

Article

# Fuzzy-Based Hybrid Control Algorithm for the Stabilization of a Tri-Rotor UAV

Zain Anwar Ali <sup>1,\*</sup>, Daobo Wang <sup>1</sup> and Muhammad Aamir <sup>2</sup>

<sup>1</sup> College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China; dbwangpe@nuaa.edu.cn

<sup>2</sup> Electronic Engineering Department, Sir Syed University of Engineering and Technology, Karachi 75300, Pakistan; muaamir5@yahoo.com

\* Correspondence: zainanwar86@hotmail.com; Tel.: +86-1301-693-1051

Academic Editor: Felipe Gonzalez Toro

Received: 3 February 2016; Accepted: 28 April 2016; Published: 9 May 2016

**Abstract:** In this paper, a new and novel mathematical fuzzy hybrid scheme is proposed for the stabilization of a tri-rotor unmanned aerial vehicle (UAV). The fuzzy hybrid scheme consists of a fuzzy logic controller, regulation pole-placement tracking (RST) controller with model reference adaptive control (MRAC), in which adaptive gains of the RST controller are being fine-tuned by a fuzzy logic controller. Brushless direct current (BLDC) motors are installed in the triangular frame of the tri-rotor UAV, which helps maintain control on its motion and different altitude and attitude changes, similar to rotorcrafts. MRAC-based MIT rule is proposed for system stability. Moreover, the proposed hybrid controller with nonlinear flight dynamics is shown in the presence of translational and rotational velocity components. The performance of the proposed algorithm is demonstrated via MATLAB simulations, in which the proposed fuzzy hybrid controller is compared with the existing adaptive RST controller. It shows that our proposed algorithm has better transient performance with zero steady-state error, and fast convergence towards stability.

**Keywords:** Unmanned Aerial Vehicle; Tri-Rotor UAV; RST controller; fuzzy hybrid controller

## 1. Introduction

One of the best inventions of today's era is the small flying machine commonly called a UAV. This research is dedicated to such types of UAVs, which are commonly used in the monitoring of disaster management and military operations, as well as small indoor activities [1–3]. The research on UAVs is based on the different knowledge banks of aeronautics, signal processing, and control automation. For this research, multiple hardware-based tests are performed to design the best flying machines with precise control mechanisms.

The current trend is focused on the design of advanced, lightweight, and perfect UAVs that can be operated in any disastrous situations over remote areas. UAVs are classified as either fixed-wing or rotary wing [4]. Rotor-based UAVs are multiple input and multiple output (MIMO) multivariable systems [5]. Rotorcraft have a great advantage over fixed-wing aircraft with respect to various applications, like vertical takeoff and vertical landing (VTOL) capability and payloads. Rotor-based UAVs include many types, such as bi-rotor, tri-rotor, quad-rotor and hex-rotor [6]. Moreover, a tri-rotor UAV with VTOL ability is considered in this paper.

Real-world application of UAVs require intense hardware testing. Before the experimental testing of our proposed algorithm in the real world, we have to simulate the numerical nonlinear simulations for the Euler angles, control commands, rotational velocities, and translational velocities [7]. In this research, our main concern is to rectify the error which occurs in a yaw moment due to the unpaired

reaction of the rotors, thereby producing torque. Brushless direct current (BLDC) motors are installed in the triangular frame of the tri-rotor craft to nullify the tilt angle moment.

The dynamics of the UAV are highly nonlinear and multi-variable, with a lot of parameter uncertainties, many effects to which a potential controller has to be robust. The aerodynamics of the actuator blades (flapping of blade and propeller), inertial torques (angular speed of propellers), and gyroscopic effects (which change the orientation of the UAV) are found in [8]. The redundancy in the rotors of a UAV formulates them towards a set of partial collapses. Although the maneuverability and performance will probably be condensed in the case of such a collapse, it is required that a controller stabilizes the system and tolerates reduced mode functions, such as safe arrival, steady hover, etc. [9,10].

Previously, many control methods were used for the stabilization of UAVs, including the conventional proportional integral derivative (PID) controller, fuzzy controller, adaptive controller, and so on [11]. For controlling the parameters of a UAV an adaptive controller has a capability to give good performance in the presence of model and parametric uncertainties, while MRAC is concerned with the vibrant reaction of the controlled system to asymptotic convergence. It follows the reference system in spite of parametric model uncertainties in the system [12].

In [13] the proposed MRAC for controlling the dynamics of a quad-rotor in the presence of actuator uncertainties was considered to enhance an existing linear controller, offering autonomous waypoint following. The stability of the adaptive controller was ensured by the Lyapunov theorem and, in a nonlinear structure, the algorithm is applied for indoor flight test.

In [14] the hybrid control scheme to fault tolerant control (FTC) for a quad-rotor aircraft in the presence of faults in their rotors during the flight have been explored and tested on the MRAC algorithm and a gain-scheduled PID (GS-PID) control. MRAC and GS-PID are used in collaboration with a linear quadratic regulator (LQR) to control the attitude of the UAV. MRAC is based on MIT rules for controlling the height and other parameters of a Qball-X4 Quad-Rotor aircraft.

Takagi-Sugeno fuzzy rules were previously used in [15,16] to control the nonlinear behavior of the vehicle. On the other hand, in [17], a twin controller approach that consists of a backstepping controller to control the nonlinear dynamics of the system and linguistic logic rules of a fuzzy logic controller (Mamdani) is used to control the attitude of tilt of a tri-rotor UAV. In [18] a dual controller approach with an adaptive fuzzy sliding mode controller is used to control the mini UAV, in which sliding mode control is utilized to control the nonlinear behavior of the UAV, and then fuzzy logic rules are implemented on it. The hybrid controller approach was also addressed in [19] in which a fuzzy-PID controller with a PSO algorithm is applied on tri-rotor dynamics.

However, in this paper, we proposed a fuzzy hybrid controller consisting of a RST with MRAC, based on MIT rules working as a main controller in the model to deal with the nonlinear system. We compare the performance of our proposed fuzzy hybrid controller with the robust adaptive RST controller of [20].

Moreover, the adaptive gains of the RST controller are (*i.e.*, regulation gain " $G_R$ ", pole-placement gain  $G_S$ , and tracking gain " $G_T$ ") tuned by a fuzzy logic controller (Mamdani technique). This means that our main controller is a RST with MRAC based on MIT rules, and for the tuning purpose we use the Mamdani fuzzy logic controller. We have to implement the gains of RST by adding fuzzy logic between uniform scales of membership functions. It shows the best results as compared to the adaptive RST controller [21,22].

In this paper, we are incorporating RST controller with our proposed system in two separate ways. First, the system is undergoes through Robust adaptive RST controller, after that we use Fuzzy-Hybrid based MIT algorithm and then conclude the results by taking the difference of robustness.

The core contributions in this research are as follows: (1) a novel fuzzy-based adaptive robust RST controller is derived by accumulating the MIT rule in the control law to remove the model disturbance and to derive the steady-state error to zero; (2) the proposed controller uses the angular responses as an input control command, which shows more accurate and practical insight in the real

world; (3) in spite of the model disturbance, the close loop system error converges to zero, proved in Theorem 2; and (4) lastly, the polynomial characteristic solution is based on the Diophantine equation while least square estimation is used to check system stability and proved in Theorem 3.

The breakup of this paper is structured as follows. The system modeling, dynamic representation of a tri-rotor UAV, and main engine model is discussed in Section 2. Section 3 demonstrates the dynamic control strategies and the control algorithm of the UAV. Moreover, the simulation results and discussions are discussed in Section 4. Lastly, Section 5 states the conclusions.

## 2. System Model and Preliminaries

### 2.1. Tri-Rotor Modeling

The equation of motion of a rigid body is defined by Newton's second law of motion [23,24]. Linear and angular forces change with respect to the timeframe, called the initial reference frame, in which the UAV has a similar velocity, force components, and moments, which are used to develop the six degrees of freedom nonlinear equations of motion. The nonlinear aerodynamic forces, aerodynamic moments, rotation motion, and translational motion of a UAV are defined by using differential Equations (1)–(4).

