, M_2, M_3, M_4) rotate in the opposite direction (e.g., M_1 and M_3 rotate clockwise and vice versa) to stabilize the quadrotor, M_2 and M_4 are rotated in an anticlockwise direction to describe the movement of a 6-DOF rigid body, earth inertial reference, and body-fixed reference [20], as described in see figure 5.

Fig. 3. 3D model of a quadrotor system

The kinematics of 6-DoF quadrotor:

$$[\Gamma^E \quad \theta^E]^T = J_B^E[N^B \quad W^B]^T \qquad (1)$$

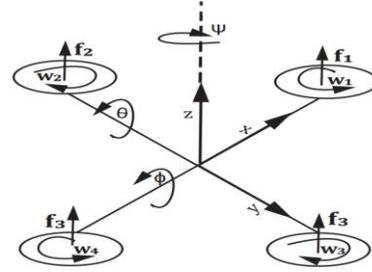Here Γ^E is the linear position, θ^E is Euler angles.
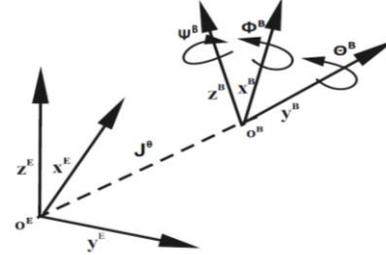

Fig. 4. Quadrotor system schematic

Fig. 5. Body and earth frames

The linear position ($\Gamma^E[m]$) and the Euler angles ($\theta^E[rad]$) of the quadrotor are defined as

$$\Gamma^E = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad \theta^E = \begin{bmatrix} \emptyset \\ \theta \\ \varphi \end{bmatrix} \qquad (2)$$

Where x, y and z are the positions of the center of gravity in E-frame, while are $\emptyset, \theta$ and φ represent the roll, pitch with a yaw in E-frame respectively. A rotation matrix is necessary to map a vector's orientation from B-frame to E-frame [21]. The composition of the rotation matrix (R_θ) that converts the B-frame vector to the E-frame,

$$R_\theta = \begin{bmatrix} C_\varphi C_\theta & -S_\varphi C_\emptyset + C_\varphi S_\theta & S_\varphi S_\emptyset + C_\varphi S_\theta C_\emptyset \\ S_\varphi C_\theta & C_\varphi C_\emptyset + S_\varphi S_\theta S_\emptyset & -C_\varphi S_\emptyset + S_\varphi S_\theta C_\emptyset \\ -S_\theta & C_\theta S_\emptyset & C_\theta C_\emptyset \end{bmatrix}$$
$$(3)$$

$$T_\theta = \begin{bmatrix} 1 & S_\emptyset t_\theta & C_\emptyset t_\theta \\ 0 & C_\emptyset & -S_\emptyset \\ 0 & \dfrac{S_\emptyset}{C_\theta} & \dfrac{C_\emptyset}{C_\theta} \end{bmatrix} \qquad (4)$$

Where T_θ is a matrix of angular transformations, where S_φ refers to $sin(\varphi)$ and C_φ refers to $\cos(\varphi)$ t_θ refers to $\tan(\theta)$.

Some physical characteristics of the quadrotor are calculated in the fixed frame of the earth (roll, pitch ,yaw angles, and angular velocities), although some characteristics are calculated in fixed frames of the body (linear accelerations) [22].

Each control input affects a certain movement, such as u_2 affects the movement of the roll, u_3 affects the movement of the pitch, u_4 affects the movement of the yaw, and u_1 affects the upward movement, where show the effect of the system in following equations [23].

$$u_1 = k_f(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \qquad (5)$$
$$u_2 = k_f(\Omega_4^2 + \Omega_2^2) \qquad (6)$$
$$u_3 = k_f(-\Omega_3^2 + \Omega_1^2) \qquad (7)$$
$$u_4 = k_M(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2) \qquad (8)$$
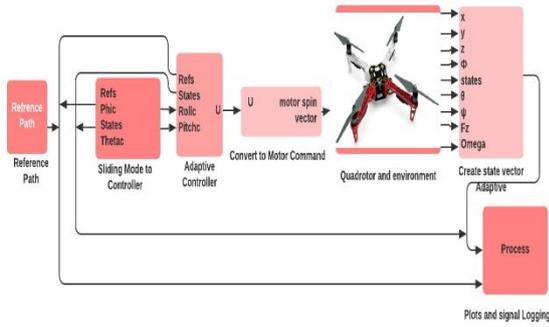
Fig .6. CAD-simulator and controller (sliding mode controller with adaptive controller)
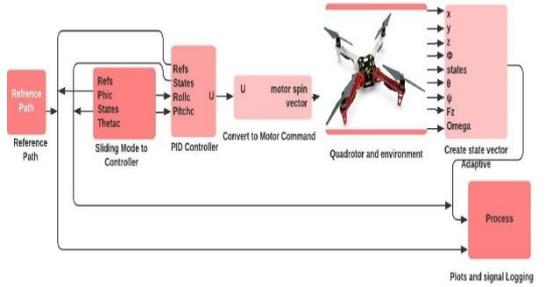

