Actuator Fault Accommodation of an Aerial Vehicle Described by Takagi-Sugeno Models

Hajer MLAYEH*, Atef KHEDHER
LATIS Laboratory, Université de Sousse, Ecole Nationale d'Ingénieurs de Sousse, Sousse 4023, Tunisia
mlayehhajer@gmail.com (**Corresponding author*)

Abstract: This study addresses the problem of fault accommodation in nonlinear systems described by multiple-models. The considered nonlinear system is an aerial vehicle with four-rotors that is subject to actuator fault. The complexity of the system dynamics poses a significant challenge to its control, especially with the presence of failures. Consequently, a new model is proposed and build, based on Takagi-Sugeno (TS) multiple models. The main objective is to stabilize the vehicle attitude even with the actuator fault. An accommodation strategy is proposed based on TS models. The stability of the suggested model is proved using Lyapunov theory and linear matrix inequalities (LMIs).

Keywords: Aerial vehicle, Actuator fault, Takagi-Sugeno, Multiple models, FTC.

1. Introduction

The field of aerial vehicles has witnessed remarkable advancements in recent years, specifically with unmanned aircrafts, commonly known as drones (AL-Dosari, Hunaiti & Balachandran, 2023). These pilotless systems find application in different fields, ranging from surveillance, security, medicine, military and agriculture to search and rescue missions. Notably, quadrotors, a subset of drones characterized by their four-rotor configuration, exhibit superior performance in security and inspection missions, due to their precise and stable hovering capabilities as well as high manoeuvrability (Sonugur, 2023).

In security-sensitive applications, drones become more vulnerable to failure scenarios, which requires the implementation of Fault-Tolerant Control (FTC) mechanisms (Mlayeh & Khedher, 2023b; Jamel, Khedher & Othman, 2017). Such control is crucial for maintaining vehicle objectives, even in the face of failures.

There are various techniques that can be used to control a drone under failure. For instance, Zhang et al. (2022) employed a predictive control strategy to reconfigure an aerial system. Such FTC allowed the handling of diverse control constraints, including those resulting from the inherent limitations of the actuators. Nevertheless, the application of this technique introduces high computational load, as it requires real-time optimization. Expert systems were employed by Feng et al. (2023) to handle different issues in a vehicle's wireless network, including multiple sensors concurrent failure. The proposed framework enabled both fault detection and diagnosis. Merheb, Noura & Bateman (2014) applied the sliding mode controller to monitor a drone with one partial rotor defection. In case a rotor was completely lost, hardware redundancy could be employed, as demonstrated by Saied, Shraim & Francis (2023). However, this hardware duplication, although efficient, increases the size, cost and weight of the vehicle. Thus, it is not always privileged. Alternative control frameworks (Dalwadi, Deb & Ozana, 2022) relied on adaptive backstepping to accommodate the complete loss of a drone's motor. Azeem et al. (2024) used Linear-matrix-inequalities (LMIs) to conceive a FTC scheme that tackles the fault in the actuator channel of an octa-rotor aerial vehicle. In the specialized literature (Saied et al., 2023), the Linear-Quadratic Regulator (LQR) was presented as an effective strategy for FTC of a quad-plane vehicle. This approach constructs feedback control by minimizing a cost function that relies on both control and state variables. The primary strengths of LQR lie in its simplicity, along with its capacity to regulate the settling time of state variables and control the amplitude of inputs.

It must be stressed that choosing the suitable control method depends on the fault type, the system nature and, mainly, the mathematical model. Indeed, establishing the appropriate system model is a crucial step in FTC. Linearizing the system equations around an equilibrium point is a widely used technique. For instance, Zhou et al. (2019) used a predictive control based on successive linearization model to control the position of a drone. Despite its simplicity, this linearized model is valid solely in a limited area

https://doi.org/10.24846/v33i2y202407

around the equilibrium point. That makes the system model "unreal" and less representative of the vehicle complexities. Mlayeh & Ben Othman (2022b) and Mlayeh & Khedher (2024) used the nonlinear modelling to establish FTC of an aircraft. Using nonlinear equations made the model more accurate and valid in a larger range. However, some simplifications such as the small angle approximation, had to be made in order to apply the control strategy. Moreover, the complexity of the model posed challenges in establishing certain mathematical theorems or assumptions, such as proving stability through Lyapunov function analysis.

While significant improvements have been made in FTC for nonlinear systems, the majority of the established controllers suffer from complexity and heavy reliance on precise mathematical formulations of the underlying models.