Equations of aerodynamic force:

$$\begin{aligned} F_X - mg \sin\theta &= m (\dot{u} + qw - rv) \\ F_Y + mg \cos\theta \sin\varphi &= m (\dot{v} + ru - pw) \\ F_Z + mg \cos\theta \cos\varphi &= m (\dot{w} + pv - qu) \end{aligned} \quad (1)$$

Equations of aerodynamic moments:

$$\begin{aligned} L &= I_x \dot{p} - I_{xz} \dot{r} + qr (I_z - I_y) - I_{xz} pq \\ M &= I_y \dot{q} + rp (I_x - I_z) + I_{xz} (p^2 - r^2) \\ N &= -I_{xz} \dot{p} + I_z \dot{r} + pq (I_y - I_x) + I_{xz} qr \end{aligned} \quad (2)$$

Rotational rates:

$$\begin{aligned} p &= \dot{\phi} - \dot{\psi} \sin\theta \\ q &= \dot{\theta} \cos\varphi + \dot{\psi} \cos\theta \sin\varphi \\ r &= \dot{\psi} \cos\theta \cos\varphi - \dot{\theta} \sin\varphi \end{aligned} \quad (3)$$

Euler angles and body angular velocities:

$$\begin{aligned} \dot{\theta} &= q \cos\varphi - r \sin\theta \\ \dot{\phi} &= p + q \sin\varphi \tan\theta + r \cos\varphi \tan\theta \\ \dot{\psi} &= (q \sin\varphi + r \cos\varphi) \sec\theta \end{aligned} \quad (4)$$

The four control commands of the tri-rotor UAV are (Col, Lat, Lon, Ped), which is similar to the conventional helicopter, in which Col is Collective, Lat is Lateral, Lon is Longitudinal, and Ped is pedal. (Col, Lat) are used to control the roll rate, (Lon) control the pitch rate and (Ped) controls the yaw rate of a UAV and tilt angle by using the parameter " $\alpha$ " [25,26]. (p, q, r) and (U, V, W) are the rotational velocity and translational velocity of the coordinate system. (L, M, N) and ( $\phi, \theta, \psi$ ) are the external moments and rotational angles of a fixed body frame.

A tri-rotor aerial vehicle exhibits many physical effects, like inertial torque, effects of aerodynamics, effects of gravity, effects of gyroscope and frictional effects, *etc.* In the presence of these physical effects it is quite difficult to design a controller which can easily handle all of the physical effects and stabilize the UAV in a fair amount of time, because it has six degree of freedom (6-DOF) with a highly-nonlinear, multivariable, under-actuated, strongly-coupled model with the rotors, as shown in Figures 1 and 2 taken from the design of Mohamed MK [27].

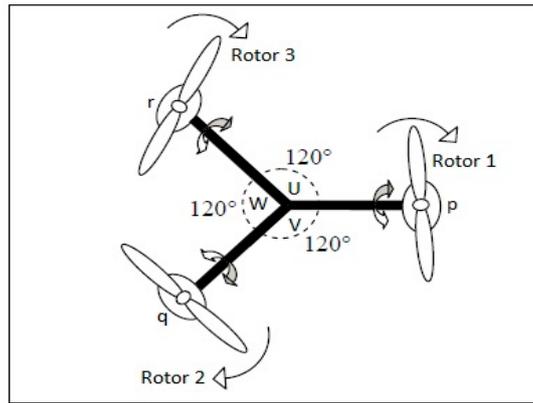


Figure 1. Top view of a tri-rotor UAV along with rotational and transitional rates.

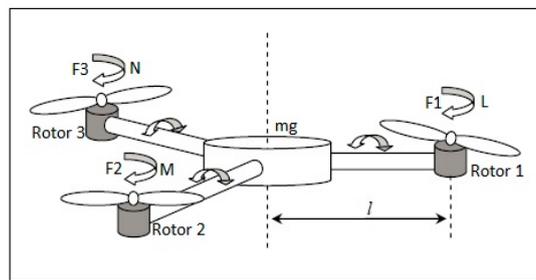


Figure 2. 3-D view of a tri-rotor UAV along with aerodynamic moment components.

The tri-rotor UAV, including translational and rotational subsystems, needs a vibrant strategy of nonlinear sequential control in the 6-DOF model. This research addresses error controlling in a tri-rotor aircraft by managing the torque produced by unpaired rotor reactions. To overcome this issue, implementation of different designs has been made with the help of BLDC motors, which actually control the nullifying angle. This method is useful for quicker turn by tilting the rotor’s axis.

2.2. Dynamic Representation of a Tri-Rotor UAV

The orientation of the UAV is explained by Euler angles having altitude, roll ( $\theta$ ), pitch ( $\varphi$ ), and yaw ( $\Psi$ ) control and it rotates along ( $x, y, z$ ) axes, respectively. The translational and rotational movement of the tri-rotor UAV into the dimensional space and dynamics of the rigid body derive from Newton’s law. Moreover, moments, forces, velocity components, and aerodynamic components are described in Table 1.

Table 1. Dynamic constants of a tri-rotor UAV.

$x, y, z$ Axis System	Roll ( $\phi$ )	Pitch ( $\theta$ )	Yaw ( $\psi$ )
Aerodynamic Force Components	X	Y	Z
Aerodynamic Moment Components	L	M	N
Translational Velocity	U	V	W
Angular Rates	p	q	r
Three-Axis Inertia	$I_x$	$I_y$	$I_z$

The overall system configuration is defined in Figure 2, where “L” is the distance from the center of the body frame to all three rotors, labelled as L1, L2, and L3. The rotor forces are f1, f2, f3, and the rotor torque is defined as  $\tau_1, \tau_2,$  and  $\tau_3,$  respectively. The angular velocity of the system is “ $\rho$ ”.

### 2.3. Main Engine (Electric Motors)

Brushless Direct Current (BLDC) motors can be used as a main engine of the UAV to achieve required electric propulsion in the system. BLDC motors found their application in the field of robotics, space crafts and medical devices, due to higher torque speed features, greater performance, minimum repairs and variable degree of speed [28]. Generally BLDC is more costly than simple DC motors, due to its better efficiency and reliability [29]. The electrical and mechanical equations of BLDC are given below.

$$V = Ri + (L - M) \frac{di}{dt} + E \quad (5)$$

$$E = K_e \omega_m F(\theta_e) \quad (6)$$

$$T = K_t i_a F(\theta_e) \quad (7)$$

$$T_e = J \frac{d^2 \theta_m}{dt^2} + \beta \frac{d\theta_m}{dt} \quad (8)$$

$$\theta_e = \frac{P}{2} \theta_m \quad (9)$$

$$\omega_m = \frac{d\theta_m}{dt} \quad (10)$$

In which,  $V$  is the applied voltage;  $R$  is the total resistance;  $E$  is the back electromagnetic force;  $L$  is the total inductance of the motor;  $M$  is the mutual inductance of the motor;  $\omega_m$  is the angular speed of the motor;  $\theta_e$  is the rotation angle of the motor;  $T_e$  is the electrical torque produced by the motor; and  $K_e$  is the back-EMF constant.

## 3. Designing of Controller

### 3.1. Tri-Rotor Dynamic Control Strategies

The flight dynamics and control strategy of a tri-rotor UAV is the same as traditional aircraft. The placement or orientation of flight dynamics control is a product of roll, pitch, and yaw. The control scheme of the UAV includes altitude, roll, pitch, yaw, and tilt angle control, having a major role for the displacement control the parameters of the system.

*Altitude Control Mechanism:* To achieve the desired altitude, the speed of all rotors must be same  $\rho_1 = \rho_2 = \rho_3$ . Increasing the speed of all rotors constantly will eventually raise the altitude of the UAV, such that the angular velocity of motors becomes equal.

*Roll Control Mechanism:* Roll control is achieved by regulating the front rotors speed. Decreasing rotor 1 velocity, rolls the system to the left and rotor 2 rolls the system towards right-side. Roll control has two conditions.

