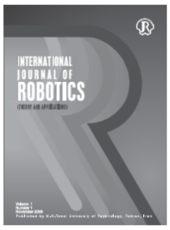


INTERNATIONAL JOURNAL OF ROBOTICS
(THEORY AND APPLICATIONS)
E-mail: IJR@kntu.ac.ir

Fault Tolerant System for Multirotor Drones: a Novel Comparison for Different Methods

H. Mazeh [a], H. D. Taghirad [a*] and J. Sahili[b,]

[a] Advanced Robotics and Automated Systems, Faculty of Electrical Engineering, K. N. Toosi University of Technology, Tehran, Iran.
[b] Mechanical Engineering Department, Faculty of Engineering- Branch 3, Lebanese University, Beirut, Lebanon.

ARTICLE INFO

Article history:
Submit: 2022-02-08
Revise: 2023-09-09
Accept: 2023-09-10

Keywords:
Unmanned Aerial Vehicle
Fault Tolerance Controller
Fault Detection Algorithm
Super Twisting Algorithm
Pseudo-inverse Control Allocation

ABSTRACT

This paper proposes a nonlinear robust passive fault tolerance controller for recovering faults and perturbation that affect the actuators of multirotor unmanned aerial vehicles. This approach is applied to a coaxial octorotor drone, benefiting from its actuator redundancy. The proposed controller is based on a second order super twisting sliding mode controller, which attenuates the chattering effect caused by first order sliding manifold. An active fault tolerance approach is also proposed based on both offline and online strategies for tolerating total effectiveness loss of actuators for an octorotor. A nonlinear Thau observer is designed firstly to detect actuator fault. Then two different control recovery algorithms are designed to compensate the fault, whenever it is detected to maintain the stability and desired behavior of the drone. The proposed algorithms are simulated and tested under fault free conditions and several fault conditions with various fault scenarios affecting the actuators through a complex 3D trajectory maneuver performed by the UAV. A new case study is presented to compare the behavior of the octorotor in case of successive total actuators loss. A novel comparison criterion for comparing various methods of fault tolerance controllers is introduced considering the design simplicity, implementation complexity, and system performance. The obtained results present suitable tracking performance for the desired trajectory, despite of different injected faults, with desirable recovery time. In addition, a weighting table is constructed to show the strength of each method.

[*] Corresponding address: K. N. Toosi University of Technology, Tehran, Iran,
Tel.: +98 21 8406 2321; Fax: +98 21 8846 2066.
E-mail address: taghirad@kntu.ac.ir.

1. Introduction

Rapid technology development in the electronic and power domains has enhanced the area of implementing unmanned aerial vehicles (UAVs). Due to its medium and low-cost availability with acceptable performance, low profile sensors and powerful battery systems, UAV systems have expanded its potential applications which benefit from these flying robots. Not only in governmental and large research program applications, nowadays UAVs are being easily available for surveillance, hobbies and sport coverage, commercial and delivery activities, small research programs, environmental studies and protection, and even more and more applications that make use of UAVs with different sizes and costs. A survey on civil applications for UAV systems was presented in [1].

Multirotor UAVs are vehicles that have been more attractive than fixed wing vehicles due to their low profile, low cost, simpler control, and hovering ability. These kinds of flying robots are formed from a combination of high spinning motor/propeller system distributed along the extremities arms of its frame. The position of the actuators in multirotor vehicles made it susceptible and sensitive to crashes due to its collision with surrounding. Some crashes may lead to total blockage of the motor or a partial loss in its efficiency. Actuators faults affect the stability of multirotor UAVs that may lead to total loss in controllability of its states in some non-redundant configurations and non-equipped flight controllers with fault-tolerant control (FTC) algorithms.

The necessary requirements for reliability, robustness, and desirable system performance have motivated research in the area of FTC benefiting from the enhancement in control theory, processing technology, and sensing technology. A comprehensive work has been made to swift the stability of non-redundant actuator configuration in multirotor as in quadrotors UAVs in case of partial loss of actuators' effectiveness [2] or assuming new control strategies to complete its tasks in case of total loss [3]. As an alternative solution for redundancy gap in quadrotor UAVs, implementing vehicles with redundant actuators configuration equipped with FTC algorithms as hexarotors and octorotors was proposed in [4] and [5] respectively.

Through the existing fault tolerance controllers, two categories can be defined for sorting these approaches; active and passive strategies. Active fault tolerance controller (AFTC) is a strategy which is based first on a fault detecting and diagnosis (FDD) scheme and then on reconfiguring the control system to recover its faulty behavior. Passive fault tolerance controller (PFTC) relies on robust controller that deals with faults as external perturbations that must be compensated for driving the system to a desired behavior. The main drawback accompanied while implementing AFTC approach in robotic autonomous systems is in the case of any improper decisions in FDD which can drive the recovery system to a lack of effectiveness, in contrast PFTCS approach is simpler and it does not depend on an FDD scheme, thus there is no delay between the fault and the recovery actions. On the other side, AFTC strategies are less conservative than PFTC due to the fact that in active strategy the fault is considered as a listing variable and not an additive uncertainty in which robust controller must counterattack its effect. Additionally, PFTC is designed for both non-faulty and faulty cases, thus this approach cannot be optimized for all problems solution. A comparative review between these two approaches was presented in [6] with numerical and experimental justification.

Several researches and works in the domain of UAVs' FTC were based on passive architecture. In the design of these controllers commonly sliding mode control (SMC), $\mathcal{H}_\infty$ control techniques, and back-stepping controller are reported in [7], [8], and [9], respectively. The author in [10] proposed an adaptive SMC (ASMC) scheme to accommodate system uncertainties caused by actuator faults in a quadrotor. The system behavior was examined through simulation when only a partial loss to a motor has occurred during a simple hovering track. A partial effectiveness loss fault was examined in [11] on a quadrotor UAV by applying sliding mode theory during a complex trajectory. A sliding mode-based fault tolerance controller was implemented in [12] on a symmetric octorotor simulation examining partial and total rotors loss through a simple linear track. Despite good tolerating abilities of sliding mode control, this kind of controllers suffers from chattering caused by the discrete part in its control input. To eliminate the chattering effect caused by discontinuity in sliding mode controller, [13] made use of higher order controller called super twisting algorithm (STA) to design a fault tolerant system for a coaxial octorotor. It was examined through a simple track with only two total motor failure described by turning it off successively during flight. In real cases, the UAV may be subjected to complex 3D maneuvers, thus this algorithm was not examined through those kinds of tracks.

AFTC algorithms are categorized into two control groups: online and offline strategies. The design of an offline AFTC approach is performed by defining a convenient cost function subjected to specific physical system constraints under different fault potentials. An optimization algorithm is proposed to minimize the defined cost function in an off-board manner, then the solution parameters are listed as lookup tables in the flight controller program which are selected according to the detected fault type and position. The author in [14] submitted a control allocation problem to formulate an online FTC strategy for an actuator-redundant multirotor UAV subjected to total actuators fault. An explicit solution for the optimization problem was proposed based on parametric programming technique, and the solution parameter gains were stored in a lookup table. Therefore, it demanded very low computational cost and provided a straightforward optimal solution to the input commands extracted by the flight controller. The proposed work was utilized to a hexarotor UAV but after mutating its conventional rotor distribution for obtaining system stability, in addition not all faults could be tolerated due to some controllability limitations. Distributing the control efforts among nonfaulty actuators while flight control algorithm is derived, is the main key feature of the online AFTC strategy. This technique is based on an onboard solver to a formulated optimization problem to maintain the stability of the control system in a desirable performance. An online AFTC algorithm relying on control allocation and reconfiguration in case of total loss of actuator(s) was presented in [15], however a modification to the UAV's structure was necessary through two extra actuators to guarantee stability when total faults occur on the actuators. This approach was also studied before in [16].