The Takagi-Sugeno (TS) models are a type of Multiple-Models (MMs) (Zhang, Wang & Wang, 2023; Elleuch, Khedher & Othman, 2018) that enable the control of nonlinear systems across a broad operational range, split into distinct linearized regions. TS models are one of the favoured tools for representing nonlinear systems, by approximating complex, nonlinear relationships (Khedher, Elleuch & Ben Othman, 2022; Jamel et al., 2010a). Such characteristic enables them to effectively capture the complexity of nonlinear systems and to handle nonlinearities in a structured manner. At the same time, linear tools and methods can be exploited by providing a piecewise linear representation, which leads to efficient modelling and control solutions (Bouguila et al., 2013).

This paper investigates the problem of FTC of a quad-rotor aerial vehicle using TS models. The considered nonlinear system was first introduced by Mlayeh & Khedher (2023a) and Mlayeh et al. (2021). In the aforementioned specialized literature, authors have proceeded either by linearizing the system and using PD control, or by keeping nonlinear equations and using recursive control techniques. The proposed control dealt with disturbances and structural faults. However, theoretical stability proofs were never made. This study proposes a new model of the vehicle, build, for the first time, using TS models. The main objective is to stabilize the vehicle orientation when faults affect the four rotors simultaneously. An accommodation strategy is proposed using LQR technique and based on the developed sub-models. The stability of the new model is proved using Lyapunov theory and LMIs.

The rest of this article is structured as follows. Section two describes the vehicle and its nonlinear dynamics. Section three represents the system modelling, using TS approach, as well as stability proofs, using Lyapunov theory. In section four, the accommodation scheme is presented using LQR technique. Section five is dedicated to simulation results. Finally, section six presents the conclusion of the present paper.

2. Vehicle Description

The vehicle considered in this study is the "DraganFly" four-rotor drone (Bresciani, 2008). A simplified structure of the drone consists of two crossing arms and two pairs of motors positioned along each of its arms. The pairs of motors spin at opposite directions to maintain stability. Moreover, when all motors rotate at identical speeds, the vehicle can sustain a consistent heading during hovering. The drone can perform three rotational movements known as roll, pitch and yaw, according to axis (x, y, z) respectively. The expression of these Euler angles (φ, θ, ψ) and their derivatives define the angular position of the drone and can be obtained using kinematics and dynamics equations along with Newton's second law (Mlayeh & Ben Othman, 2022a).

The nonlinear equations in (1) represent the final expression of the vehicle's attitude

$$\begin{cases} \ddot{\phi} = \eta_1\dot{\theta}\dot{\psi} + \alpha_1\dot{\theta}U_4 + \beta_1 U_1 \\ \ddot{\theta} = \eta_2\dot{\phi}\dot{\psi} + \alpha_2\dot{\phi}U_4 + \beta_2 U_2 \\ \ddot{\psi} = \eta_3\dot{\phi}\dot{\theta} + \beta_3 U_3 \end{cases} \tag{1}$$

where:

$$\begin{cases} \eta_1 = \frac{I_y - I_z}{I_x}; \eta_2 = \frac{I_z - I_x}{I_y}; \eta_3 = \frac{I_x - I_y}{I_z} \\ \alpha_1 = \frac{I_r}{I_x}; \alpha_2 = -\frac{I_r}{I_y} \\ \beta_1 = \frac{L}{I_x}; \beta_2 = \frac{L}{I_y}; \beta_3 = \frac{1}{I_z} \end{cases} \tag{2}$$

U_i represents the system's inputs that depend on the motors' speed, L is the arm size, $I_{x,y,z}$ represent the body inertia and I_r is the inertia moment.

3. Takagi-Sugeno Modelling

The multiple model approach consists of segmenting the system's operating space into a finite set of distinct operating zones. Consequently, the dynamic characteristics of the system within each operating zone can be represented using a simple linear sub model. Each submodel contributes to the system by means of a weighting function. The final approximated system is obtained by establishing connections among the submodels, while accounting for their individual contributions (Jamel et al., 2010b).

The objective in this section is to build a T-S model from the nonlinear system equations given in (1). In general, there exist three main approaches for the construction of fuzzy models: either through identification, when there is data on the inputs and outputs, or through linearization around different operating points, or by a convex polytopic transformation, when an analytical model is available (Chadli & Borne, 2012). The aforementioned strategy is used in this study. In general, a TS model has the following form:

$$\dot{x}(t) = \sum_{i=1}^{N} \mu_i\big(z(t)\big)\big(A_i x(t) + B_i u(t)\big) \tag{3}$$

where $x(.) \in \mathcal{R}^n$ is the vector of state variables, $u(.) \in \mathcal{R}^m$ is the inputs vector and $z(.) \in \mathcal{R}^q$ is the vector of decision variables, also called fuzzy variables. A_i and B_i are constant matrices with appropriate dimensions. $\mu_i(.)$ are weighting nonlinear functions that satisfy the convexity property (4):

$$\sum_{i=1}^{N} \mu_i\big(z(t)\big) = 1 \, and \;\; 0 \le \mu_i\big(z(t)\big) \le 1 \tag{4}$$

where $i \in \{1...N\}$ and N represents the number of models.