- i When moving clockwise roll  $\rho_2 > \rho_1 > \rho_3$ .
- ii When moving counter-clockwise roll  $\rho_2 < \rho_1 < \rho_3$ .

*Pitch Control Mechanism:* Regulating the speed of Rotor 1 and rearward rotors will change the pitch. The system pitches downward if the speed of Rotors 1 and 2 decreases while Rotor 3 speed keeps rising. If we decrease the speed of Rotor 3 and increase the speed of Rotors 1 and 2, the UAV pitch rises and fly flight reverses. Pitch control also has two conditions:

- i When nose-up  $\rho_2 = \rho_3 > \rho_1$ .
- ii When nose-down  $\rho_2 = \rho_3 < \rho_1$ .

*Yaw Control Mechanism:* The product of reaction torque and tilt angle " $\alpha$ " of Rotor 3 is used to control yaw movement. The value of the tilt angle is too small and helps to maneuver the UAV quickly. The yaw control condition is:  $\rho_1 = \rho_2 = \rho_3$ , with  $\alpha = 0$ .

### 3.2. Control Algorithm

In this section the overall control hierarchy is defined to control the attitude and altitude of the tri-rotor UAV. In which we assume the desired attitude variables are  $K_1 = \varnothing$  (roll),  $K_2 = \theta$  (pitch), and  $K_3 = \Psi$  (yaw), respectively, and  $K_T$  is a generalized term for the rotational angles of the system. Now the control algorithm for the attitude controlling of an actual system of the UAV is written as:

$$G_{KT_1}(j) = \frac{B(j)}{A(j)} = Y(t) \quad (11)$$

The degree of the system model numerator  $B(j)$  and denominator  $A(j)$  is found to be “1” and “3”. Where  $B = (B^+ * B^-)$ , such that  $B^+$  is variable and  $B^-$  is the constant.

Equation (12) presents the desired attitude response of our UAV model:

$$G_{KT_2}(j) = \frac{B_m(j)}{A_m(j)} = Y_m(t) \quad (12)$$

Now the gradient theory which was defined by the model reference adaptive control method is implemented:

$$\text{deg}A_c = 2 * \text{deg}A(j) - 1 = 5 \quad (13)$$

So, the RST controller will be second-ordered, and now  $\text{deg}A_0 = \text{deg}A(j) - \text{deg}B^+ - 1 = 1$ .

**Remark 1.** For the perfect system model  $A_m(j) = A(j)$  and  $B_m(j) = B(j)$ . Where  $A_C$  and  $A_{C_m}$  is the characteristic polynomials of the actual and desired system models and  $A_{C_m} = A_C$  for the constraints and will not affect the system to change the close loop poles of the model. Otherwise,  $A_{C_m}$  differs the system model mismatch.

**Fuzzy Logic Controller.** The vibrant performance of the fuzzy logic controller is described by the set of linguistic procedures that was established by a knowledgeable acquaintance in [30–32], in which the system “error” and variation in error rate are the input constraints, and RST are the variable outputs in our proposed controller. Formerly, RST can be improved online, using the set of rules, existing error, and variation in the error. In general, the error in the angle, combined with the mechanism output, increases. Furthermore, the controller performs well whether the error rises or the rate of error difference falls. It is important that, in the minor error phase, the earmarked control output is required to influence the change in error as soon as the error falls suddenly.

With the help of Equation (13), the degree of the proposed controller is found to be 5. The fuzzy logic-based adaptive RST controller is written in Equations (14)–(16):

$$FR = q^2 + (r_0 \times q) + r_1 \quad (14)$$

$$FS = (s_0 \times q^2) + (s_1 \times q) + s_2 \quad (15)$$

$$FT = (t_0 \times q \times A_0) + (t_1 \times A_0) \quad (16)$$

Put the values of  $FR$ ,  $FS$ ,  $FT$  in the above:

$$U_{FR_{KT}}(j) = FR(e, \frac{de}{dt})G_R \times e(j) \quad (17)$$

$$U_{FS_{KT}}(j) = FS(e, \frac{de}{dt})G_S \times \Sigma e(ij) \Delta(t) \quad (18)$$

$$U_{FT_{KT}}(j) = FT(e, \frac{de}{dt})G_T \times \frac{\Delta e(j)}{\Delta t} \quad (19)$$

where  $G_R$ ,  $G_S$  and  $G_T$  are the delayed control gains of the signal scaling factors.

**Remark 2.**  $G_{KT_1}(j)$  and  $G_{KT_2}(j)$ , are the system models. Moreover, the system diverges at a low phase angle with an unstable control signal. The proposed control algorithm is not better in this case. Previously, in [33], a better controller was proposed for this type of case, a model having a low phase angle lying in the complex plane with some damping issues. As a result, in [34], zero cancellation in the system is situated esoteric the area which will cancel.

**Remark 3.** The proposed controller design at sampling of time  $[NT]$ , which will remove the difficulties in the planning stage of controller implementation.

**Theorem 1.** After designing the control system, the next step is stability analysis, which shows the robustness level of the designed controller. The stability is highly vulnerable due to modeling errors called sensitivity. Therefore, model mismatch and model sensitivity are added in the system divergence between the actual and desired response depending upon the performance of the control system and its stability which will be taken from [34] Theorem 5.4, and examine the stability of the control system by using the model disturbance.

**Lemma.** The proposed equation illustrates the close loop system model having  $(NT)$  sampling period:

$$\frac{[Y^T(t)]^{NT}}{FR^{NT}} = \frac{[G_{KT_1}^T(j) G_{KT_2}^T(j) H^{T,NT}]^{NT} G_{KT_1}^{NT}(j) \left[ \frac{FT^{NT}}{FR^{NT}} \right]}{1 + [G_{KT_1}^T(j) G_{KT_2}^T(j) H^{T,NT}]^{NT} G_{KT_1}^{NT}(j) \left[ \frac{FS^{NT}}{FR^{NT}} \right]}$$

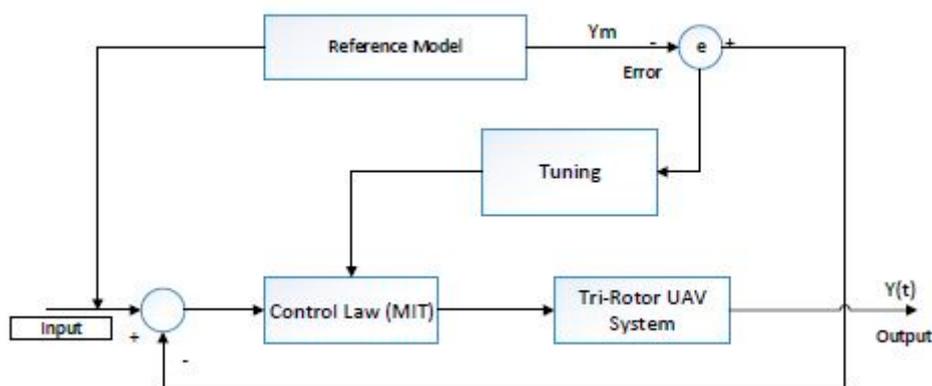
$H^{T,NT}$  is the conversion rate and will depend on the orientation of the desired signal.

**Remark 4.** The disturbance in the pole placement method is obsolete in [34]. Since the reference model, observer polynomial, and reference model disturbance act as constraints, they are proved in convergence analysis.

**Proof of Theorem 1.** The output of the unvarying model is considered in [35]:

$$Y^T(t) = G_{KT_1}^T(j) G_{KT_2}^T(j) H^{T,NT} \left[ G_{KT_1}^{NT}(j) \right]^{NT} \left[ \frac{FT^{NT}}{FR^{NT}} FR^{NT} - \frac{FS^{NT}}{FR^{NT}} [Y^T(t)]^{NT} \right]^T$$

**Remark 5.** Figure 3, gives complete work flow of proposed system using model reference adaptive control algorithm.



**Figure 3.** The model reference adaptive control system.

*Convergence:* Taking the input constraints of our controller signals gives the close loop error of the system model and unstable part of the disturbance. In other words, our proposed algorithm is able to stabilize the unstable part of the noise or disturbance model. The convergence at the desired value of parameters is done by an optimal control method based on the MIT rule to identify the errors.