An actuator redundant octorotor UAV is considered in this paper to overcome the noncontrollable issues in case of total actuators faults. To overcome the chattering drawbacks accompanied with first order SMC, a second order SMC is proposed as a PFTC strategy for a coaxial octorotor UAV. The gains of this controller shall be designed in a manner to compensate various probable actuators' fault: partial effectiveness loss, total effectiveness loss, and random disturbance noise during a complex maneuver. This work aims to examine the performance of the controller approaching to real cases by desiring a complex 3D trajectory and tuning the gains of the proposed PFTC precisely. In addition, AFTC strategy is investigated through both online and offline approaches. A nonlinear fault detection algorithm is first designed based on Thau observer for detecting the occurrence of actuator fault and specifying its location with respect to the UAV's frame. Then the offline control distribution algorithm and online control allocation reallocation algorithm are both designed. A novel physical comparison is performed between PFTC and AFTC approaches regarding the system recovery time, position deviation when fault exists, and control efforts performed by the UAV about the roll, pitch, and yaw rotation axes, for an octorotor UAV performing a 3D complex helical maneuver and subjected to four consecutive total actuator loss. A new case study is presented to compare the behavior of the octorotor in case of successive total actuators loss. A novel comparison criterion for comparing various methods of fault tolerance controllers is introduced considering the design simplicity, implementation complexity, and system performance. A weighting table is constructed to show the strength of each FTC method.

This paper is organized as follows. Section 2 states the derivation of the dynamic model of a coaxial octorotor using Newton-Euler formulation. A general definition for sliding mode controller is shown in section 3. This section describes the proposed second order SMC based on super twisting algorithm applied to a coaxial UAV for controlling attitude angles and altitude state and projects the experimental results applying complex 3D trajectory in fault free scenario and with different predicted actuators faults. In section 4, the design of AFTC is performed by proposing a nonlinear Thau observer for fault detection then system recovery algorithms are designed. The performance of the detection and tolerance combined blocks are both examined through simulation. Section 5 is dedicated for result analysis and physical comparison between AFTC and PFTC proposed algorithms. Finally, concluding remarks and future prospect are given in section 6.

2. Octorotor Modelling

A coaxial octorotor UAV is an actuator-redundant multirotor system with same number of arms as in quadrotors but with double sided rotors distributed at its end arms. The coaxial geometry of the vehicle permits it to carry more payload compared to a conventional quadrotor, with less profile and arms' length, thus less control efforts and power consumption compared to a symmetric octorotor. The equations of motion may be described in both inertial and body reference frame. The geometry of a coaxial UAV is shown in Fig. 1 with its distributed motors and propellers in both body and inertial frames. Transformation between inertial and body frame is given by simple successive rotation Euler matrices. The combination of eight rotors aims to stabilize the vehicle in three directions roll, pitch and yaw. Furthermore, it generates convenient total thrust to attain desired hovering state. By this means, UAV is a six degree of freedom robot controlled through four virtual inputs: three are the moments across roll, pitch, and yaw directions and the fourth is the total thrust.

The dynamics of an octorotor UAV is derived using Newtons Euler formulation as reported in [17]. Various assumptions are considered while deriving the nonlinear model of the octorotor due to its geometry and low motors' inertia: neglecting friction forces and gyroscopic effects of the propellers, constant drag and thrust coefficients, and symmetricity in the vehicle's geometry with diagonal inertia matrix. Let, $\boldsymbol{F}$ denotes the vector of forces acting on the octorotor in the body frame, $\boldsymbol{V}$ denotes the vector of linear velocities along x, y and z axes in the body frame, $\boldsymbol{\omega}$ denotes the vector of angular velocities p, q and r in the body frame, and $\boldsymbol{\tau}$ denotes the vector of moments acting on the octorotor in the body frame. The Newton-Euler formulation for the robot is then stated in body coordinate frame as:

$$\begin{bmatrix} m\boldsymbol{I}_{3\text{x}3} & 0 \\ 0 & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \dot{\boldsymbol{V}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\omega} \text{ x } m\boldsymbol{V} \\ \boldsymbol{\omega} \text{ x } I\boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{\tau} \end{bmatrix} \qquad (1)$$

where m is the vehicle's mass. I_{xx}, I_{yy}, I_{zz}, are the moments of inertia of the UAV respectively along the x, y, z axis such that $\boldsymbol{I} = diag(I_{xx}, I_{yy}, I_{zz})$,

Moreover, by considering small angles, the system can be further simplified as:

$$\dot{\boldsymbol{X}} = f(\boldsymbol{X}) + g(\boldsymbol{X})\boldsymbol{U} \tag{2}$$

with $\boldsymbol{X}$ is the state vector formed from Euler angles, vehicle's position, and their derivatives, and $\boldsymbol{U}$ is the input vectors formed from total thrust and moments in three directions. φ, θ and ψ are the roll, pitch, and yaw Euler angles.

$$\begin{gathered}\boldsymbol{X} = [\, x\ \dot{x}\ \ y\ \dot{y}\ z\ \dot{z}\ \ \phi\ \dot{\phi}\ \ \theta\ \dot{\theta}\ \psi\ \dot{\psi}\,]^T \\ \boldsymbol{U} = [\, u_f\ \tau_\phi\ \tau_\theta\ \tau_\psi\,]^T\end{gathered} \tag{3}$$

By applying appropriate transformations, the parameters of the dynamic system are stated in (4).

$$\boldsymbol{f} = \begin{bmatrix} \dot{x} \\ 0 \\ \dot{y} \\ 0 \\ \dot{z} \\ -g \\ \dot{\phi} \\ \frac{I_{yy} - I_{zz}}{I_{xx}}\dot{\theta}\dot{\psi} \\ \dot{\theta} \\ \frac{I_{zz} - I_{xx}}{I_{yy}}\dot{\phi}\dot{\psi} \\ \dot{\psi} \\ \frac{I_{xx} - I_{yy}}{I_{zz}}\dot{\phi}\dot{\theta} \end{bmatrix}, \ \boldsymbol{g} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi)}{m} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi)}{m} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \frac{(\cos\phi\cos\theta)}{m} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{1}{I_{xx}} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix} \tag{4}$$

The relation between virtual input $\boldsymbol{U}$ and the real input to the actuators in which $\boldsymbol{v}$ is the vector of square of speed for each motor is given by the effectiveness input matrix $\boldsymbol{B}$:

$$\boldsymbol{U}(t) = \boldsymbol{Bv}(t) \tag{5}$$

such that:

$$\boldsymbol{B} = \begin{bmatrix} K_f & K_f & K_f & K_f & K_f & K_f & K_f & K_f \\ -\frac{K_f l\sqrt{2}}{2} & -\frac{K_f l\sqrt{2}}{2} & -\frac{K_f l\sqrt{2}}{2} & -\frac{K_f l\sqrt{2}}{2} & \frac{K_f l\sqrt{2}}{2} & \frac{K_f l\sqrt{2}}{2} & \frac{K_f l\sqrt{2}}{2} & \frac{K_f l\sqrt{2}}{2} \\ -\frac{K_f l\sqrt{2}}{2} & -\frac{K_f l\sqrt{2}}{2} & \frac{K_f l\sqrt{2}}{2} & \frac{K_f l\sqrt{2}}{2} & \frac{K_f l\sqrt{2}}{2} & \frac{K_f l\sqrt{2}}{2} & -\frac{K_f l\sqrt{2}}{2} & -\frac{K_f l\sqrt{2}}{2} \\ K_t & -K_t & K_t & -K_t & K_t & -K_t & K_t & -K_t \end{bmatrix} \tag{6}$$