Fig. 7. CAD-simulator and controller (sliding mode controller with PID Controller)

Figure 6 and figure 7 show the model that was used in this work, which consists of the reference path block that produces the path to follow for the quadrotor. The block sliding mode generates the required roll and pitch angle to ensure the required quadrotor position in the $x - y$ plane. The adaptive controller block uses an architecture from the model reference adaptive control to simulate and monitor the quadrotor's ideal dynamics. The classical PID controller shown in figure 5 is implemented in the PID controller's block. The motor control commands block translates the computed control vector into the force application of all propellers by using the adaptive controller block or the PID controller block [24].

3. CONTROL DESIGN

Sliding mode control is a very common strategy for controlling nonlinear, unconfirmed systems and the main disadvantage of controlling sliding mode is the phenomenon of chatter. Adaptive sliding mode regulation Is carried out to it be checked by simulation based on the Lyapunov Principle [25].

$$\dot{\chi} = f(\chi) + g(\chi)u + \Delta \qquad (9)$$

Here: χ is the state vector, u is the system control vector, Δ is the unidentified vector that represents the modeling error. $f(\chi)$ and $g(\chi)$ are believed to be consistently continuously differentiable [25]. Where the vector of the error state is $e = \chi - \chi_r$. The sliding surface's first derivative can be defined as:

$$\dot{s} = \dot{e} = \dot{\chi} - \dot{\chi}_r = f(\chi) + g(\chi)u + \Delta - \dot{\chi}_r \quad (10)$$

In order to let the $\dot{s} = 0$, let us propose the following control input,

$$u = g^{-1}(\chi)(-ks - f(\chi) - \hat{\Delta}) \qquad (11)$$

Where k is a positive definite diagonal matrix. Substitute the proposed control input vector with the following shape for the first derivative of the sliding surface.

$$\dot{s} = -ks + \Delta - \dot{\chi}_r - \hat{\Delta} = -ks + \tilde{\Delta} \quad (12)$$

Where $\hat{\Delta}$ is an online estimated vector, assume $\tilde{\Delta} = \Delta - \dot{\chi}_r - \hat{\Delta}$, then construct the adaptive dynamic vector as follows:

$$\dot{\tilde{\Delta}} = \dot{\Delta} - \ddot{\chi}_r - \dot{\hat{\Delta}} = -\lambda s \qquad (13)$$

Where λ is the definite positive diagonal gain matrix. Lyapunov candidate function is defined as

$$V = \frac{1}{2} s^T s + \frac{1}{2} \tilde{\Delta}^T \lambda^{-1} \tilde{\Delta}. \qquad (14)$$

The first derivative of the Lyapunov candidate function is defined as:

$$\dot{V} = s^T \dot{s} + \tilde{\Delta}^T \lambda^{-1} \dot{\tilde{\Delta}} \qquad (15)$$

By expanding the preceding equation, we obtain:

$$\dot{V} = s^T \dot{s} + \tilde{\Delta}^T \lambda^{-1} \dot{\tilde{\Delta}} = s^T(-ks + \tilde{\Delta}) + \qquad (16)$$
$$\tilde{\Delta}^T \lambda^{-1}(-\lambda s) = -s^T ks < 0.$$

So it can be stated that the sliding surface converges to zero as time increases, yield the error states goes to zero [25]. The relationship between attitude angle and position derivative is defined as

$$\varphi_c = (\ddot{x}\sin \psi - \ddot{y}\cos \psi)m/F_z$$
$$\theta_c = (\ddot{x}\cos \psi + \ddot{y}\sin \psi)m/F_z \qquad (17)$$
$$F_{zc} = (\ddot{Z} + g)m + \hat{\Delta}_1)$$