**Remark 6.** For the proposed controller the adaptive gain is in the range of 0.15 to 5 and above this range the controller performance deteriorates.

$$e(j) = Y_{\text{actual}}(t) - Y_{m(\text{desired})} \quad (20)$$

The sensitivity derivative is presented in Equation (21):

$$Y(t) = \frac{B(j)T}{(A(j)R + B(j)S)} \quad (21)$$

$$U_c(t) = \frac{A(j)R + B(j)S}{B(j)T} * Y(t) \quad (22)$$

The convergence proof contracts distinctly with the constraints, firstly identifying the sensitivity derivative of all of the parameters ( $t_0, t_1, r_0, r_1, s_0, s_1, s_2$ ) of the controller.

The Diophantine equation represents as  $A_C = A(j)R + B(j)S$  and  $A_C = A_0A_m(j)$ ; therefore:

$$A(j)R + B(j)S = A_0A_m(j) \quad (23)$$

The MIT rule-based sensitivity derivative of Equation (20) is:

$$e(j) = \frac{B(j)T}{A(j)R + B(j)S} * U_c(t) - Y_m \quad (24)$$

**Theorem 2.** The model in Equation (11) with controller Equation (14–16) on the basis of system  $(A(j), B(j))$ , having UAV model disturbance, the error of close loop output Equation (24) goes to zero asymptotically if and only if  $A_{Cm} = A_C$ .

**Proof of Theorem 2.** The close loop output error Equation (24) is settled by  $A_C$  and compared with Equation (23).

1. If, and only if,  $A(j) = A_m(j)$  and  $B(j) = B_m(j)$ ; therefore, the close loop output error responds to  $e(j)$  and goes to zero asymptotically because  $A_C$  is stable in Equation (13) and  $A_{Cm}$  is supposed to be stable.

2. Contradiction: if  $A(j) \neq A_m(j)$  and  $B(j) \neq B_m(j)$ , the value is not cancelled by the close loop output error in Equation (24). The instability is in the denominator of  $e(j)$  if  $A_{Cm}$  is unstable. Hereafter, if  $A_C \neq A_{Cm}$  or  $A_{Cm}$  is unstable, then  $e(j)$  close loop output error raises, unbounded, and it is necessary for  $A_{Cm}$  to be stable to make the close loop output error zero.

Now put in the value of  $T = (t_0q + t_1)A_0$  in Equation (24):

$$e(j) = \frac{B(j)(t_0q + t_1)A_0}{A(j)R + B(j)S} * U_c(t) - (Y_m) \quad (25)$$

Therefore, w.r.t “ $t_0$ ”, the partial derivative of Equation (25) is:

$$\left\{ \begin{array}{l} \frac{\delta e(j)}{\delta(t_0)} = \frac{B(j)qA_0}{(A(j)R + B(j)S)} * U_c(t) \\ \frac{\delta e(j)}{\delta(t_0)} = \left( \frac{B(j)qA_0}{(A(j)R + B(j)S)} \right) * \frac{A(j)R + B(j)S}{B(j)T} * Y(t) \\ \frac{\delta e(j)}{\delta(t_0)} = \frac{A_0q}{T} * Y(t) \end{array} \right. \quad (26)$$

The partial derivative of Equation (25) w.r.t “ $t_1$ ” is:

$$\begin{cases} \frac{\delta e(j)}{\delta(t_1)} = \left( \frac{B(j)}{(A(j)R+B(j)S)} \right) * U_c(t) \\ \frac{\delta e(j)}{\delta(t_1)} = \left( \frac{B(j)}{(A(j)R+B(j)S)} \right) * \frac{A(j)R+B(j)S}{B(j)T} * Y(t) \\ \frac{\delta e(j)}{\delta(t_1)} = \frac{A_0}{T} * Y(t) \end{cases} \quad (27)$$

From Equation (24), replace the value of R: ( $R = (q)^2 + r_0q + r_1$ )

$$e(j) = \frac{B(j)T}{A(j)(q^2 + r_0q + r_1) + B(j)S} * U_c(t) - (Y_m) \quad (28)$$

Partial differentiate Equation (25) w.r.t “ $r_0$ ”:

$$\begin{cases} \frac{\delta e(j)}{\delta(r_0)} = - \left( \frac{B(j)TAS}{(A(j)R+B(j)S)^2} \right) * U_c(t) \\ \frac{\delta e(j)}{\delta(r_0)} = - \left( \frac{B(j)TAS}{(A(j)R+B(j)S)^2} \right) * \frac{A(j)R+B(j)S}{B(j)T} * Y(t) \\ \frac{\delta e(j)}{\delta(r_0)} = - \frac{AS}{A_0A_m(j)} * Y(t) \end{cases} \quad (29)$$

Partial differentiate Equation (25) w.r.t “ $r_1$ ”:

$$\frac{\delta e(j)}{\delta(r_1)} = - \frac{A(q)}{A_0A_m(j)} * Y(t) \quad (30)$$

Put the value of S,  $S = (s_0 * q^2 + s_1 * q + s_2)$  in Equation (24):

$$e(j) = \frac{B(j)T}{A(j)R + B(j)(s_0q^2 + s_1q + s_2)} * U_c(t) - (Y_m) \quad (31)$$

Now the Partial derivative of Equation (31) w.r.t “ $s_0$ ” :

$$\begin{cases} \frac{\delta e(j)}{\delta(s_0)} = - \frac{(B(j)^2 * q^2)}{(A(j)R+B(j)S)^2} * U_c(t) \\ \frac{\delta e(j)}{\delta(s_0)} = - \frac{(B(j)^2 * q^2)}{(A(j)R+B(j)S)^2} * \frac{A(j)R+B(j)S}{B(j)T} * Y(t) \\ \frac{\delta e(j)}{\delta(s_0)} = - \frac{B(j) * q^2}{A_0A_m(j)} * Y(t) \end{cases} \quad (32)$$

Now taking Partial derivative of Equation (31) w.r.t “ $s_1$ ”

$$\frac{\delta e(j)}{\delta(s_1)} = - \frac{B(j)q}{(A_m(j)A_0)} * Y(t) \quad (33)$$

Likewise, Partial derivative of the Equation (31) w.r.t “ $s_2$ ” gives

$$\frac{\delta e(j)}{\delta(s_2)} = - \frac{B(j)}{(A_m(j)A_0)} * Y(t) \quad (34)$$

By applying the MIT algorithm in the desired model of the system, where Equations (35) and (36) denote the cost function which is based on the MIT rule. Whitaker demonstrates the difference in system bounds as a function of the system error and the gradient of the system error with respect to the system constraints and takes the partial derivative of the gradient error with respect to its constraints. The constraints of the particular model with the initial estimate  $J(KT)$ , and the rate of change of speed

among the desired and actual model is taken from [36]. To minimize the error with respect to time so that the desired response is achieved requires an optimum control with cost function.

**Theorem 3.** In [20] the least square estimation  $J(\overline{KT})$  and their polynomial characteristic solution depends upon the Diophantine equation and stability of  $A_{Cm}$ .