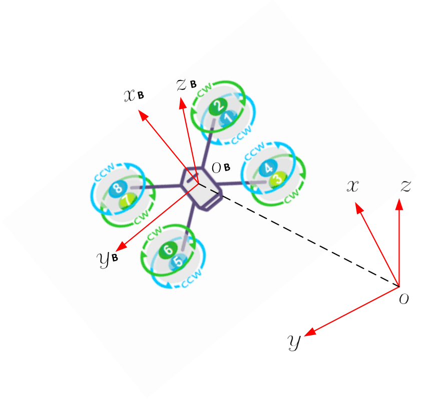


Fig. 1: Coaxial octorotor reference frames.

in which l is the arm's length, K_f and K_t are the thrust and drag coefficients respectively, and:

$$\boldsymbol{v} = [\omega_1{}^2\ \omega_2{}^2\ \omega_3{}^2\ \omega_4{}^2\ \omega_5{}^2\ \omega_6{}^2\ \omega_7{}^2\ \omega_8{}^2]^T \tag{7}$$

3. PFTC Robust Control Design

Sliding Mode Controller (SMC) is a nonlinear control strategy that modifies the nonlinear dynamic characteristics of the system by stratifying a discrete control input which drives the system states towards a specific surface in the state space, defined by sliding surface. At the instance of reaching the sliding surface, SMC conserves the system in the states close to the sliding surface as shown in Fig. 2. Designing an SMC control law is performed by two stages: selecting the appropriate sliding surface for reaching the desired dynamics, then designing the discrete control law that imposes the system states to reach the sliding surface and track it in spite of external disturbances. For selecting the sliding surface, a continuous function $s(x,t)$ is supposed in a way the system track offers eligible attitude when it is zero. This function is denoted by switching variable. When the sliding surface is defined, the next step is to select the layout of the control strategy to stabilize the switching function about the center in finite duration. The manner of the closed loop system may be divided into two phases: reaching phase where the controller attracts the system states towards the sliding surface at a time less than t_s and tries to reach it, and sliding phase where the time is greater than t_s and the system states are kept in the neighborhood of the sliding surface thus the system is tolerated to disturbances and perturbations.

3.1. STA controller:

SMC algorithm generates a discontinuous control input to the system which may create undesirable chattering at high frequency. This aspect may affect the system performance and may even become a serious issue for the moving parts. One approach to rectify the chattering effect in SMC is to implement higher order SMC. This approach can attenuate the chattering effect but not eliminate it. Another approach is to smooth the discontinuous function by an accurate approximation function. In this paper the two approaches are considered and a second order SMC algorithm is proposed implementing STA controller with suitable smoothing function.

Super twisting algorithm (STA) is a higher order SMC of second order. Suppose the nonlinear system states are defined as:

$$\dot{\boldsymbol{x}} = f(x,t) + g(x,t)\boldsymbol{u}(t) \tag{8}$$

such that $\boldsymbol{x}$ is the state vector for the system, $\boldsymbol{u}$ is the system's input. The continuous function defining the system states are f and g. As stated in [18], the sliding variable is posed by the definition s that has the following derivative:

$$\dot{s}(s,t) = \Gamma(s,t) + \Lambda(s,t)\boldsymbol{u}(t) \tag{9}$$

where $\Gamma(s,t)$ and $\Lambda(s,t)$ are unknown bounded functions. The control goal is to realize the convergence to the sliding phase stated as $s = 0$. Consider that a positive constants S_0, c_{min}, c_{max}, C_0, and U_{max} exist such that $\forall \boldsymbol{x} \in R^n$ and $|s(x,t)| < S_0$, the system satisfies the following conditions:

$$\begin{cases} |\boldsymbol{u}(t)| < U_{max} \\ 0 < c_{min} \leq |\Lambda(s,t)| \leq c_{max} \\ |\Gamma(s,t)| < C_0 \end{cases} \tag{10}$$

STA is defined by the controller $\boldsymbol{u}(t) = \boldsymbol{u}_1 + \boldsymbol{u}_2$, where:

$$\begin{cases} \boldsymbol{u}_1 = -\boldsymbol{\gamma}_1 |s|^{\lambda} sign(s),\ \lambda \in]0{,}0.5[\\ \dot{\boldsymbol{u}}_2 = -\boldsymbol{\gamma}_2 sign(s) \end{cases} \tag{11}$$

γ_1 and γ_2 are positive gains. The finite time convergence is obtained by the conditions defined in (12).

$$\begin{cases} \boldsymbol{\gamma}_2 > \frac{C_0}{c_{min}} \\ \boldsymbol{\gamma}_1 \geq \sqrt{\frac{4C_0(c_{max}\gamma_2 + C_0)}{c^2{}_{min}(c_{min}\gamma_2 - C_0)}} \end{cases} \tag{12}$$

3.2. Application to a coaxial octorotor:

The octorotor UAV is controlled by addressing a virtual control vector of total thrust and three moments. The general block diagram for a multirotor UAV's control architecture is presented in Fig. 3. A general scheme for controlling such robots is cascade control, in which there is an outer loop position feedback that generates the desired attitude angles for the inner loop. The outer loop is fed by the desired trajectory planned for the UAV. The inner loop is responsible for the stability of the vehicle with an input of the desired attitude angles. In this work, the control strategy in x,y positions are based on a simple linear PD controller. The STA is implemented for controlling altitude and attitude angles for the octorotor.

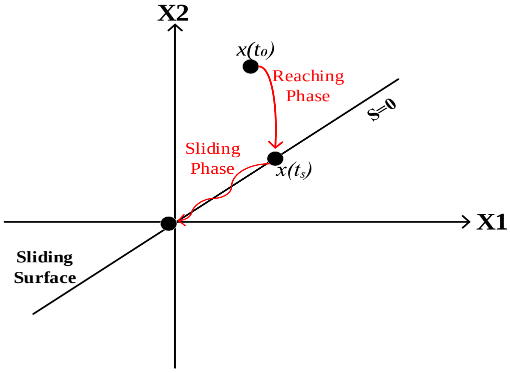


Fig. 2: Sliding mode controller phases.

Let $\boldsymbol{x}_1 = [\, z\ \phi\ \theta\ \psi \,]^T$ and $\boldsymbol{x}_2 = [\, \dot{z}\ \dot{\phi}\ \dot{\theta}\ \dot{\psi} \,]^T$. The system is formulated as follows:

$$\begin{cases} \dot{\boldsymbol{x}}_1 = \boldsymbol{x}_2 \\ \dot{\boldsymbol{x}}_2 = f(x,t) + g(x,t)\boldsymbol{u}(t) + \boldsymbol{\mu}(t) \end{cases} \tag{13}$$

such that $\boldsymbol{\mu}(t) = [\mu_1\ \mu_2\ \mu_3\ \mu_4]^T$ is the external disturbance vector, the vector *f(x)* and the matrix *g(x)* are given as:

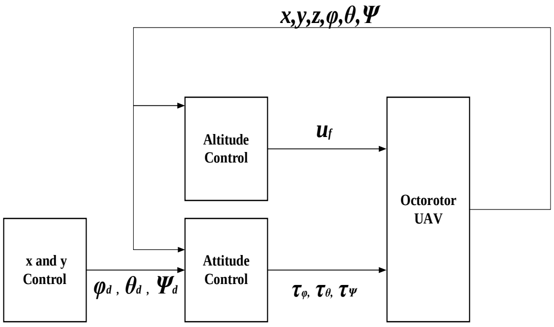


Fig. 3: Octorotor's control architecture.

$$f = \begin{bmatrix} -g \\ \frac{I_{yy} - I_{zz}}{I_{xx}} \dot{\theta}\dot{\psi} \\ \frac{I_{zz} - I_{xx}}{I_{yy}} \dot{\phi}\dot{\psi} \\ \frac{I_{xx} - I_{yy}}{I_{zz}} \dot{\phi}\dot{\theta} \end{bmatrix}, g = \begin{bmatrix} \frac{(\cos\phi\cos\theta)}{m} & 0 & 0 & 0 \\ 0 & \frac{1}{I_{xx}} & 0 & 0 \\ 0 & 0 & \frac{1}{I_{yy}} & 0 \\ 0 & 0 & 0 & \frac{1}{I_{zz}} \end{bmatrix} \tag{14}$$