3.1 Polytopic Transformation

This technique relies on the boundedness of nonlinear terms and consists of convex transformation of the latter. It permits a maximum reduction of the number of models. The first step consists of identifying nonlinearities in the system, by putting the nonlinear system in the following form:

$$\dot{x}(t) = \begin{bmatrix} f(x) & g(x) \end{bmatrix} \begin{bmatrix} x(t) \\ u(t) \end{bmatrix} \tag{5}$$

The number N of local models depends on the number s of nonlinearities in (5) where $N = 2^s$. The second step is to identify the fuzzy variables z_j, j ∈ {1...s}. The choice is not unique as they might depend on the states, the inputs or both. For each variable z_j, there exist two membership functions $M_1(z_j)$ and $M_2(z_j)$ such that:

$$\begin{cases} M_1(z_j) + M_2(z_j) = 1 \\ z_j = M_1(z_j).z_j max + M_2(z_j).z_j min \end{cases} \tag{6}$$

where z_j is bounded on $[z_j min,\ z_j max]$. The membership functions can be expressed as follows:

$$\begin{cases} M_1(z_j) = \dfrac{z_j - z_j min}{z_j max - z_j max} \\ M_2(z_j) = \dfrac{z_j max - z_j}{z_j max - z_j max} \end{cases} \tag{7}$$

Finally, weighting functions $\mu_i\big(z(t)\big)$ (see the system equations in (8)) can be obtained by multiplying the different membership functions and by following the TS fuzzy rules (Mehran, 2008).

$$\begin{cases}
\mu_1(z) = M_2(z_1) \times M_2(z_2) \times M_2(z_3) \times M_2(z_4) \\
\mu_2(z) = M_2(z_1) \times M_2(z_2) \times M_2(z_3) \times M_1(z_4) \\
\mu_3(z) = M_2(z_1) \times M_2(z_2) \times M_1(z_3) \times M_2(z_4) \\
\mu_4(z) = M_2(z_1) \times M_2(z_2) \times M_1(z_3) \times M_1(z_4) \\
\mu_5(z) = M_2(z_1) \times M_1(z_2) \times M_2(z_3) \times M_2(z_4) \\
\mu_6(z) = M_2(z_1) \times M_1(z_2) \times M_2(z_3) \times M_1(z_4) \\
\mu_7(z) = M_2(z_1) \times M_1(z_2) \times M_1(z_3) \times M_2(z_4) \\
\mu_8(z) = M_2(z_1) \times M_1(z_2) \times M_1(z_3) \times M_1(z_4) \\
\mu_9(z) = M_1(z_1) \times M_2(z_2) \times M_2(z_3) \times M_2(z_4) \\
\mu_{10}(z) = M_1(z_1) \times M_2(z_2) \times M_2(z_3) \times M_1(z_4) \\
\mu_{11}(z) = M_1(z_1) \times M_2(z_2) \times M_1(z_3) \times M_2(z_4) \\
\mu_{12}(z) = M_1(z_1) \times M_2(z_2) \times M_1(z_3) \times M_1(z_4) \\
\mu_{13}(z) = M_1(z_1) \times M_1(z_2) \times M_2(z_3) \times M_2(z_4) \\
\mu_{14}(z) = M_1(z_1) \times M_1(z_2) \times M_2(z_3) \times M_1(z_4) \\
\mu_{15}(z) = M_1(z_1) \times M_1(z_2) \times M_1(z_3) \times M_2(z_4) \\
\mu_{16}(z) = M_1(z_1) \times M_1(z_2) \times M_1(z_3) \times M_1(z_4)
\end{cases} \tag{8}$$

3.2 TS Model of the Drone

Considering the nonlinear equations in (1), the system can be transformed based on formula (5) where:

$$f(x) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \eta_1 x_4 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \eta_2 x_6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad g(x) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & \alpha_1 x_4 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & \alpha_2 x_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix} \tag{9}$$

By selecting $x=\begin{bmatrix}\varphi & \dot{\varphi} & \theta & \dot{\theta} & \Psi & \dot{\Psi}\end{bmatrix}^T$, one can choose four fuzzy variables (10) dependent on the state.

$$\begin{cases} z_1 = \eta_1 x_4; z_2 = \eta_2 x_6 \\ z_3 = \alpha_1 x_4; z_4 = \alpha_2 x_2 \end{cases} \tag{10}$$

One must remind that the choice of fuzzy variables is not unique. For instance, z_1 and z_3 can be chosen as a single variable that depend on x_4. Given the physical properties of the drone, the bounds of the decision variables are as follows.