**Proof of Theorem 3.** The solution also depends upon the Diophantine equation as well as the stability of  $A_{Cm}$ . By using Theorem 2:

1. If  $e(j) \rightarrow 0$  as  $l \rightarrow \infty$  and leading  $J(\overline{KT}) = 0$ , the solution gives the smallest positive value of cost function.
2.  $A_{Cm}$  is a contradiction, if it is not stable from Theorem 2;  $e(j) \rightarrow 0$  as  $l \rightarrow \infty$ , thus  $J(\overline{KT}) \neq 0$ . This is not optimal because, as seen in step 1 of the proof, there exists a solution that makes  $J(\overline{KT}) = 0$ .

$$J(KT) = \frac{1}{2}e^2(KT) \quad (35)$$

$$\frac{dKT}{(dt)} = -\gamma' \frac{1}{l_0} y_{(m)} e = -\gamma * y_{(m)} * e(j) \quad (36)$$

After that, apply the MIT rule to derivate control variables ( $s_0, s_1, s_2, r_0, r_1, t_0$ , and  $t_1$ ) and set in the controller gives

$$\frac{d(s_0)}{dt} = -\gamma_e \frac{\delta e}{\delta s_0} \frac{d(s_0)}{dt} = \frac{\gamma_e q^2 B(j) Y(t)}{A_m(j) A_0} \quad (37)$$

$$\frac{d(s_1)}{dt} = -\gamma_e \frac{\delta e}{\delta s_1} \frac{d(s_1)}{dt} = \frac{\gamma_e B(j) q Y(t)}{A_m(j) A_0} \quad (38)$$

$$\frac{d(s_2)}{dt} = -\gamma_e \frac{\delta e}{\delta s_2} \frac{d(s_2)}{dt} = \frac{\gamma_e B(j) Y(t)}{A_m(j) A_0} \quad (39)$$

$$\frac{d(r_0)}{dt} = -\gamma_e \frac{\delta e}{\delta r_0} \frac{d(r_0)}{dt} = \frac{\gamma_e A(j) q Y(t)}{A_m(j) A_0} \quad (40)$$

$$\frac{d(r_1)}{dt} = -\gamma_e \frac{\delta e}{\delta r_1} \frac{d(r_1)}{dt} = \frac{\gamma_e A(j) Y(t)}{A_m(j) A_0} \quad (41)$$

$$\frac{d(t_0)}{dt} = -\gamma_e \frac{\delta e}{\delta t_0} \frac{d(t_0)}{dt} = -\frac{\gamma_e B(j) q}{A_0} \quad (42)$$

$$\frac{d(t_1)}{dt} = -\gamma_e \frac{\delta e}{\delta t_1} \frac{d(t_1)}{dt} = -\frac{\gamma_e}{(A_m(j))} \quad (43)$$

Now, the main controller equation becomes:

$$U_{F-Hybrid_{\phi, \theta, \psi}} = \left( \frac{(F(T_0 q + T_1) A_0)}{F(q^2 + r_0 q + r_1)} \right) * (U_C(t)) - \left( \frac{F(S_0 q^2 + S_1 q + S_2)}{F(q^2 + r_0 q + r_1)} \right) Y(t)$$

The change in UAV orientation depends upon the rate of change of the control commands, which makes the system respond quite better towards stability by using our proposed algorithm.

The linguistics levels of the fuzzy hybrid controller are assigned as (BN) below negative, (SN) small negative, (ZR) zero, (SP) small positive, and (BP) big positive. The fuzzy logic controller if-then rules are defined in Tables 2–4 such that error “e” is the rotor speed having range in  $-10$  to  $+10$ , the derivative error range is  $-5$  to  $+5$ , and the output range of the fuzzy hybrid controller is  $0$  to  $1$  with  $R = 0.667$ ,  $S = 0.5$ , and  $T = 0.5$ , respectively.

**Table 2.** Fuzzy logic controller If-Then rule for “R”.

$\vec{de/dt}$	BN	SN	ZR	SP	BP
<i>error</i> ↓					
BN	ZR	SP	MP	MP	SP
SN	SP	SP	SP	MP	MP
ZR	SP	MP	MP	MP	MP
SP	SP	SP	MP	SP	SP
BP	ZR	SP	MP	LP	S

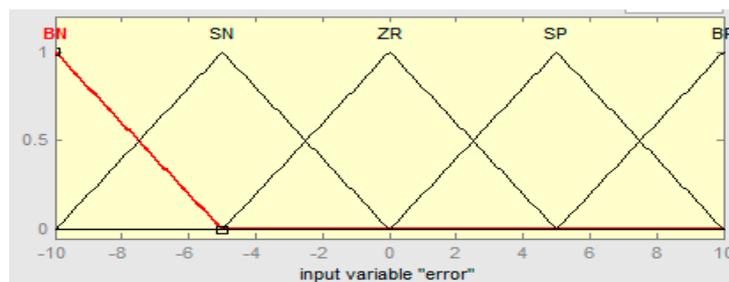
**Table 3.** Fuzzy logic controller If-Then rule for “S”.

$\vec{de/dt}$	BN	SN	ZR	SP	BP
<i>error</i> ↓					
BN	ZR	ZR	SP	BP	BP
SN	ZR	SP	SP	SP	BP
ZR	ZR	SP	SP	SP	BP
SP	ZR	SP	SP	SP	BP
BP	ZR	ZR	SP	SP	BP

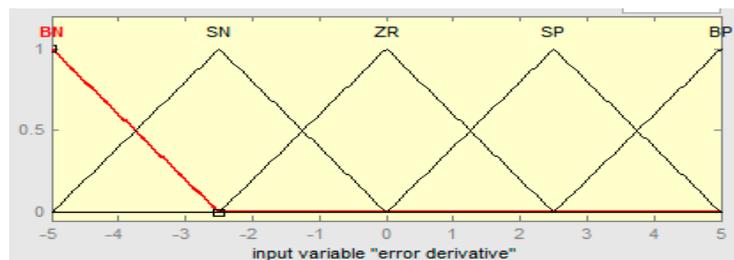
**Table 4.** Fuzzy Logic Controller If-Then rule for “T”.

$\vec{de/dt}$	BN	SN	ZR	SP	BP
<i>error</i> ↓					
BN	SP	SP	ZR	ZR	ZR
SN	BP	SP	SP	SP	BP
ZR	BP	SP	SP	SP	BP
SP	BP	SP	SP	SP	SP
BP	BP	BP	BP	SP	SP

The input error membership function of the fuzzy logic controller (FLC) is shown in Figure 4. Figure 5 shows the input derivative error of FLC. Moreover, Figures 6–8 show the output gains of RST-based FLC logic.



**Figure 4.** Input error membership function of fuzzy Logic.



**Figure 5.** Input derivative error membership function of fuzzy logic.

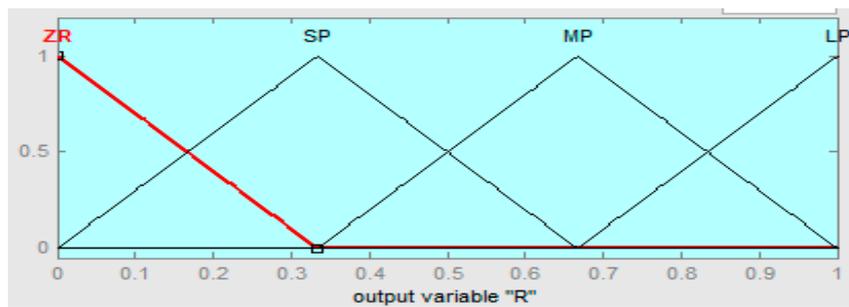


Figure 6. Regulation output gain membership function of fuzzy logic.

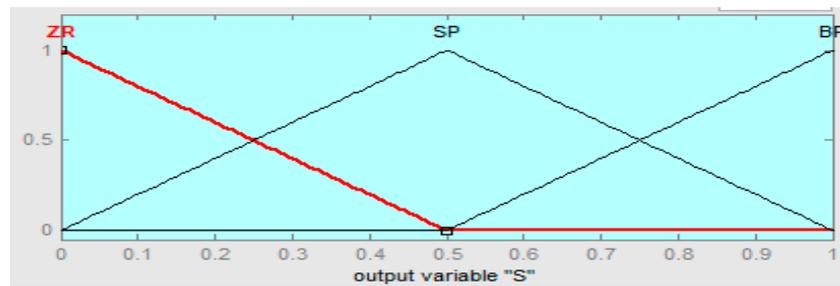


Figure 7. Pole-placement output gain membership function of fuzzy logic.