Where the predicted $\hat{\Delta}_1$ is online estimated to adapt the simplified model inversion deficiency according to formula (13). The second derivative of variable z can be treated as the first derivative of variable $\dot{z}$ in equations (17). The sliding surface:

$$s_1 = \ddot{Z} \qquad (18)$$

The adaptive sliding mode control is

$$\hat{\Delta}_1 = \lambda_1 s_1 \qquad (19)$$

Trajectory control is considered to drive all three waypoint error components exponentially to stabilize. We would then like to order the acceleration to satisfy the following:

$$\ddot{\chi} = -2\xi\omega\dot{\chi} - \omega^2(\chi - \chi_c)$$
$$= -k_u\dot{\chi}$$
$$- k_u k_x(\chi - \chi_c)$$
$$\ddot{\gamma} = -2\xi\omega\dot{\gamma} - \omega^2(\gamma - \gamma_c)$$
$$= -k_v\dot{\gamma} \qquad (20)$$
$$- k_v k_\gamma(\gamma - \gamma_c)$$
$$\ddot{Z} = -2\xi\omega\dot{Z} - \omega^2(Z - Z_c)$$
$$= -k_w\dot{Z}$$
$$- k_w k_Z(Z - Z_c)$$

Where is ξ the damping factor of the system, ω is the natural frequency of the system. From the above formula (17), (19), and (2). The slow-loop position control provides the attitude set points for the fast loop attitude controller. The control factor can be defined

$$k_x = k_y = k_z = \frac{\omega}{2\xi}, k_u = k_v = k_w = \qquad (21)$$
$$2\omega\xi$$

4. SIMULATION RESULTS

In this section, Thanks to the SimMechanics environment, the 3D quadrotor model is completely simulated as a real one, and the model has been validated using Simscape tools by creating a control algorithm in terms of stability and the ability of the quadcopter to fly at a certain altitude. Where a helix path was created in the CAD model, in addition comparison was made in the case of the adaptive controller and PID controller for paths. Two distinct control approaches are employed in this work to stabilize the quadrotor system's motion. First sliding mode controller with the adaptive controller. The second is the sliding mode controller with the PID controller. The parameter values are listed in table 1.

Table 1. Parameters of CAD Model

Parameter	Units	Value
t	second	140
m	kg	1
d	N.m. s^2	2.5×10^{-7}
b	m	8.1×10^{-6}
I_{xx}	N.m. s^2	0.005161
I_{yy}	N.m. s^2	0.0051571
I_{zz}	N.m. s^2	0.01005
g	m/s^2	9.81
H	m	2.5
L	m	0.1813
J_P	N.m. s^2	2.03×10^{-5}

The helix path generated in Simulink is presented using equation which expresses the path $x = rcos\theta, y = rsin\theta, z = \frac{\theta}{4}, yaw = 0$. The simulation results are display in figure (8–20). Were, can be seen in figure 8 pictures showing the movement of the path for the quadrotor at a different time.
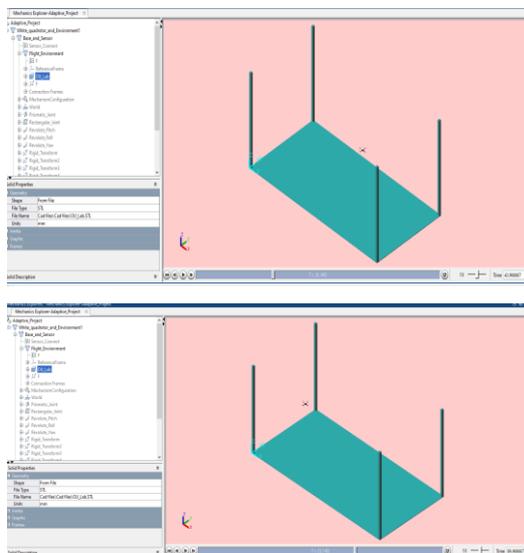

Fig. 8. 3D image to conical path of quadrotor at (43-89) sec

Figure 9 shows the trajectory of the conical helix, and it was observed that the quadrotor drive is able to follow the required flight path correctly. Figure 10, shows the change in quadrotor position values with acceptable desired response time. Figure 11, also demonstrates excellent tracking of the attitude reference trajectory, and it can be noted that the yaw response is different from the pitch and roll response, where the yaw response approaches zero.