$$\begin{cases} z_1 max = -2\pi\eta_1;\ z_1 min = 2\pi\eta_1 \\ z_2 max = 2\pi\eta_2;\ z_2 min = -2\pi\eta_2 \\ z_3 max = 2\pi\alpha_1;\ z_3 min = -2\pi\alpha_1 \\ z_4 max = -2\pi\alpha_2;\ z_4 min = 2\pi\alpha_2 \end{cases} \tag{11}$$

The membership functions can be computed using relation (7). Finally, the drone can be described using 16 local models, $\{A_1 \ldots A_{16}\}$ and $\{B_1 \ldots B_{16}\}$ as given below.

$$A_1 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & -2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & 2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix} \quad A_9 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_9 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & -2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & 2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & -2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & -2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix} \quad A_{10} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_{10} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & -2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & -2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & 2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & 2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix} \quad A_{11} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_{11} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & 2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & 2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & 2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & -2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix} \quad A_{12} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_{12} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & 2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & -2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix}$$

$$A_5 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_5 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & -2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & 2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix} \quad A_{13} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_{13} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & -2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & 2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix}$$

$$A_6 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_6 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & -2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & -2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix} \quad A_{14} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_{14} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & -2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & -2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix}$$

$$A_7 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_7 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & 2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & 2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix} \quad A_{15} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_{15} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & 2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & 2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix}$$

$$A_8 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_8 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & 2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & -2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix} \quad A_{16} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -2\pi\eta_1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 2\pi\eta_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad B_{16} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \beta_1 & 0 & 0 & 2\pi\alpha_1 \\ 0 & 0 & 0 & 0 \\ 0 & \beta_2 & 0 & -2\pi\alpha_2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta_3 & 0 \end{bmatrix}$$

3.3 Stability Analysis

For a given nonlinear system described by TS models, Chadli & Borne (2012) presented the following theorem as a sufficient stability condition for multiple models.

Theorem: The TS model (3) is asymptotically stable, if there exists a matrix P that is symmetric and positive definite, such that the LMIs (12) are verified:

$$A_i^T P + PA_i \leq 0 \qquad \forall i \in \{1 \dots N\} \tag{12}$$

Once matrix P is obtained, the assymptotic stability can be confirmed by deriving the quadratic Lyapunov function $V(x(t)) = x(t)^T Px(t)$. The existence of a matrix P that satisfies the LMIs (12) depends on two conditions. The first one is that each local model is stable (i.e. A_i has negative eigenvalues). The second condition is the existence of a matrix P that is common to all submodels.

In this study, the LMIs (12) were solved using a specific MATLAB toolbox, which lead to the required matrix P in (13):

$$\mathrm{P} = \begin{bmatrix} 2.8378 & 0 & 0 & 0.6159 & 0 & -0.0004 \\ 0 & 0.0029 & 0 & 0 & 0 & 0.0656 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1337 & 0 & 0 \\ 0.6159 & 0 & 0 & 2.9253 & 0 & 1 \\ -0.0004 & 0.0656 & 0 & 0 & 0 & 8.0019 \end{bmatrix} \tag{13}$$

4. Fault Accommodation

This section proposes a FTC for the vehicle described by TS model. The accommodation procedure is based on the LQR approach. The formulation of the LQR problem, in the infinite horizon case, can be expressed in (14). For a given linear system in the state space form, the objective is to find the optimal control $u(t)$ that minimizes the cost function J. Q and R are called "weighting matrices" and are selected according to the desired action on the state or the input.

$$\begin{cases} \dot{x} = Ax + Bu \\ J = \int_0^\infty \left(x^T Qx + u^T Ru \right) dt \\ u(t) = -Fx(t) \end{cases} \tag{14}$$

The feedback gain is obtained using (15):

$$F = R^{-1} B^T K \tag{15}$$

where K represents the solution of the Riccati equation (16):

$$A^T K + KA - KBR^{-1} B^T K + Q = 0 \tag{16}$$

For a system described by TS model (3), the appearance of an actuator fault at time t_f affects the different input matrices B_i. For each submodel (A_i, B_i), equation (16) is solved and the new FTC u_{ci} can be obtained using the relation in (14). Considering the closed-loop TS system (17), the asymptotic stability of the entire model can be proved using Lyapunov approach (Chadli, 2002):

$$\dot{x}(t) = \sum_{i=1}^{N} \mu_i \left(z(t) \right) \left(A_i - B_i F_i \right) x(t) \tag{17}$$

A transition process is used to monitor the different intervals (18) that describe the system behaviour:

$$\begin{cases} \dot{x}(t) = \sum_0^n \mu_i(z(t)) \left(A_i x(t) + B_{ni} u_{ni}(t) \right), t \in [t_0, t_f[\\ \dot{x}(t) = \sum_0^n \mu_i(z(t)) \left(A_i x(t) + B_{fi} u_{ni}(t) \right), t \in [t_f, t_c[\\ \dot{x}(t) = \sum_0^n \mu_i(z(t)) \left(A_i x(t) + B_{fi} u_{ci}(t) \right), t \in [t_c, t_\infty[\end{cases} \tag{18}$$

where n, f, c denote "nominal", "fault" and "correction", respectively.