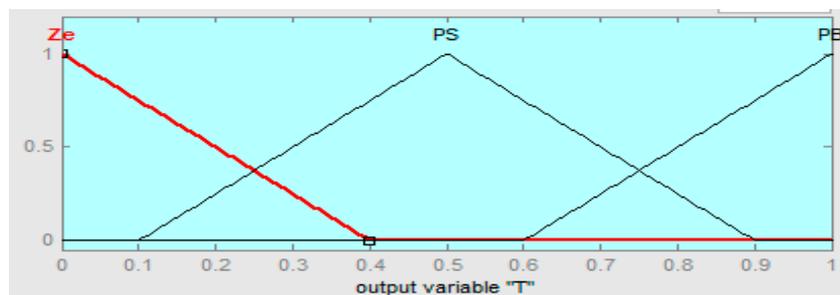


Figure 8. Tuning the output gain membership function of fuzzy logic.

#### 4. Simulation Results and Discussions

The validity and robustness of our proposed algorithm are presented in this section. Moreover, the nonlinear simulations for the stabilization of tri-rotor parameters are shown in Table 5. All of the simulations were done via Simulink, MATLAB. In all simulations, we compared our proposed fuzzy-hybrid controller with the adaptive RST controller of [20].

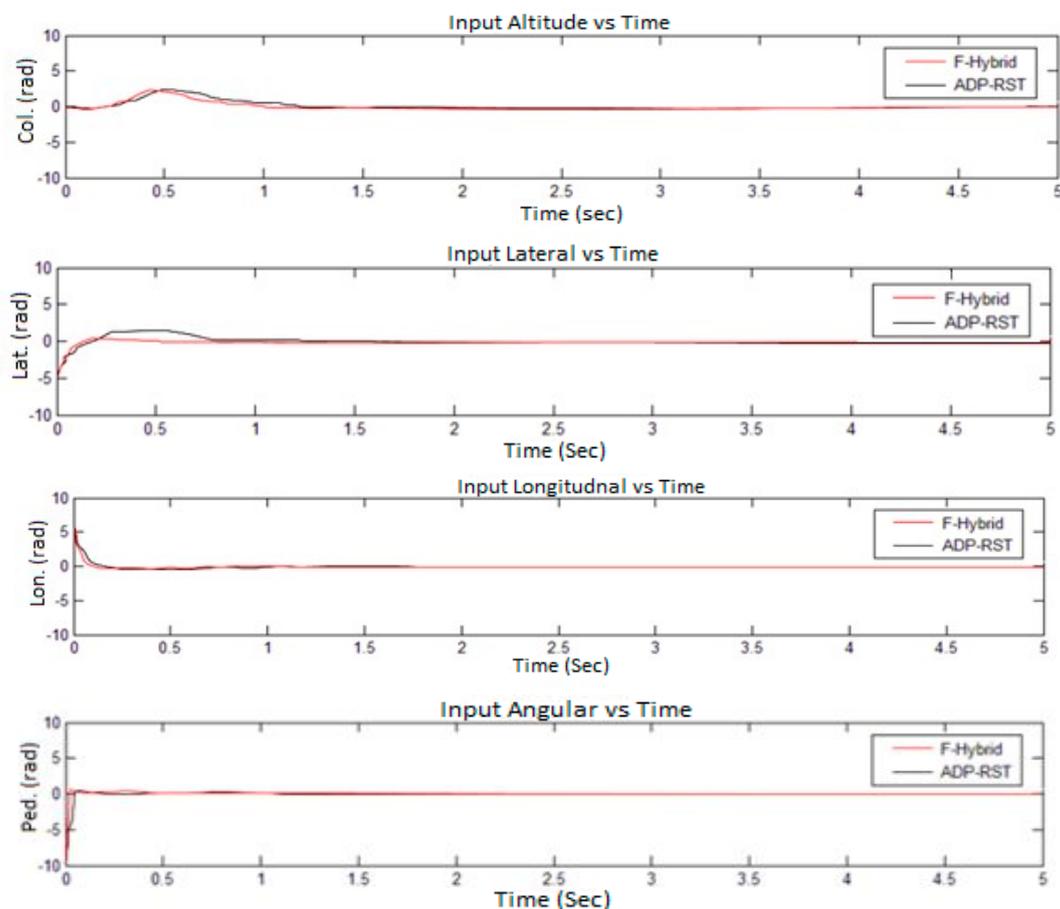
Table 5. Tri-rotor parameters.

Parameters	Values	Si Units
Ix	0.3105	kg·m <sup>2</sup>
Iy	0.2112	kg·m <sup>2</sup>
Iz	0.2215	kg·m <sup>2</sup>
l	0.3050	m
Mass	0.785	kg

Ideally, we can say that there are two subsystems of the UAV; one is rotational, while the other is a translational velocity response. The rotational subsystem is responsible for controlling the initial errors, thereby stabilizing the attitude of UAV and converges to zero. Pitch, roll, and yaw angles are also converging to zero to realize the hovering state. The rotational subsystem and Euler angles do not depend on translational components. However, the translational components depend on Euler angles.

The four control commands, *i.e.*, altitude, lateral, longitudinal, and angular, are shown in Figure 9. The settling and rise time are within one to one and a half seconds for each input channel along with zero overshoot. By comparing the adaptive RST controller control commands, it is not perfectly linear because in the adaptive RST controller the control commands do not perfectly converge to zero and have undershoot and overshoot, initially, which causes the UAV to dislocate from the desired position due to the presence of errors. However, our proposed method completely converges to zero and exactly reaches the perfect location as shown in Figure 10, which shows the actual reached height and vertical speed response of the UAV.

The Euler angular responses are shown in Figure 11; one can observe that initial angles are non-zero, but it will converge to zero, which means that it will perform the attitude hold control by comparing with the adaptive RST controller. Figure 12 shows the rotational rate responses and, by comparing the transitional rate responses given in Figure 13, observe that there are several “bumps” but ultimately converge to the desired value. In Figures 12 and 13 our proposed algorithm converges to zero at about one second and have no settling time error, but the adaptive RST controller settled at about two to two and a half seconds.



**Figure 9.** Comparison of F-Hybrid with adaptive RST for control commands.

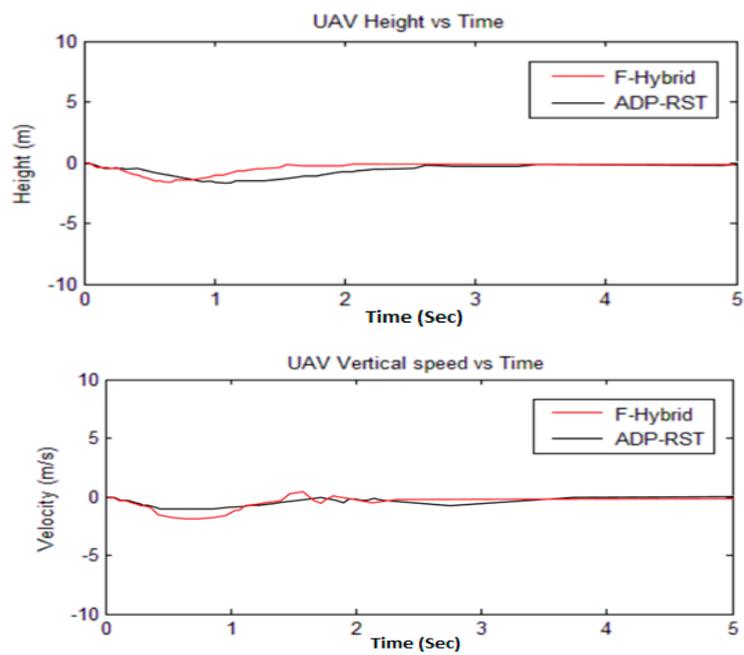


Figure 10. Comparison of F-Hybrid with adaptive RST for height and vertical speed.