Furthermore, figures 10 and 11 illustrate a comparison of tracking results in the position and attitude subsystems produced with the adaptive sliding mode controller with the adaptive controller. Figure 12 be seen in that the force start from zero then increases and is fixed at 100s. Figure 13 shows the angular velocity where the four propellers angular velocity is required to balance the acceleration due to gravity, allowing the quad to hover instantaneously.
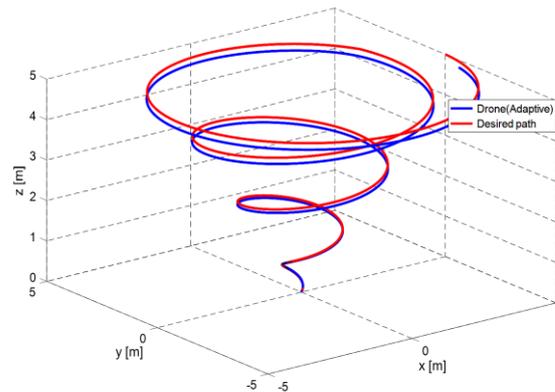

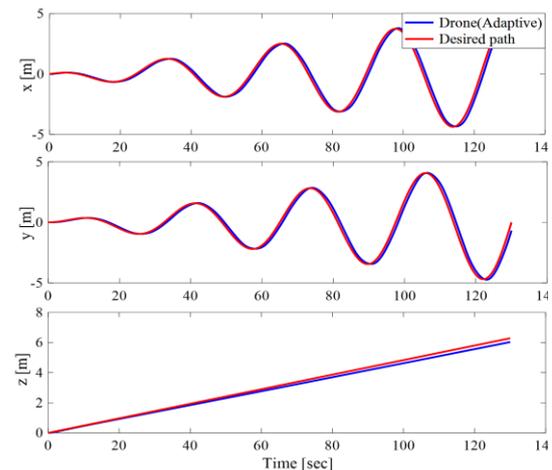Fig .9. 3D Conical helix trajectory with the adaptive controller

Fig. 10. Quadrotor position with adaptive controller

Figure 14 shows the trajectory of the quadrotor by used the adaptive sliding mode controller with the PID controller.

Quadrotor position as shown in figure 15 for the conical helix trajectory. It can be observed from figure 14 and figure 15 that the quadrotor is capable of properly monitor the appropriate fly path. Figure 16 demonstrates good tracking of the conical helix trajectory's attitude. It can be noted also that the yaw

response is different from the pitch and roll response, where the yaw response approaching zero.
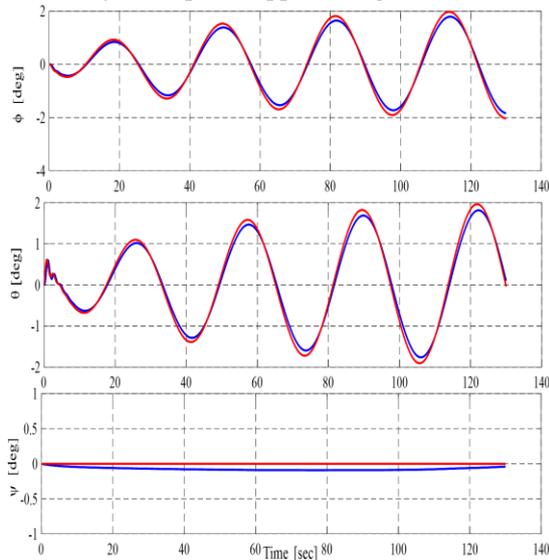

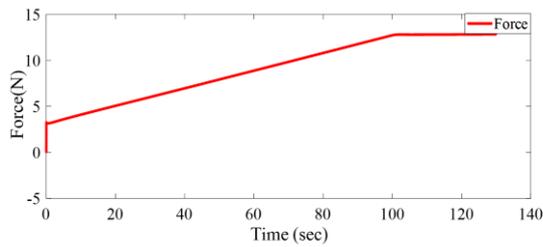Fig. 11. Attitude angles of the quadrotor with the adaptive controller

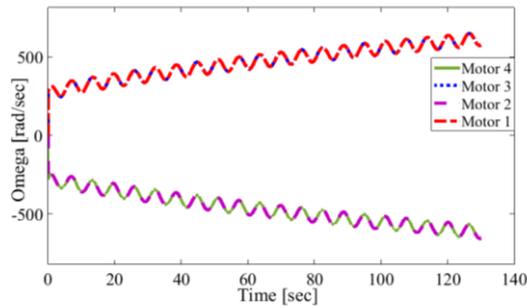Fig. 12. Force with the adaptive controller.