In this study, it is assumed that the diagnosis procedure is already performed. Therefore, the FTC is applied with the assumption that the fault model is known.

5. Simulation Results

The current section presents the simulation results of the vehicle during the different time intervals, as described in (18).

5.1 Nominal Conditions

In order to validate the TS model developed in section 3, simulation is performed using the 16 local matrices $\{A_1 \dots A_{16}\}$ and $\{B_1 \dots B_{16}\}$. The outputs of the system are the different Euler angles and their respective derivatives $x = \begin{bmatrix} \varphi & \dot{\varphi} & \theta & \dot{\theta} & \Psi & \dot{\Psi} \end{bmatrix}^T$. The weighting matrices are chosen for simplicity: $R = I_4$ and $Q = I_6$. In this first interval, $t \in [t_0, t_f[$ (see equation

(18)), it is considered that $\{B_1..B_{16}\} = \{B_{n1}..B_{n16}\}$ and, for each submodel, a gain F_{ni} is computed using (15), leading to the control u_{ni}. Starting from the initial conditions $CI = [0.5; 0; 0.02; 0; -0.5; 0]$ and using a fixed step $Te = 0.005$, one gets the results shown in Figure 1 and Figure 2.

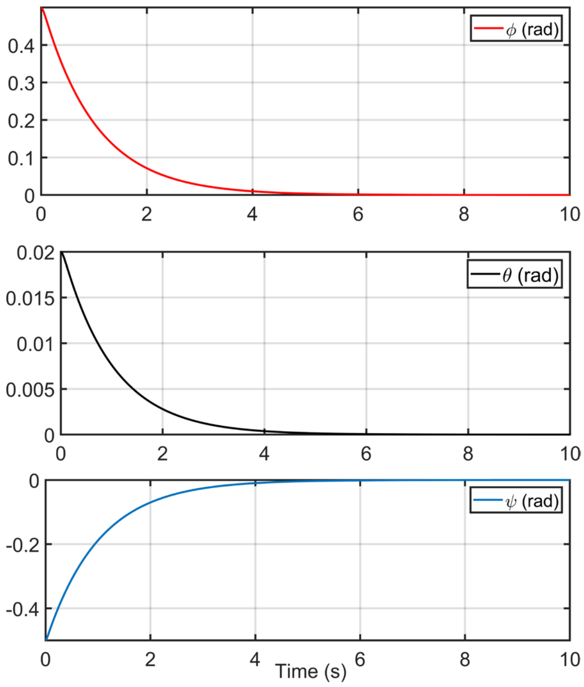

Figure 1. Orientation angles in nominal conditions

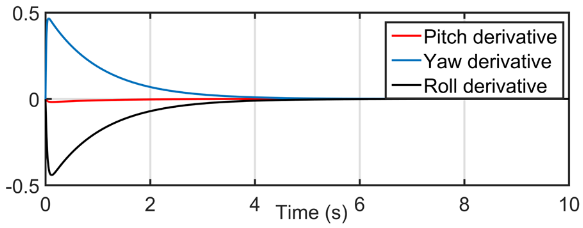

Figure 2. Angles derivatives in nominal conditions

Figure 3 illustrates the time evolution of the weighting functions $\mu_i(z(t))$ (denoted in the figure by h_i), that are clearly nonlinear and obey to the convexity property (4). Both Euler angles and their derivatives converge successfully to the equilibrium in 4*s*, which proves the validity of the model, as well as the adopted control.

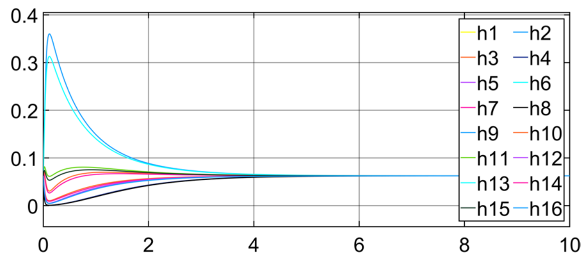

Figure 3. Weighting functions

5.2 Fault Conditions

For the second interval $t \in [t_f, t_c[$ from equation (18) an actuator fault occurs at $t_f = 4s$. This fault affects the four propellers of the drone and is modelled using an additive matrix, added to each submodel, such that:

$$B_{fi} = B_{ni} + B_a,$$

$$B_a = \begin{bmatrix} 638 & 0 & 0 & 0 \\ 0 & 638 & 0 & 0 \\ 0 & 0 & 638 & 0 \\ 0 & 50 & 0 & 400 \\ 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 10 \end{bmatrix} \tag{19}$$