An actuator failure can be defined as:

$$\bar{v}_i = (1 - \epsilon_i)v_i = (1 - \epsilon_i)\,\omega_i{}^2 \tag{15}$$

In a fault free condition $\epsilon_i = 0$, and when $\epsilon_i = 1$ the actuator effectiveness is totally lost. Using the relations (5) and (7), the system with faults can be represented as in (13) by:

$$\begin{cases} \dot{\boldsymbol{x}}_1 = \boldsymbol{x}_2 \\ \dot{\boldsymbol{x}}_2 = f(x,t) + g(x,t)\boldsymbol{B}\boldsymbol{v} - g(x,t)\boldsymbol{B}\boldsymbol{\epsilon}\boldsymbol{v} \\ \quad\ = f(x,t) + g(x,t)\boldsymbol{u} - g(x,t)\boldsymbol{B}\boldsymbol{\epsilon}\boldsymbol{v} \end{cases} \tag{16}$$

where $\boldsymbol{\epsilon} = diag(\epsilon_1, \dots, \epsilon_8)$, is a matrix that represents the failures of individual actuators. Examining the structure of the faults in (16), they can be treated as matched uncertainties and then they can be entirely rejected by the sliding mode control [19].

Assuming that the UAV behave in a bounded performance with no acrobatic behavior, the matrix $g(x)$ is bounded and invertible, while the disturbance vector is assumed to be norm bounded, as well, and satisfy:

$$|\dot{\mu}_i(t)| < \beta_i \tag{17}$$

The criteria for choosing the sliding surface s are presented by:

$$\boldsymbol{s} = \dot{\boldsymbol{e}} + \boldsymbol{\alpha}\boldsymbol{e} \tag{18}$$

where $\boldsymbol{e} = \boldsymbol{x}_{1d} - \boldsymbol{x}_1$ and $\dot{\boldsymbol{e}} = \boldsymbol{x}_{2d} - \boldsymbol{x}_2$ are the position and velocity errors, and $\boldsymbol{\alpha}$ is a positive definite matrix. The controller is designed by the following input equation (19).

$$\boldsymbol{u} = g^{-1}(\ddot{\boldsymbol{x}}_{1d} - \boldsymbol{\alpha}\dot{\boldsymbol{e}} - \boldsymbol{\gamma}_1 |s|^{0.5} sign(s) - \boldsymbol{\gamma}_2 \int_0^t sign(s(\tau))d\tau - f(x)) \tag{19}$$

The gain matrices are defined by:

$$\begin{cases} \gamma_{2,i} > \beta_i \\ \gamma^2{}_{1,i} > 4\gamma_{2,i} \end{cases} \qquad i = 1,2,3,4 \tag{20}$$

In an octorotor UAV, any actuator fault will affect moments and total thrust and generate a perturbation vector $\mu(t)$. STA is designed to compensate the introduced perturbation and recover the loss in virtual input $\boldsymbol{u}$, in other words keeping the system on the sliding plane. Redundancy in octorotor geometry aids the robust controller to redistribute efforts among nonfaulty actuators.

3.3. PFTC Algorithm Validation:

The proposed nonlinear robust PFTC approach is examined in this section by simulating the behavior of the derived model for octorotor UAV and controllers in case of fault free situation of the vehicle while performing a 3D helical complex path using MATLAB. The tracking error of the nonlinear controller is compared with that of a linear PID controller designed in our previous work. Then, three fault scenarios are subjected to the UAV's motors: partial effectiveness loss, total loss, and significant noise. The results are also reported and discussed.

The proposed second order SMC is first examined in a fault free condition. The gains of the STA controller are well tuned and given in (21) to compensate the injected faults.

$$\begin{aligned}\boldsymbol{\gamma_1} &= diag(5,5,5,5),\\ \boldsymbol{\gamma_2} &= diag(5,5,5,5),\\ \boldsymbol{\alpha} &= diag(42,26,26,26)\end{aligned} \tag{21}$$

A function is proposed in (22) to smooth the sign function in (19).

$$sign(s) = \frac{s}{|s|+0.05} \tag{22}$$

and the desired 3D trajectory is a helical path defined by:

$$\begin{cases} x = 5\cos(0.05t) + 7.5 \\ y = 5\sin(0.05t) + 7.5 \\ z = 0.1t \end{cases} \tag{23}$$

where t is the simulation time expressed in seconds and x,y, and z are the desired position for the UAV expressed in meters. The desired and real trajectories for the octorotor controlled by STA is shown in Fig. 4.

The corresponding tracking error in three directions is shown in Fig. 5 with an error less than 0.05 meters. Furthermore, the desired and measured attitude angles are presented in Fig. 6 showing a suitable tracking performance. To compare the performance of the UAV controlled by a nonlinear robust controller with that of a linear controller, same scenario is simulated on an octorotor vehicle with a well-tuned PID controller. Comparing Fig. 7 to Fig. 5, it can be clearly seen that the proposed nonlinear controller has significantly reduced the tracking errors to about half of that in linear controller.

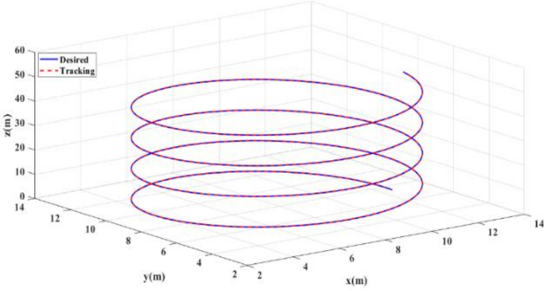


Fig. 4: Trajectory of the octorotor with STA controller.

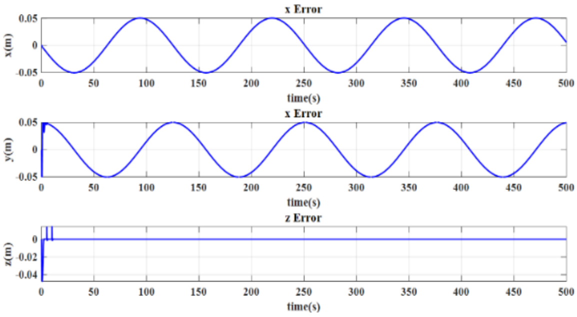


Fig. 5: Tracking error of the octorotor with STA controller while tracking 3D path.

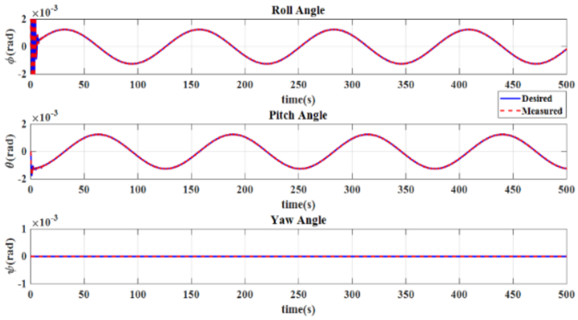


Fig. 6: Desired and measured attitude angles of the octorotor with STA controller while tracking 3D path.

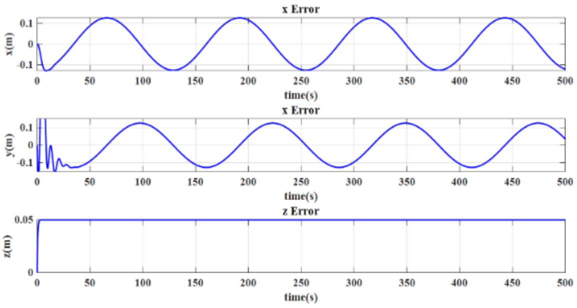


Fig. 7: Tracking error of the octorotor with PID controller while tracking 3D path.