Fig. 13. Shows the angular velocity with the adaptive controller

Fig. 14. Trajectory of the quadrotor with the PID

Fig. 15. Quadrotor position with PID controller

Fig. 16. Attitude angles of the quadrotor (ϕ, θ, Ψ) with PID controller.

Fig. 17. The angular velocity with PID controller

Figure 15 and figure 16 provide a comparison between the tracking results in the position and attitude obtained by using the adaptive sliding mode controller with a PID controller and the desired path. figure 17 expresses also angular velocity but with the PID controller, where it can note in figure 17 motor 4 and motor 2 rotate contrariwise to motor 3 and motor 1.

For comparison, error can be seen in figures 18-20, when the adaptive controller and PID are using. it can be seen that PID controller outperforms the adaptive controller.
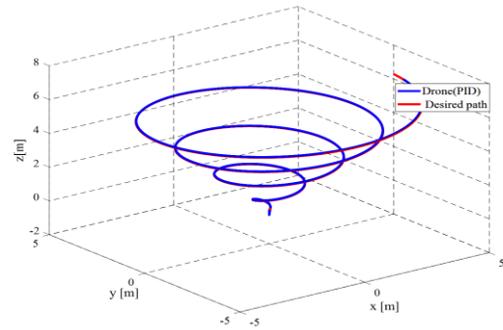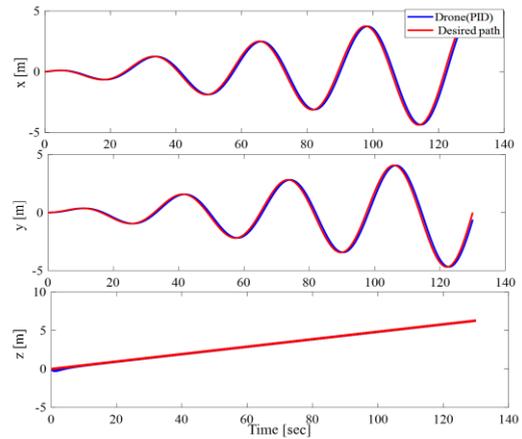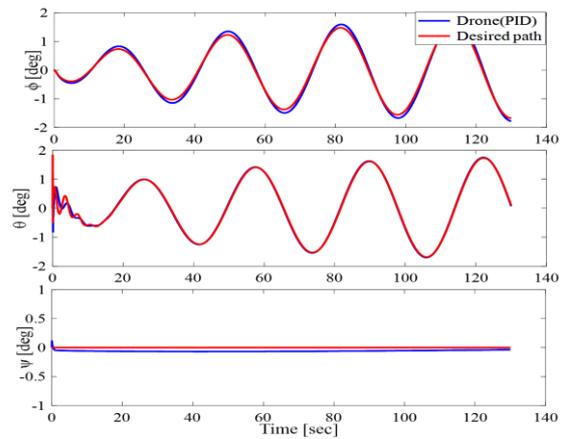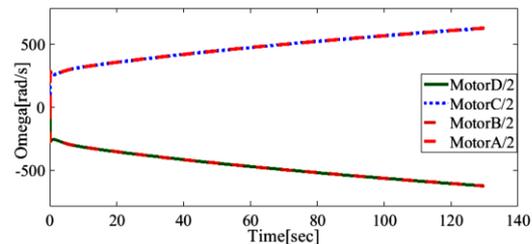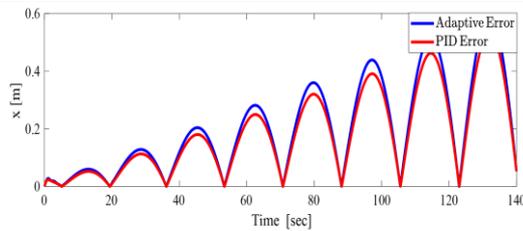
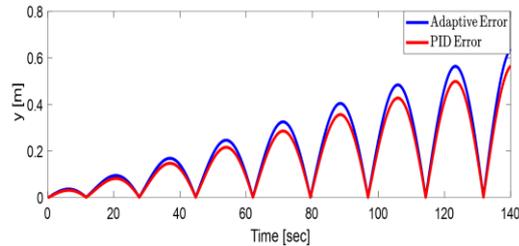Fig.18. Error position x between adaptive and PID controller