For the drone, this fault can be considered as an actuator bias. Since the physical inputs of the system are the motors' speed, adding a constant value will act like an undesired acceleration command that will affect the intended control. Consequently, the motors will spin with inappropriate speed, causing the drone to change its normal direction. In reality, this can be caused by different scenarios such as sensor bias, mechanical issues in the motor itself or electronic component issues in the driver circuit of the motor.

It must be stressed that matrix B_a is selected using a trial-and-error method. Although this method is time consuming and might not lead to the most accurate result, it still has several advantages such as the simplicity to implement. Also, it gives quick initial results by achieving a basic level of fault representation. The main criterion is to produce a significant deviation or divergence in the angular position of the vehicle. Once the fault occurs, simulation is performed using the new input matrices B_{fi} (see equation (19)), but with the same control u_{ni}, in order to analyse the impact of the defection on the vehicle.

Figure 4 and Figure 5 showcase that the vehicle's orientation angles witness a significant deviation and diverge completely from the nominal position at $t = 6.5s$.

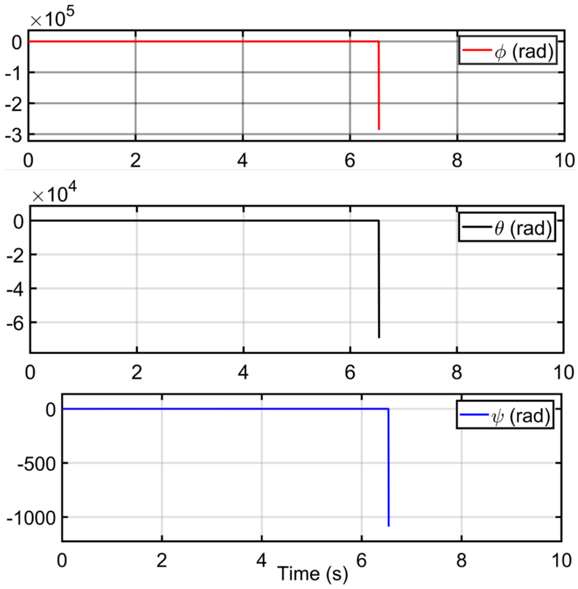

Figure 4. Orientation angles in failure conditions

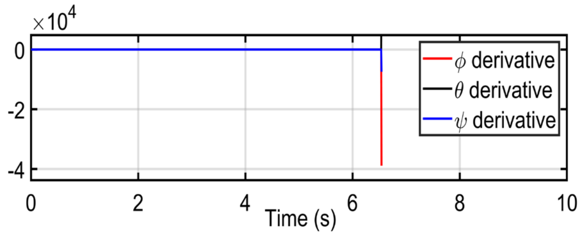

Figure 5. Angles derivatives in failure conditions

5.3 Correction with FTC

In the last interval $t \in \left[t_c, t_\infty\right[$ from equation (18) the FTC u_{ci} is computed for each model, by solving the Riccati equation (16), while considering the matrices B_{fi} (see equation (19)). Then, it is applied at the instance $t_c = 5.5s$.

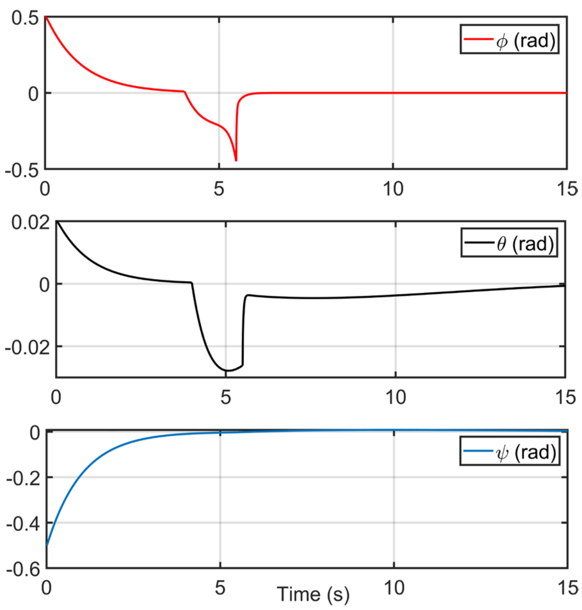

Figure 6. Orientation angles with FTC

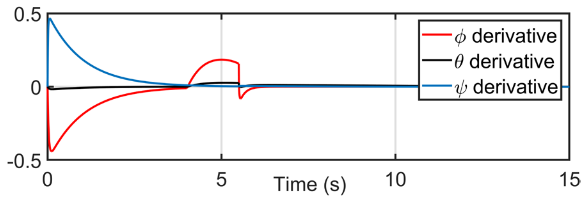

Figure 7. Angles derivatives with FTC

Figure 6 and Figure 7 demonstrate the efficiency of the FTC approach. Starting from the instance of correction t_c, all orientation angles and their derivatives converge again to the equilibrium position and the vehicle resumes again its normal behaviour.