After testing the proposed controller in a fault free approach, the partial loss of effectiveness in the actuators is then investigated. The simulation results are reported by injecting four successive motors partial efficiency loss failures. Table I shows the injected failures according to its percentage and time to each corresponding motor position. The same helical trajectory in (23) is considered for this experiment. The angular speeds of the eight motors are shown in Fig. 8, which also distinguishes the healthy and faulty motors.

For more clarity, Fig. 9 describes the performance of the UAV in each of the three axes by showing the desired and measured positions. This result shows suitable tracking performance with fast robust response even in partial effectiveness loss of the motors.

The total actuators loss is then studied by injecting four successive motors total loss failures with the same sequence as in Table I but with efficiency loss of 100% for all four motors. Same helical trajectory in (23) is desired in this experiment for the UAV to move through. The angular speeds of the eight motors are shown in Fig. 10, which shows the four faulty motors and the redistribution among healthy motors by the robust controller in a smooth manner. Despite of total loss in the effectiveness of the motors, Fig. 11 shows respectively fast response and fault recovery with good tracking behavior in three positions.

Finally for testing the robustness of the proposed controller, a case of significant external disturbance specified by noise input signal speed is injected to motor1 when the UAV is performing a complex maneuver. The input signal is contaminated by a White Gaussian Noise at 200 sec to produce a noisy signal as shown in Fig. 12. The desired and measured attitude angles are stated in Fig. 13, which shows the effect of the injected noise on the three angles performance. Although the noise affects the attitude angles, these effects keep the octorotor in a desirable performance as shown in Fig. 14 that presents the three tracking errors in three directions of the vehicle.

TABLE I . Successive Partial Loss Fault Events.

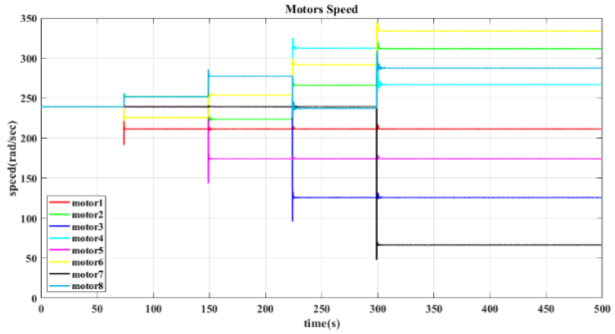


Fig. 8: Angular speeds for the motors of the octorotor with STA controller subjected to four partial successive failures.

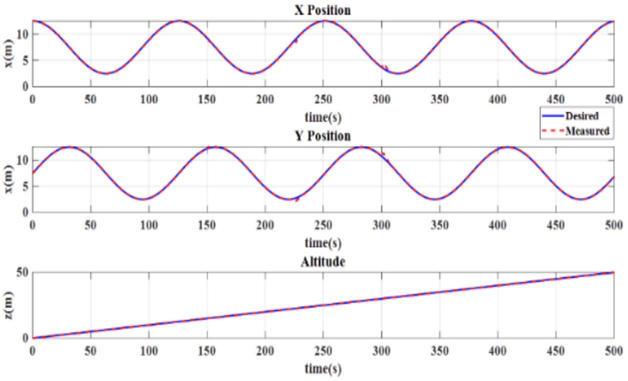
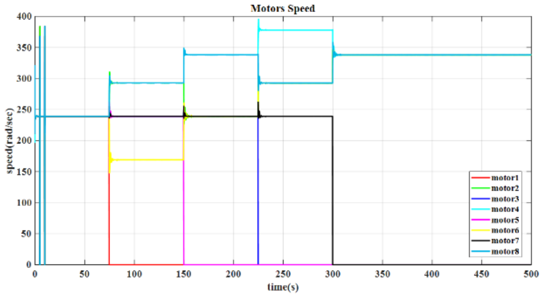


Fig. 9: Desired and measured position of the octorotor with STA controller subjected to four partial successive failures.

4. AFTC Design Structure

This section studies the AFTC design strategy for an octorotor UAV in the case of actuators total loss of effectiveness. Two main and important steps are considered when designing such controllers: fault detection and fault tolerance or recovery.



#	Time (sec)	Motor	Efficiency loss %
1	75	m_1	20
2	150	m_5	40
3	225	m_3	60
4	300	m_7	80

Fig. 10: Angular speeds for the motors of the octorotor with STA controller subjected to four total successive failures.

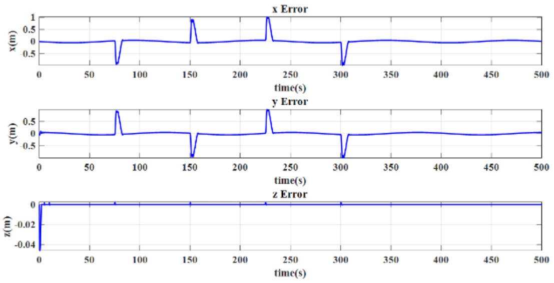


Fig. 11: Position errors of the octorotor with STA controller subjected to four total successive failures.

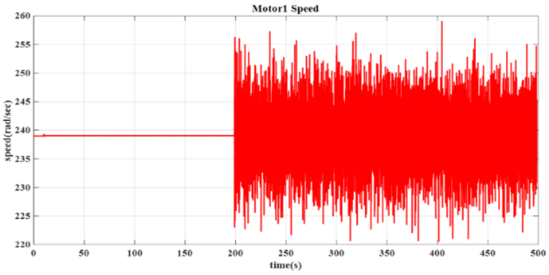


Fig. 12: Motor1 of the octorotor with STA controller speed subjected to Gaussian noise.

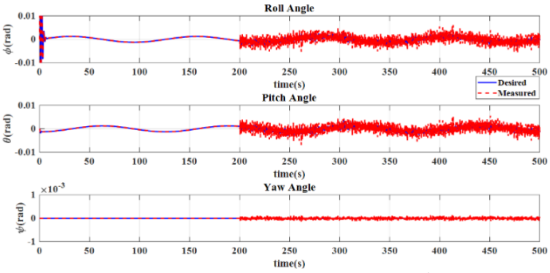


Fig. 13: Desired and measured attitude angles of the octorotor with STA controller with motor subjected to noise.

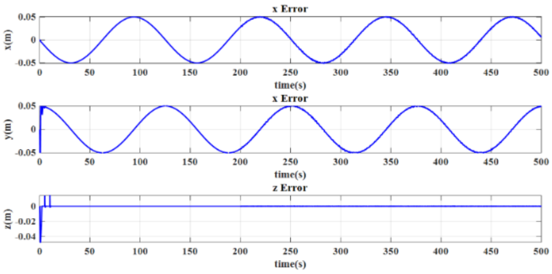


Fig. 14: Tracking error of the octorotor with motor subjected to noise.