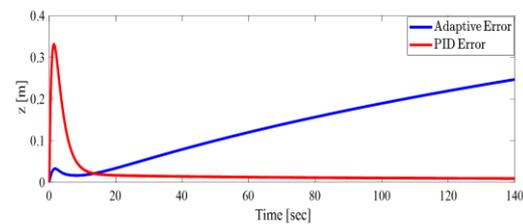Fig. 19. Error position y between adaptive and PID controller

Fig. 20. Error position z between adaptive and PID

5. CONCLUSION

In this paper, The CAD model for the quadrotor system demonstrates the ability to model and control the 3D model. Also, it can see the architecture of the CAD model is used instead of deriving mathematical equations. The modeling method is based on a model of 3D CAD using tools from SolidWorks and SimMechanics/MATLAB. To see the efficacy of 3D model-based control, a sliding mode controller is used and two separate methods were tested. First by using the adaptive sliding mode controller with the adaptive controller, second by using the adaptive sliding mode controller with the PID controller. where the benefit of sliding mode control is that it is not subject to model errors, parametric uncertainties, and other disturbances. The CAD model is improved in this paper through the use of a single trajectory in the MATLAB/Simulink program and it can be seen from the simulation result that it is obtained a position and attitude perfectly where the error rate is reduced between the adaptive sliding mode controller and the desired path, and it was obtained comparison of position and attitude results obtained using the adaptive sliding mode controller and the desired path. It can be seen from the result that convergence occurs between the adaptive sliding mode controller and the desired path and the yaw angle is close to zero.

Author contributions: *research concept and design, F.A.R., M.T.H.; Collection and/or assembly of data, F.A.R., M.T.H.; Data analysis and interpretation, F.A.R.,* *M.T.H.; Writing the article, F.A.R., M.T.H.; Critical revision of the article, F.A.R., M.T.H.; Final approval of the article, F.A.R., M.T.H.*

Declaration of competing interest: *The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.*

REFERENCES

1. Elruby A, El-khatib M, El-Amary N, Hashad A. Dynamic modeling and control of quadrotor vehicle. Int. Conf. Appl. Mech. Mech. Eng. 2012;15:1–10. https://doi.org/10.21608/amme.2012.37092 .
2. Abbasi E, Mahjoob M. Controlling of quadrotor UAV using a fuzzy system for tuning the PID gains in hovering mode. 10th Int. Conf. Adv. Comput. Entertain. Technol. 2013:1–6. http://conf-scoop.org/ACE-2013/13_Reza2_ACE.pdf.
3. Jatsun S, Lushnikov B, Emelyanova O, Leon ASM. Synthesis of simmechanics model of quadcopter using solidworks cad translator function. Smart Innov. Syst. Technol. 2021;187:125–137. https://doi.org/10.1007/978-981-15-5580-0_10.
4. Cekus D, Posiadała B, Warys P. Integration of modeling in solidworks and matlab/simulink environments. Arch. Mech. Eng. 2014;61(1):57–74. https://doi.org/10.2478/meceng-2014-0003.
5. Gordon RF, Kumar P, Ruff R. Simulating quadrotor dynamics using imported CAD data. AIAA Model. Simul. Technol. Conf. 2013:1–7. https://doi.org/10.2514/6.2013-4735.
6. Grau A, Bolea Y, Sanfeliu A. CAD-based Approach for Identification of UAVs. MATEC Web Conf. 2019;291(9):1004. https://doi.org/10.1051/matecconf/201929101004.
7. Elsamanty M, Khalifa A, Fanni M, Ramadan A, Abo-Ismail A. Methodology for identifying quadrotor parameters, attitude estimation and control. 2013 IEEE/ASME Int. Conf. Adv. Intell. Mechatronics Mechatronics Hum. Wellbeing. AIM. 2013:1343–1348. https://doi.org/10.1109/AIM.2013.6584281.
8. Makkonen T, Nevala K, Heikkilä R. A 3D model based control of an excavator. Autom. Constr. 2006; 15(5):571–577. https://doi.org/10.1016/j.autcon.2005.07.009.
9. Velasco O, Valente J, Alhama Blanco PJ, Abderrahim M. An open simulation strategy for rapid control design in aerial and maritime drone teams: A comprehensive tutorial. Drones. 2020;4(3):1–20. https://doi.org/10.3390/drones4030037.
10. Shaqura M, Shamma JS. An automated quadcopter cad based design and modeling platform using solidworks API and smart dynamic assembly. ICINCO 2017 - Proc. 14th Int. Conf. Informatics Control. Autom. Robot. 2017;2:122–131. https://doi.org/10.5220/0006438601220131.
11. Simscape Multibody - MATLAB & Simulink. https://www.mathworks.com/products/simscape-multibody.html.
12. Simulink Documentation. https://www.mathworks.com/help/simulink.
13. Gouasmi M, Ouali M, Fernini B, Meghatria M. Kinematic modelling and simulation of a 2-R robot using solidworks and verification by matlab/Simulink. Int. J. Adv. Robot. Syst. 2012;9:1–13. https://doi.org/10.5772/50203.