6. Conclusion

This paper investigates the problem of FTC of a quad-rotor aerial vehicle using TS models. A new model of the vehicle was proposed and built, for the first time, using TS models. The main objective was to stabilize the vehicle orientation when faults affect the four propellers simultaneously. These faults caused the Euler angles to diverge, which significantly affected the stability of the vehicle. An accommodation strategy was then proposed using LQR technique and based on the developed-sixteen submodels. The proposed FTC had successfully brought the vehicle to the equilibrium position. The stability of the new model was proved using Lyapunov theory and LMIs.

The validity of the developed new model as well as the FTC were proved through varied simulation. Moreover, the main advantages of the polytopic transformation used in TS modelling are to avoid the generation of approximation errors and to reduce the number of local models, compared to the linearization method (Chadli & Borne, 2012). It should be noted that reducing the number of local models leads to the reduction of the LMIs number, which consequently increases the chances of finding a Lyapunov solution for stability analysis.

In forthcoming studies, it would be interesting to develop a comprehensive framework that includes both fault diagnosis and FTC. Artificial Neural Networks can be investigated to estimate faults in nonlinear systems modelled by TS. Additionally, this research can be extended to achieve complete control of the drone, including its altitude and position regulation.

REFERENCES

AL-Dosari, K., Hunaiti, Z. & Balachandran, W. (2023) Systematic Review on Civilian Drones in Safety and Security Applications. *Drones*. 7(3), 210. doi: 10.3390/drones7030210.

Azeem, O., Hamayun, M. T., Ijaz, S. & Tariq, K. (2024) Output Feedback Fault Tolerant Control for Linear Parameter Varying Plants Considering Imperfect Fault Information. *IEEE Access*. 12, 12257-12270. doi: 10.1109/ACCESS.2024.3354847.

Bouguila, N., Jamel, W., Khedher, A. & Ben Othman, K. (2013) Multiple observer design for a non-linear Takagi–Sugeno system submitted to unknown inputs and outputs. *IET Signal Processing*. 7(8), 635–645. doi: 10.1049/iet-spr.2012.0398.

Bresciani, T. (2008) *Modelling, Identification and Control of a Quadrotor Helicopter*. Master thesis. Lund University.

Chadli, M. (2002) *Stabilité et commande de systèmes décrits par des multimodèles: Approche LMI*. PhD thesis. Institut National Polytechnique de Lorraine.

Chadli, M. & Borne, P. (2012). *Multimodèles en automatique*. *Outils avancés d'analyse et de synthèse*. Hermes-Lavoisier, p. 192.

Dalwadi, N., Deb, D. & Ozana, S. (2022) Rotor Failure Compensation in a Biplane Quadrotor Based on Virtual Deflection. *Drones*. 6(7), 176. doi: 10.3390/drones6070176.

Elleuch, I., Khedher, A. & Othman, K. B. (2018) Design of a proportional integral observer based on sliding mode principle for uncertain Takagi-Sugeno fuzzy systems: applications to a turbo-reactor. *International Journal of Automation and Control*. 12(2), 179. doi: 10.1504/IJAAC.2018.090809.

Feng, Z., Zhou, Z., Yang, R., Ban, X. & Hu, C. (2023) Fault-tolerant control based on belief rule base expert system for multiple sensors concurrent failure in liquid launch vehicle. *Nonlinear Dynamics*. 111(5), 4357–4373. doi: 10.1007/s11071-022-08061-7.

Jamel, W., Bouguila, N., Khedher, A. and Ben Othman, K. (2010a) Observer design for nonlinear systems represented by Takagi-Sugeno models. *WSEAS Transactions on Systems.* 9(7), 804–813.

Jamel, W., Khedher, A., Bouguila, N. & Ben Othman, K. (2010b) State Estimation via Observers with Unknown Inputs: Application to a Particular Class of Uncertain Takagi-Sugeno Systems. *Studies in Informatics and Control*. 19(3), 219-228.

Jamel, W., Khedher, A. & Othman, K. B. (2017) Observer design and active fault tolerant control for Takagi-Sugeno systems affected by sensors faults. *International Journal of Modelling, Identification and Control*. 27(1), 22. doi: 10.1504/IJMIC.2017.10003318.