4.1. Fault Detection Technique

Fault detection is the primary step that must be designed to detect the faulty element regarding its identity, position, and time of occurrence. A nonlinear observer relying on a Thau observer is proposed in this work to detect actuators' fault when it exists in the UAV during its normal flight trajectory. By referring to dynamic equations, it is obvious that any fault that attacks one of the motors should alters the behavior of the UAV according to the attitude and altitude parameters. These parameters can be presented in a simplified state space representation (24) as a base to the proposed observer by separating the linear component from the nonlinear one.

$$\begin{gathered}\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \Delta\big(\boldsymbol{x}(t), \boldsymbol{u}(t)\big) \\ \boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t)\end{gathered} \quad (24)$$

Where $\boldsymbol{x} = \left[\, x\; y\; z\; \phi\; \theta\; \psi\; \dot{x}\;\; \dot{y}\; \dot{z}\; \dot{\phi}\; \dot{\theta}\; \dot{\psi}\right]^T$, and $\boldsymbol{u} = [\, u_f\; \tau_\phi\; \tau_\theta\; \tau_\psi\,]^T$ is the virtual control input vector formed from total thrust and the moments with respects to the three attitude axes. The vector $\boldsymbol{y} = [\, x\; y\; z\; \phi\; \theta\; \psi]^T$ is stated as the output vector. $\Delta(x, u) = [0\; 0\; 0\;\; 0\; 0\; 0\; \boldsymbol{\delta}(x, u)\; 0\; 0\; 0]^T$ defines the nonlinear dynamic term with:

$$\boldsymbol{\delta}(x, u) = \begin{bmatrix}(\cos\phi \sin\theta \cos\psi + \;\sin\phi \sin\psi) \\ (\cos\phi \sin\theta \sin\psi - \;\sin\phi \cos\psi) \\ (\cos\phi \cos\theta)\end{bmatrix} * \frac{u_f}{m} - \begin{bmatrix}0\\0\\g\end{bmatrix} \quad (25)$$

Then the nonlinear Thau observer can be stated as follows:

$$\begin{gathered}\dot{\hat{\boldsymbol{x}}}(t) = \boldsymbol{A}\hat{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \Delta\big(\hat{\boldsymbol{x}}(t), \boldsymbol{u}(t)\big) + \boldsymbol{K}(\boldsymbol{y}(t) - \hat{\boldsymbol{y}}(t)) \\ \hat{\boldsymbol{y}}(t) = \boldsymbol{C}\hat{\boldsymbol{x}}(t)\end{gathered} \quad (26)$$

Such that $\hat{\boldsymbol{x}}(t)$ is the observer state, $\hat{y}(t)$ is the observer output, and $\boldsymbol{K}$ is the observer feedback gain matrix. More details regarding the observer condition design and stability analysis can be found in a previous author's work [20]. When the UAV is exposed to a fault, the system will lose the ability to follow its desired states. Therefore, the residue $\boldsymbol{r} = \boldsymbol{y}(t) - \hat{\boldsymbol{y}}(t)$ will deviate from zero indicating a fault existence.

In an octorotor UAV situation, any particular motor that is subjected to fault will cause a predefined sign set of three residues roll, pitch and yaw. Eight various sign collection are acquired so the detecting topology relies on sign combination. Table 2. shows the eight different sign combinations in terms of faulty motor. An example of the behavior of Thau observer is presented in Fig. 15 in the situation of fault in motor3 at time 50 sec. In order to handle detection false alarms, a threshold value is considered such that the decision is taken due to the comparison with this value.

4.2. Fault Tolerance and Recovery

AFTC strategies are classified into two different approaches: offline and online methods. Differentiating criteria between these two methods is based on the instance of solving the control parameters of the recovery algorithm. Offline FTC methods are based on reconfiguring the remaining nonfaulty motors by distributing the control efforts among it when a fault is detected in a specific defected motor. The distribution protocol is computed offline by solving the constrained optimization problem that guarantees the stability and desired performance of the octorotor. Once all the fault cases are studied and the corresponding new control configurations are computed, a lookup table is formed of new angular velocity gains for the nonfaulty actuators is built and fed to the flight controller to rely on when a fault occurs.

TABLE II. Residues sign combination relative to faulty motor position.

motor	e_φ	e_θ	e_ψ
1	+	+	+
2	+	+	-
3	+	-	-
4	+	-	+
5	-	-	+
6	-	-	-
7	-	+	-
8	-	+	+

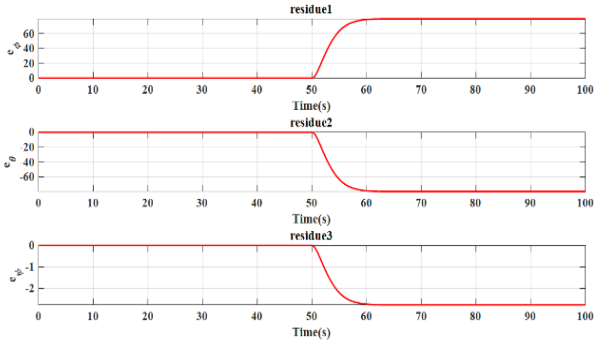


Fig. 15: Thau observer behavior of motor3 total fault situation.

A constrained optimization problem may be formulated as follows:

$$\min_v \frac{1}{2} v^T v$$
$$\text{under the conditions } \begin{cases} v_{min} \leq v \leq v_{max} \\ Bv = u \end{cases} \quad (26)$$

Where $\boldsymbol{v} = [\omega_1{}^2 \ldots \omega_8{}^2]^T$ is the actual control input to the actuators specified by the angular speeds, $\boldsymbol{u} = [\, u_f \; \tau_\phi \; \tau_\theta \; \tau_\psi \,]^T$ is the virtual control input, and $\boldsymbol{B}$ is the effectiveness input matrix stated in (6). The optimization problem is solved applying different algorithms such as parametric programming method [21], and Particle Swarm Optimization (PSO) [22].

On the other side, the online AFTC is an algorithm that runs onboard in both nonfaulty and faulty conditions. The upper-level motion control component for the UAV moves the system through its desired values, meanwhile the components of the virtual input vector $\boldsymbol{u} = [\, u_f \; \tau_\phi \; \tau_\theta \; \tau_\psi \,]^T$ are computed through this algorithm according to the driving control laws. The computed virtual inputs are distributed to motors through the angular speed by the relation (5). At the moment when the fault detection block detects a fault and specifies its position, the redistribution manner is updated taking in consideration the faulty and noneffective motor. The proposed strategy to solve this problem is based on pseudo-inverse control allocation algorithm which solves the optimization equation presented in (27) by the solution presented in (28) through an online manner.

$$\begin{gathered} \min_u \boldsymbol{v}(t)^T \boldsymbol{E} \boldsymbol{v}(t) \\ \ni \; \boldsymbol{B}\boldsymbol{v}(t) = \boldsymbol{u}(t) \end{gathered} \quad (27)$$

$$\boldsymbol{v} = [\boldsymbol{E}\boldsymbol{B}^T(\boldsymbol{B}\boldsymbol{E}\boldsymbol{B}^T)^{-1}]\boldsymbol{u} \quad (28)$$

Where $\boldsymbol{E} = diag(\varepsilon_1, \ldots, \varepsilon_8)$ is the effectiveness weight matrix of the actuators which is defined by $0 < \varepsilon_i < 1$, such that $\varepsilon_i = 0$ and $\varepsilon_i = 1$ are related respectively to complete effectiveness loss and no-fault indication for motor i. This method is also denoted by control allocation reallocation method.

The two proposed AFTC algorithms well studied and examined in a previous author's work [20] and a comprehensive comparison between both controllers based on the simulation results was performed. The functional block diagrams of both AFTC approaches are shown in Fig. 16 and Fig. 17.

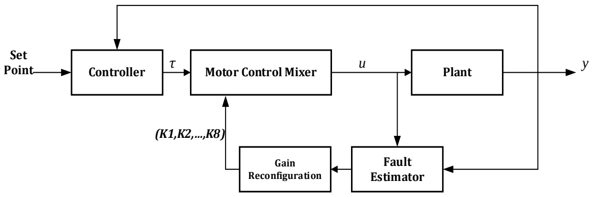


Fig. 16: Offline reconfiguration AFTC diagram. [20]

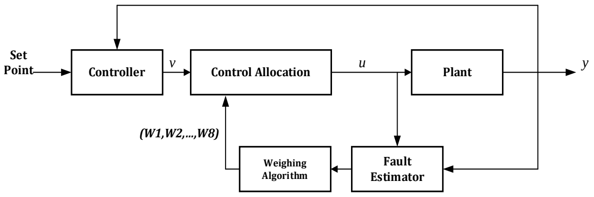


Fig. 17: Online control allocation reallocation AFTC diagram. [20]

4.3. AFTC Algorithm Validation

The proposed nonlinear AFTC approach is examined in this section by simulating the behavior of the derived model for octorotor UAV and controllers in case of fault free situation of the vehicle while performing a 3D helical complex path as in (23) using MATLAB. The outer loop controller is designed based on a simple PID controller, and the inner loop is based on a PD controller. Four successive total failures are injected with same manner as section 3.3 before. Based on the comparison between offline and online AFTC performed in the authors previous work [20] and due to the smooth actuators performance noted while applying the online approach, the experimental results of the proposed online topology are examined in this paper to be compared later on with the PFTC algorithm.