14. Zátopek J, Urednícek Z, Machado J, Sousa J. Dynamic simulation of the CAD model in SimMechanics with multiple uses. Turkish J. Electr. Eng. Comput. Sci. 2018;26(3):1278–1290. https://doi.org/10.3906/elk-1712-217.

15. De Simone MC, Russo S, Rivera ZB, Guida D. Multibody model of a UAV in Presence of wind fields. Proc. - 2017 Int. Conf. Control. Artif. Intell. Robot. Optim. ICCAIRO 2017. 2017:83–88. https://doi.org/10.1109/ICCAIRO.2017.26.

16. Srinivasan V, Mandal E, Akleman E. Solidifying wireframes creating the solid wireframe. Renaissance Banff: Mathematics, Music, Art, Culture. 2005: 203-210.

17. Mariappan SM, Veerabathiran A., Modelling and simulation of multi spindle drilling redundant SCARA robot using SolidWorks and MATLAB/ SimMechanics. Rev. Fac. Ing. 2016;81:63–72. https://doi.org/10.17533/udea.redin.n81a06.

18. Yun C, Li XM. Aerodynamic model analysis and flight simulation research of UAV based on Simulink. J. Softw. Eng. Appl. 2013;6(2):43–47. https://doi.org/10.4236/jsea.2013.62007.

19. Schlotter M. Multibody system simulation with SimMechanics. Analysis. 2003:1–23.

20. Emran BJ. Nonlinear Adaptive control of a Quadcopter. 2914:106.

21. Benic Z, Piljek P, Kotarski D. Mathematical modelling of unmanned aerial vehicles with four rotors. Interdiscip. Descr. Complex Syst. 2016;14(1):88–100. https://doi.org/10.7906/indecs.14.1.9.

22. Islam M, Okasha M, Idres MM. Dynamics and control of quadcopter using linear model predictive control approach. IOP Conf. Ser. Mater. Sci. Eng.. 2017;270(1):0–9. https://doi.org/10.1088/1757-899X/270/1/012007.

23. Moshayedi AJ, Gheibollahi M, Liefa L. The quadrotor dynamic modeling and study of meta-heuristic algorithms performance on optimization of PID controller index to control angles and tracking the route. IAES International Journal of Robotics and Automation 9.4. 2020:256.

24. Lee S, Ha C, Kim BS. Adaptive nonlinear control system design for helicopter robust command augmentation. Aerosp. Sci. Technol. 2005;9(3):241–251. https://doi.org/10.1016/j.ast.2004.12.007.

25. Wang L, He C, Zhu P. Adaptive sliding mode control for quadrotor aerial robot with I type configuration. Int. J. Autom. Control Eng. 2014;3(1):20. https://doi.org/10.14355/ijace.2014.0301.03.

Mustafa T. HUSSEIN Basim was born in Diwaniyah, Iraq in 1983. He received the B.Sc. degree in Mechanical engineering from the University of Babylon, in 2005, an M.Sc. in mechanical engineering in 2008 from the University of Baghdad, and the Ph.D. degree in mechanical engineering (mechatronics) in 2014.

He was an assistant professor in applied mechanics from 5/9/2018 till now at the University of Babylon, college of engineering, Mechanical engineering department.

Fatimah A. RAHEEMAH was born in Baghdad, Iraq in 1996. She received the B.Sc. degree in Mechanical engineering from the University of Babylon, in 2018 and M.Sc. student from 15-9-2019 till now at the University of Babylon, College of Engineering, Department of Mechanical Engineering.