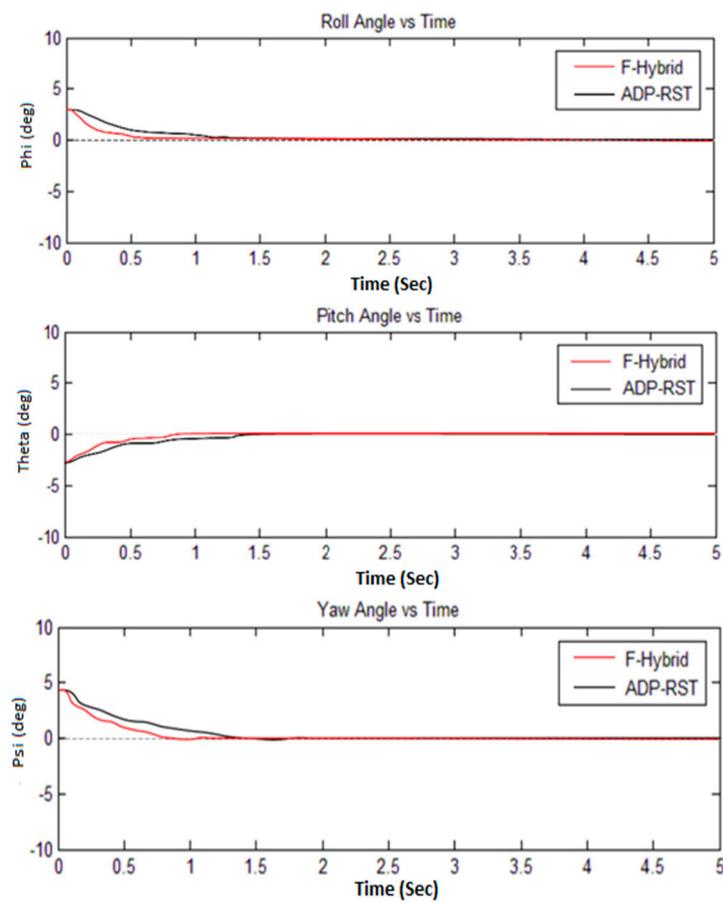


Figure 11. Comparison of F-Hybrid with adaptive RST for Euler angle response.

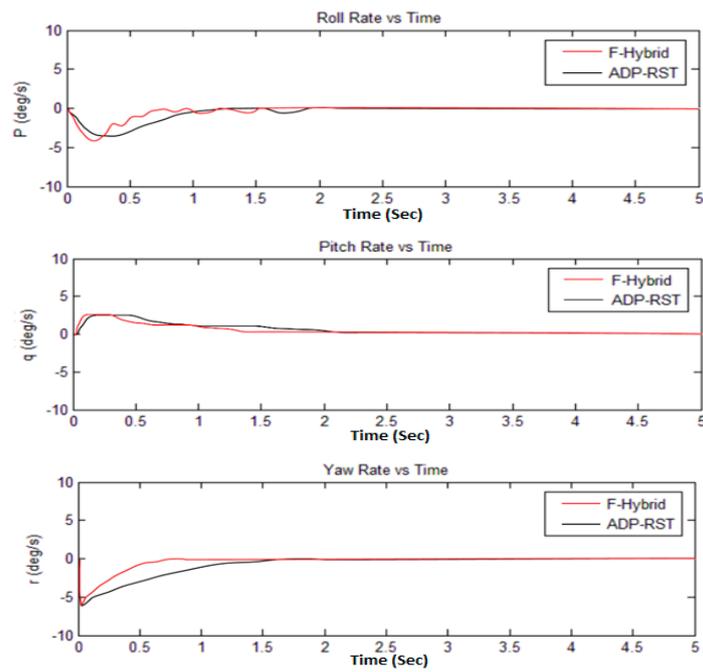


Figure 12. Comparison of F-Hybrid with adaptive RST for rotational velocity responses.

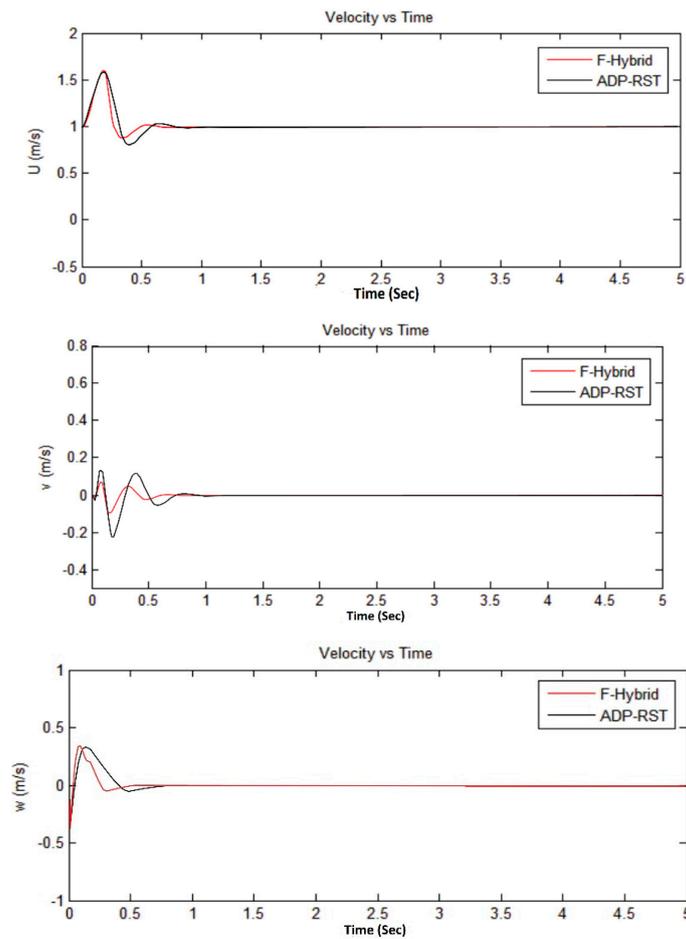


Figure 13. Comparison of F-Hybrid with adaptive RST for translational velocity responses.

The sampling time was set to 0.2 s for all simulations, whereas every simulation was done in five seconds. The simulation results from Figures 9–12 show that our proposed algorithm is quite fast and converges to zero without any settling time error. The Euler angle initial conditions are  $\varphi = 2.5$ ,  $\theta = -2$ , and  $\psi = 4.7$  degrees. In Figure 13, the translational velocity components are  $U = 1$ ,  $V = W = 0$ , meaning it can easily control the attitude of UAV, without any overshoot, undershoot, and transient errors. Furthermore, the settling time response also gets better than the previously-used adaptive RST controller. In a very short time of about 1 to 1.5 s, all of the simulated parameter values yield to the original trimmed conditions. The results show that our proposed controller is able to stabilize the attitude angles and altitude of the tri-rotor UAV.

## 5. Conclusions

In this paper, a new and novel approach of a fuzzy hybrid controller is presented for stabilizing the nonlinear behavior of a tri-rotor UAV and to achieve the desired altitude. The model of the tri-rotor UAV is based on the Newton-Euler method. The method is applied on the translational and rotational velocity subsystems of the aircraft. To observe the stability of the UAV we must stabilize the attitude and altitude responses. The RST with MRAC nonlinear controller algorithm is based on the MIT rule and the sensitivity function of the closed-loop system has shown significant results. Moreover, fuzzy logic controllers have been proposed with linguistic logic, which enhanced the performance. The effectiveness and stability of our algorithm are implemented using nonlinear simulation and it is observed that the proposed method has better transient characteristics and performance with zero steady state error on rigid environments with reasonable rise and settling time. Furthermore, the proposed method shows better robustness and very fast convergence towards stability in the presence of a system disturbance.

**Acknowledgments:** This work is sponsored by the National Natural Science Foundation of China (NSFC) under the grant number is 61503185 and supported by Key Laboratory of College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu 210016, China.