The desired and real trajectories for the octorotor with online AFTC subjected to four total successive failures is shown in Fig. 18 which shows an acceptable tracking performance of the UAV for a complex 3D trajectory. The proposed fault detection Thau observer algorithm shows an accurate performance in Fig. 19 for detecting consecutive faults when it occurs. Position errors in three dimensions are presented in Fig. 20, which shows a small deviation when successive total faults exist. The corresponding angular speeds for eight motors are shown in Fig. 21 that validates the reallocation algorithm of the angular speeds when fault occurs through nonfaulty motors.

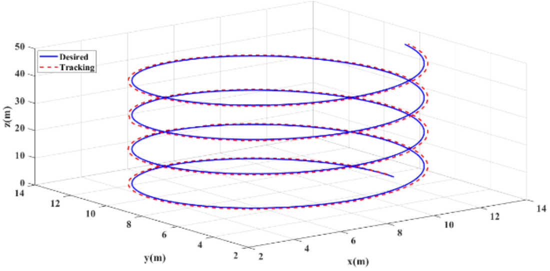


Fig. 18: Trajectory of the octorotor with online AFTC subjected to four total successive failures.

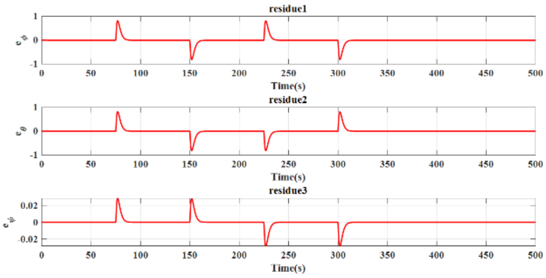


Fig. 19: Residues for the three attitude angles of the octorotor with online AFTC subjected to four total successive failures.

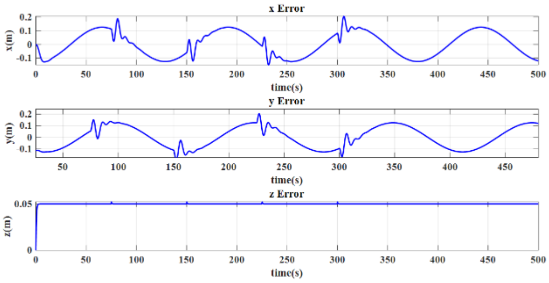


Fig. 20: Position errors of the octorotor with online AFTC subjected to four total successive failures.

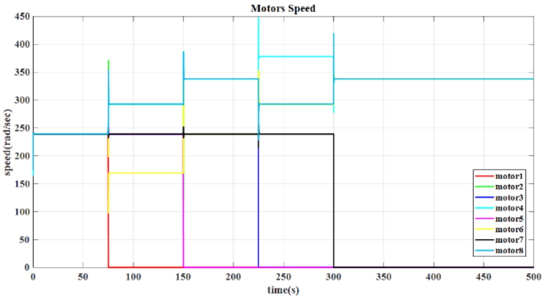


Fig. 21: Angular speeds for the motors of the octorotor with online AFTC subjected to four total successive failures.

5. Results Analysis and Comparison Study

This section analysis the experimental results by summarizing the main advantages and disadvantages of both passive and active fault tolerant controllers for the redundant type UAV, octorotor, while performing a 3D complex trajectory and under actuators loss of effectiveness. A comparison is done between these two strategies taking in consideration recovery time, position deviation when fault exists, and control efforts performed by the UAV in three rotation axes. As an extension to work done in [20], a novel weighting criterion is proposed to compare offline AFTC, online AFTC, and STA PFTC. This criterion is summarized in TABLE III. The main focus of this criteria are the applicability and simplicity indexes. For that purpose, a weighing scale is performed in the table.

By comparing Fig. 11 to Fig. 20, which both show the position errors of the octorotor subjected to four total successive failures, the AFTC controller has shown less deviation errors in both x and y axes when fault occurs. This expected result can be related to the fact that AFTC strategies consists of a fault detection algorithm that detects specific fault then a predesigned fault tolerance algorithm will take the decision to recover this fault, rather than PFTC strategies which deal with the fault as a general disturbance effect. Otherwise, the both figures show that the recovery time is less when applying PFTC approach, this may be justified by the detection time needed in AFTC approach.

An additional parameter result is further examined for more physical comparison between two FTC approaches in case of actuators loss of effectiveness. The control efforts specified by the control moments in three orientation direction are shown in Fig. 22 and Fig. 23 for both UAVs with PFTC and AFTC respectively to tolerate total failure. It is clear that the control efforts are less in AFTC approach this is due to the fact that a correction term is added in this approach to compensate the fault when it exists by reallocation of the angular speeds for the nonfaulty motor for maintaining stability and desired behavior of the UAV.

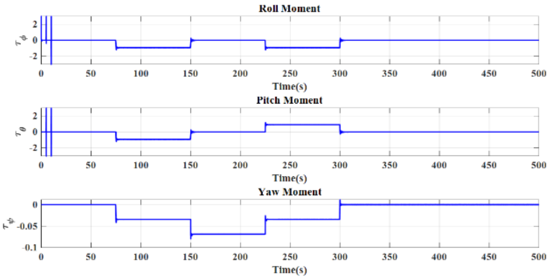


Fig. 22: Control input efforts of the octorotor with PFTC subjected to four total successive failures.

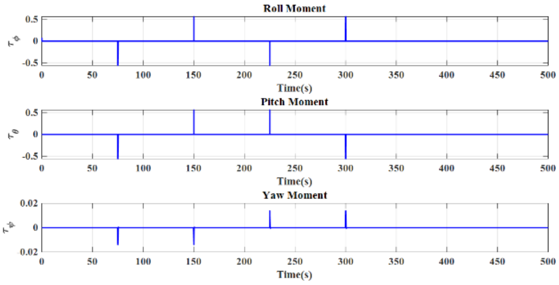


Fig. 23: Control input efforts of the octorotor with AFTC subjected to four total successive failures.

Furthermore, the main aspect for designing the PFTC algorithm is robust control design. So, such algorithms can deal with external disturbances and perturbations that affect the control system. This was validated through Fig. 9 that shows the good performance of the octorotor in the presence of partial loss of effectiveness in the actuators. This fault can be considered as an external disturbance that affect the rotor speed. Additionally, Fig. 14 shows a desirable performance in the presence of external disturbances presented by Gaussian noise that affects the speed controller of the high spinning motor.

Although PFTC shows some advantages through its robustness design, but it deals with faults as a general external disturbance that must be compensated. This aspect makes such approaches not feasible to some fault cases, rather than AFTC approaches which specifies the fault and take the action according to that specified fault. ***Remark:*** Although, the three methods listed in TABLE III show the same total weight, researcher can select the suitable method due to his research and application specified index interest.

TABLE III: Comparison weighting table for various FTC algorithms.