Khedher, A., Elleuch, I. & Ben Othman, K. (2022) Observer Desing for Non Linear TakagiSugeno Fuzzy Systems. Application to Fault Tolerant Control. In: Naifar, O. & Ben Makhlouf, A. (eds.) *Advances in Observer Design and Observation for Nonlinear Systems. Studies in Systems, Decision and Control*. vol 410. Cham, Switzerland, Springer International Publishing, pp. 101–133. doi: 10.1007/978-3-030-92731-8_7.

Mehran, K. (2008) Takagi-Sugeno Fuzzy Modeling for Process Control. *Industrial Automation, Robotics and Artificial Intelligence (EEE8005)*. 262, 1–31.

Merheb, A.-R., Noura, H. & Bateman, F. (2014) Active fault tolerant control of quadrotor UAV using Sliding Mode Control. In: *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, *27-30 May 2014, Orlando, FL, USA*. New Jersey, SUA, Institute of Electrical and Electronics Engineers (IEEE). pp.156–166. doi: 10.1109/ICUAS.2014.6842251.

Mlayeh, H. & Ben Othman, K. (2022a) Attitude control of a drone subject to disturbance and structural fault. In: *2022 IEEE 21st international Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*. pp. 232–236.

Mlayeh, H. & Ben Othman, K. (2022b) Nonlinear Accommodation of a DC-8 Aircraft Affected by a Complete Loss of a Control Surface. *Studies in Informatics and Control*. 31(3), 107–116. doi: 10.24846/v31i3y202210.

Mlayeh, H., Ghachem, S., Nasri, O. & Ben Othman, K. (2021) Stabilization of a Quadrotor Vehicle Using PD and Recursive Nonlinear Control Techniques. *International Review of Aerospace Engineering (IREASE)*. 14(4), 211. doi: 10.15866/irease.v14i4.19739.

Mlayeh, H. & Khedher, A. (2023a) Nonlinear Fault Tolerant Control of an Aerial Four Rotor Vehicle. In: *2023 IEEE International Conference on Artificial Intelligence & Green Energy (ICAIGE)*, *12-14 October 2023, Sousse, Tunisia*. New Jersey, SUA, Institute of Electrical and Electronics Engineers (IEEE). pp. 1–5. doi: 10.1109/ICAIGE58321.2023.10346282.

Mlayeh, H. & Khedher, A. (2023b) Overview on Active Fault Tolerant Control. In: Makhlouf, B., Hammami, M. A. & Naifar, O. (eds.) *State Estimation and Stabilization of Nonlinear Systems. Studies in Systems, Decision and Control*. vol 491. Cham, Switzerland, Springer Nature Switzerland, pp. 221–239. doi: 10.1007/978-3-031-37970-3_12.

Mlayeh, H. & Khedher, A. (2024) Trajectory-Tracking Problem of an Airplane with Damaged Control Surfaces. *Journal of Electrical Engineering & Technology*. doi: 10.1007/s42835-023-01772-w.

Saied, M., Kassem, M., Hussein Mazeh, Shraim, H. & Francis, C. (2023) Fault Tolerance Evaluation of Model-Free Controllers with Application to Unmanned Aerial Vehicles. *Drone Systems and Applications*. 11, 1–10. doi: 10.1139/dsa-2022-0048.

Saied, M., Shraim, H. & Francis, C. (2023) A Review on Recent Development of Multirotor UAV Fault-Tolerant Control Systems. *IEEE Aerospace and Electronic Systems Magazine*. 1–30. doi: 10.1109/MAES.2023.3327697.

Sonugur, G. (2023) A Review of quadrotor UAV: Control and SLAM methodologies ranging from conventional to innovative approaches. *Robotics and Autonomous Systems*. 161, 104342. doi: 10.1016/j.robot.2022.104342.

Zhang, B., Sun, X., Liu, S. & Deng, X. (2022) Distributed fault tolerant model predictive control for multi-unmanned aerial vehicle system. *Asian Journal of Control*. 24(3), 1273-1292. doi:10.1002/asjc.2521.

Zhang, Y., Wang, X. & Wang, Z. (2023) Adaptive multiple model control for nonlinear constrained systems with a novel barrier Lyapunov function. *International Journal of Robust and Nonlinear Control*. 33(11), 6533-6550. doi: 10.1002/rnc.6692.

Zhou, W., Li, B., Sun, J., Wen, C.-Y. & Chen, C.-K. (2019) Position control of a tail-sitter UAV using successive linearization based model predictive control. *Control Engineering Practice*. 91, 104-125. doi: 10.1016/j.conengprac.2019.104125.