**Author Contributions:** The central arguments and structure of the paper were equally devised by each of the authors through a series of discussions. Zain Anwar Ali conceived the idea and carry out experimental work under supervision of Daobo Wang and Muhammad Amir was involved in the writing of this research article and in the results.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Yoon, S.; Lee, S.J.; Lee, B.; Kim, C.J.; Lee, Y.J.; Sung, S. Design and flight test of a small Tri-rotor unmanned vehicle with a LQR based onboard attitude control system. *Int. J. Innov. Comput. Inf. Control* **2013**, *9*, 2347–2360.
2. Jones, P.; Ludington, B.; Reimann, J.; Vachtsevanos, G. Intelligent Control of Unmanned Aerial Vehicles for Improved Autonomy. *Eur. J. Control* **2007**, *13*, 320–333. [[CrossRef](#)]
3. Metni, N.; Hamel, T. A UAV for bridge inspection: Visual servoing control law with orientation limits. *Autom. Constr.* **2007**, *17*, 3–10. [[CrossRef](#)]
4. Tayebi, A.; McGilvray, S. Attitude stabilization of a four-rotor aerial robot. In Proceedings of the IEEE Conference on Decision and Control, Paradise Island, Bahamas, 14–17 December 2004; Volume 2, pp. 1216–1221.
5. Salazar-cruz, S.; Kendoul, F.; Lozano, R.; Fantoni, I. Real-Time Stabilization of a Small Three-Rotor Aircraft. *IEEE Trans. Aerosp. Electr. Syst.* **2008**, *44*, 783–794. [[CrossRef](#)]
6. Guenard, N.; Hamel, T.; Moreau, V. Dynamic modeling and intuitive control strategy for an “X4-flyer”. In Proceedings of the International Conference on Control and Automation, Hungarian Academy of Science, Budapest, Hungary, 26–29 June 2005; Volume 1, pp. 141–146.
7. Yoo, D.-W.; Oh, H.-D.; Won, D.-Y.; Tahk, M.-J. Dynamic Modeling and Stabilization Techniques for Tri-Rotor Unmanned Aerial Vehicles. *Int. J. Aeronaut. Space Sci.* **2010**, *11*, 167–174. [[CrossRef](#)]

8. Benallegue, A.; Mokhtari, A.; Fridman, L. High-order sliding-mode observer for a quadrotor UAV. *Int. J. Robust Nonlinear Control* **2008**, *18*, 427–440. [[CrossRef](#)]
9. Kim, J. Model-Error Control Synthesis: A New Approach to Robust Control. Ph.D. Thesis, Texas A&M University, USA, 2002.
10. Castillo, P.; Lozano, R.; Dzul, A. *Modelling and Control of Mini-Flying Machines*; Springer-Verlag London: London, UK, 2005.
11. Desa, H.; Ahmed, S.F. Adaptive Hybrid Control Algorithm Design for Attitude Stabilization of Quadrotor (UAV). *Arch. Des. Sci.* **2013**, *66*, 51–64.
12. Mohammadi, M.; Shahri, A.M. Adaptive Nonlinear Stabilization Control for a Quadrotor UAV: Theory, Simulation and Experimentation. *J. Intell. Robot. Syst.* **2013**, *72*, 105–122. [[CrossRef](#)]
13. Dydek, Z.T.; Annaswamy, A.M. Adaptive Control of Quadrotor UAVs in the Presence of Actuator Uncertainties. In Proceedings of the AIAA Guidance, Navigation, and Control Conference 2010, Atlanta, Georgia, 20–22 April 2010.
14. Sadeghzadeh, I.; Mehta, A.; Zhang, Y. Fault/Damage Tolerant Control of a Quadrotor Helicopter UAV Using Model Reference Adaptive Control and Gain-Scheduled PID. In Proceedings of the AIAA Guidance, Navigation, and Control Conference, Portland, Oregon, 8–11 August 2011.
15. Chadlia, M.; Aouaoudab, S.; Karimic, H.R.; Shid, P. Robust fault tolerant tracking controller design for a VTOL aircraft. *J. Frankl. Inst.* **2013**, *350*, 2627–2645. [[CrossRef](#)]
16. Aouaouda, S.; Chadli, M.; Karimi, H.-R. Robust static output-feedback controller design against sensor failure for vehicle dynamics. In Proceedings of the IET Control Theory & Applications, 12 June 2014; Volume 8, pp. 728–737. [[CrossRef](#)]
17. Chiou, J.S.; Tran, H.K.; Peng, S.T. Attitude control of a single tilt tri-rotor UAV SYSTEM: Dynamic modeling and each channel's nonlinear controllers design. *Math. Probl. Eng.* **2013**, *2*. [[CrossRef](#)] [[PubMed](#)]
18. Yeh, F.K.; Huang, C.W.; Huang, J.J. Adaptive fuzzy sliding-mode control for a mini-UAV with propellers. In Proceedings of the InSICE Annual Conference (SICE), Tokyo, Japan, 13–18 September 2011; pp. 645–650.
19. Chiou, J.S.; Tran, H.K.; Peng, S.T. Design Hybrid Evolutionary PSO Aiding Miniature Aerial Vehicle Controllers. *Math. Probl. Eng.* **2015**, *12*. [[CrossRef](#)]
20. Ali, Z.A.; Wang, D.; Javed, R.; Aklbar, A. Modeling & Controlling the Dynamics of Tri-rotor UAV Using Robust RST Controller with MRAC Adaptive Algorithm. *Int. J. Control Autom.* **2016**, *9*, 61–76.
21. Cuenca, A.; Salt, J. RST controller design for a non-uniform multi-rate control system. *J. Process Control* **2012**, *22*, 1865–1877. [[CrossRef](#)]
22. Landau, I.D. The R-S-T digital controller design and applications. *Control Eng. Pract.* **1998**, *6*, 155–165. [[CrossRef](#)]
23. Stevens, B.L.; Lewis, F.L. *Aircraft Control and Simulation*; Wileys: New York, NY, USA, 1992.
24. Padfield, G.D. The Theory and Application of Flying Qualities and Simulation Modeling. In *Helicopter Flight Dynamics*; AIAA: Education Series, Washington, DC, 1996.
25. Deuflhard, P. *Newton Methods for Nonlinear Problems*, 1st ed.; Springer: Berlin, Germany, 2005.
26. Weihua, Z.; Go, T.H. Robust decentralized formation flight control. *Int. J. Aerosp. Eng.* **2011**. [[CrossRef](#)]
27. Mohamed, K.M. Design and Control of UAV Systems: A Tri-Rotor UAV Case Study. Ph.D. Thesis, The University of Manchester, Manchester, UK, 31 October 2012.
28. Shanmugasundram, R.; Zakaraiah, K.M.; Yadaiah, N. Modeling, simulation and analysis of controllers for brushless direct current motor drives. *J. Vib. Control* **2012**. [[CrossRef](#)]
29. Sincero, G.; Cros, J.; Viarouge, P. Efficient simulation method for comparison of brush and brushless DC motors for light traction application. In Proceedings of the 13th European Conference on Power Electronics and Applications, Barcelona, Spain, 8–10 September 2009; pp. 1–10.
30. Mamdani, E.H. Application of fuzzy logic to approximate reasoning using linguistic synthesis. *IEEE Trans. Comput.* **1977**, *100*, 1182–1191. [[CrossRef](#)]
31. Passino, K.M.; Yurkovich, S. *Fuzzy Control* by Addison-Wesley: Reading, PA, USA, 1998.
32. Mamdani, E.H. Application of fuzzy algorithms for control of simple dynamic plant. *Proc. Inst. Electr. Eng.* **1974**, *121*, 1585–1588. [[CrossRef](#)]
33. Salt, J.; Sala, A.; Albertos, P. A transfer-function approach to dual-rate controller design for unstable and non-minimum-phase plants. *IEEE Trans. Control Syst. Technol.* **2011**, *19*, 1186–1194. [[CrossRef](#)]

34. Astrom, K.J.; Wittenmark, B. *Computer-Controlled Systems: Theory and Design*; Prentice Hall: Englewood Cliffs, NJ, USA, 1984.
35. Salt, J.; Albertos, P. Model-based multirate controllers design. *IEEE Trans. Control Syst. Technol.* **2005**, *13*, 988–997. [[CrossRef](#)]
36. Schmidt, P. Intelligent model reference adaptive control applied to motion control. In Proceedings of the Record of the 1995 IEEE Industry Applications Conference Thirtieth IAS Annual Meeting IAS-95, Orlando, FL, USA, 8–12 October 1995.



© 2016 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).