Evaluation Index	***FTC Method***		
	AFTC		***PFTC***
	Offline	***Online***	***STA***
Design Simplicity *(simpler)*	3	2	2
Actuator Behavior *(more stable)*	1	3	2
Computation Cost *(less cost)*	3	2	2
FDD *(additional burden)*	3	3	0
Recovery Time *(less)*	1	1	3
Control Effort *(less)*	1	1	3
Total Weight	12	12	12

6. Conclusion and Future Prospect

This paper proposed a passive fault tolerance algorithm based on robust controller for a redundantly actuated UAV. The proposed controller is a nonlinear controller based on a super twisting sliding mode control. The proposed algorithm is planned to control a coaxial octorotor UAV during normal behavior and actuators faults. This proposed approach aimed to robustly redistribute efforts among unfaulty during the fault. The performed simulation results were implemented applying 3D helical trajectory in four normal, partial loss, total loss and noisy motor cases. An active fault tolerance algorithm specified by a nonlinear detection observer for actuators subjected to total fault was also presented. An offline and online strategies were investigated for system recovery when a fault is detected. A novel physical comparison was made between active and passive algorithms passing through its advantages and disadvantages.

We are now working on real implementation of the proposed algorithms and examining its execution complexity and weight through onboard implementation. In addition, a hybrid passive-active FTC will be designed to benefit from the advantages of two approaches by combining robustness to fault detection and recovery.

References

[1] H. Shakhatreh et al., "Unmanned Aerial Vehicles (UAVs): A Survey on Civil Applications and Key Research Challenges," *in IEEE Access*, vol. 7, pp. 48572-48634, 2019.

[2] B. Yu, Y. Zhang, Y. Yi, Y. Qu, and P. Lu, “Fault detection for partial loss of effectiveness faults of actuators in a quadrotor unmanned helicopter,” *in Proceeding of the 11th World Congress on Intelligent Control and Automation*, 2014, pp. 3204–3209.

[3] A.-R. Merheb, H. Noura, and F. Bateman, “Emergency control of AR drone quadrotor UAV suffering a total loss of one rotor,” *IEEE ASME Trans. Mechatron*., vol. 22, no. 2, pp. 961–971, 2017.

[4] C. D. Pose, J. I. Giribet and A. S. Ghersin, "Hexacopter fault tolerant actuator allocation analysis for optimal thrust," *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2017, pp. 663-671.

[5] A. Marks, J. F. Whidborne, and I. Yamamoto, “Control allocation for fault tolerant control of a VTOL octorotor,” *in Proceedings of 2012 UKACC International Conference on Control*, 2012, pp. 357–362.

[6] M. Saied, B. Lussier, I. Fantoni, H. Shraim, and C. Francis, “Active versus passive fault-tolerant control of a redundant multirotor UAV,” *Aeronaut. J.*, vol. 124, no. 1273, pp. 385–408, 2020.

[7] S. H. Almutairi and N. Aouf, “Aircraft robust flight tracking control against actuator efficiency faults,” *in 2015 19th International Conference on System Theory, Control and Computing (ICSTCC)*, 2015, pp. 901–906.

[8] T. Li, Y. Zhang, and B. W. Gordon, “Nonlinear fault-tolerant control of a quadrotor UAV based on sliding mode control technique,” *IFAC proc. vol.*, vol. 45, no. 20, pp. 1317–1322, 2012.

[9] R. R. Benrezki, M. Tadjine, F. Yacef, and O. Kermia, “Passive fault tolerant control of quadrotor UAV using a nonlinear PID,” *in 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2015, pp. 1285–1290.

[10] B. Wang and Y. Zhang, “Adaptive sliding mode fault-tolerant control for an unmanned aerial vehicle,” *Unmanned syst.*, vol. 05, no. 04, pp. 209–221, 2017.

[11] A.-R. Merheb, H. Noura, and F. Bateman, “Design of passive fault-tolerant controllers of a quadrotor based on sliding mode theory,” *Int. J. Appl. Math. Comput. Sci.*, vol. 25, no. 3, pp. 561–576, 2015.

[12] H. Alwi and C. Edwards, “Fault tolerant control of an octorotor using LPV based sliding mode control allocation,” *in 2013 American Control Conference*, 2013, pp. 6505–6510.

[13] M. Saied, B. Lussier, I. Fantoni, H. Shraim, and C. Francis, “Passive fault-tolerant control of an octorotor using super-twisting algorithm: Theory and experiments,” *in 2016 3rd*

Conference on Control and Fault-Tolerant Systems (SysTol), 2016, pp. 361–366.

[14] T. Schneider, G. Ducard, K. Rudin, and P. Strupler, “Fault-tolerant control allocation for multirotor helicopters using parametric programming,” *rN.* 2012;1:r2.

[15] I. Sadeghzadeh, A. Chamseddine, Y. Zhang, and D. Theilliol, “Control allocation and re-allocation for a modified quadrotor helicopter against actuator faults,” *IFAC proc. vol.*, vol. 45, no. 20, pp. 247–252, 2012.

[16] Q.-L. Zhou, Y. Zhang, C.-A. Rabbath, and J. Apkarian, “Two reconfigurable control allocation schemes for unmanned aerial vehicle under stuck actuator failures*,” in AIAA Guidance, Navigation, and Control Conference*, 2010, 10.2514/6.2010-8419.

[17] S. Bouabdallah, “Design and Control of Quadrotors With Application to Autonomous Flying,” *PhD thesis*, *Ecole Polytechnique Federale de Lausanne*, 2007.

[18] L. Derafa, A. Benallegue, L. Fridman, “Super twisting control algorithm for the attitude tracking of a four rotors UAV,” *J. Franklin Inst.*, vol. 349, no. 2, pp. 685–699, 2012.

[19] C. Edwards and S. Spurgeon, “Sliding Mode Control: Theory And Applications,” *Taylor & Francis*, 1998.

[20] H. Mazeh, H. D. Taghirad and J. Sahili, "Offline and Online Active Fault Tolerant System For Multirotor Drones," *2021 9th RSI International Conference on Robotics and Mechatronics (ICRoM)*, 2021, pp. 151-156.

[21] H. Mazeh, M. Saied, H. Shraim and C. Francis, "Fault-tolerant control of an hexarotor unmanned aerial vehicle applying outdoor tests and experiments", *IFAC-PapersOnLine*, vol. 51, no. 22, pp. 312-317, 2018.

[22] H. Mazeh and J. Sahili, "Fault - tolerant control of a multirotor unmanned aerial vehicle applying particle swarm optimization", *2019 7th International Conference on Robotics and Mechatronics (ICRoM)*, pp. 619-624, 2019.

Biography

Hussien Mazeh is currently a Ph.D. student in control engineering at K. N. Toosi University of Tech. He received his Master’s Degree in Electrical-Electronic Engineering from the Lebanese University in the 2011, and double program degree of Research Master2 in Robotics and Intelligent Systems from both Lebanese University and UTC France in 2017. His current research interests include Multi-agents Robots and UAVs, and he is currently a member of ARAS Multi-Agent Robotics.

Hamid D. Taghirad has received his B.Sc. degree in mechanical engineering from Sharif U. of Tech., Tehran, Iran, in 1989, his M.Sc. in mechanical engineering in 1993, and his Ph.D. in electrical engineering in 1997, both from McGill University, Canada. He is currently a Professor and the Director of the Advanced Robotics and Automated System (ARAS) at K.N. Toosi University of Technology, Tehran, Iran. He is a senior member of IEEE, member of the board of Industrial Control Center of Excellence (ICCE), at K.N. Toosi University of Technology, and Editorial board of International Journal of Robotics: Theory and Application, and International Journal of Advanced Robotic Systems. His research interest is robust and nonlinear control applied to robotic systems. His publications include five books, and more than 250 papers in peer-reviewed international Journals and conference proceedings.

Jihad Sahili received a B.S. degree in physics in 1988 and a diploma in mechanical engineering in 1991, both from the Lebanese University, the master’s degree in intensive calculus from the Lebanese university in collaboration with AUPELF-UREF in 1999, and the Ph.D. degree in Mechanics and Energetics from the University of Burgundy-France in 2004, He is currently a professor at the Faculty of Engineering in the Lebanese University, Beirut-Lebanon.