Adaptive Neural Network for a Quadrotor
Unmanned Aerial Vehicle

Hana Boudjedir\textsuperscript{1}, Fouad Yacef\textsuperscript{1}, Omar Bouhali\textsuperscript{1} and Nassim Rizoug\textsuperscript{2}

\textsuperscript{1}LAJ Lab, Automatic department, Jijel University, Algeria
hana_boudjedir@yahoo.fr, Yaceffouad@yahoo.fr, bouhali_omar@yahoo.fr

\textsuperscript{2}Mecatronic Lab, ESTACA School, Laval, France
nrizoug@yahoo.fr

\textbf{ABSTRACT}

A new adaptive neural control scheme for quadrotor helicopter stabilization at the presence of sinusoidal disturbance is proposed in this paper. The adaptive control classical laws such $e$-modification presents some limitations in particular when persistent oscillations are presenting in the input. These techniques can create a dilemma between weights drifting and tracking errors. To avoid this problem in adaptive Single Hidden Layer neural network scheme, a new solution is proposed in this work. The main idea is based on the use of two SHL in parallel instead of one in the closed loop in order to estimate the unknown nonlinear function in Quadrotor dynamical model. The learning algorithms of the two SHL Networks are obtained using the Lyapunov stability method. The simulation results are given to highlight the performances of the proposed control scheme.

\textbf{KEYWORDS}

Adaptive control, $e$-modification, Neural Network control, Quadrotor, Single Hidden Layer, Lyapunov stability

\textbf{1. INTRODUCTION}

In the last years, unmanned aerial vehicles (UAV) have gained a strong interest. The recent advances in low-power embedded processors, miniature sensors and control theory are opening new horizons in terms of miniaturization and field of application [1]. The Quadrotor Helicopter is considered as one of the most popular UAV platform. The main reasons for all this attention have stemmed from its simple construction and its large payload as compared with the conventional helicopter.

The Quadrotor is an under actuated system with six degrees of freedom and only four control inputs. To solve the Quadrotor UAV tracking control problem many techniques have been proposed [1-6] where the control objective is to control four outputs (three desired cartesian positions and a desired yaw angle). Nonlinear Backstepping and sliding mode controls have been proposed in [1, 2]. In [3] the $H\infty$ type robust control law was used and in [4] a PID and LQR controls have been applied. However, all these techniques use the dynamic model in the controller which requires the accuracy of the dynamic model.

DOI:10.5121/ijfcst.2012.2401
A popular method for handling unknown nonlinearities is based on the introduction of neural networks, tuned online using adaptive control techniques [5]. Several Neural Networks adaptive schemes have been proposed for quadrotor’s control [6-9]. The classical adaptive technique may cause weight drifting which can cause control signal chatter. To overcome the problem of drifting weights, several methods have been proposed as projection algorithms [10], $\sigma$-switching [11] and dead zone [12]. However, these solutions require a priori knowledge about model for the first and the second techniques and about modeling errors for the last one, which is not easy to determine in practice. Other solutions, they can be found in the literature where no information about the system is required as $\sigma$-modification method [13]; however this technique never leads to tracking error equal to zero. For that reason other version of this technique was proposed, named $e$-modification [14]. Nevertheless this method is characterized by its dilemma between weight drifting and tracking errors especially when sinusoidal oscillations are presenting. In fact, using small values for $e$-modification term achieves a high performance in the short term but may cause the divergence of Neural Network’s weight and therefore the instability of the closed loop system, on the other hand the use of large values for $e$-modification limits the weight drift sufficiently, however, such values can cause an important increase in tracking errors with time or even instability of all system.

In this present paper we propose a new robust adaptive scheme that does not need any prior information on model dynamics and does not require a sacrifice of performance to stop weight drifting. The main idea is based on the property that many different sets of weights can uniformly approximate the same nonlinear function. It follows that there will be an alternate NN that can uniformly approximate the same nonlinear function as the control NN. This kind of solutions was first proposed in [15] and [16] where two RBF (radial basis function) network were used in the control loop. In our work we will use two SHL Neural Networks to halt the weight drift while maintaining high level of performance. The first difference between our scheme and the one proposed in [15], [16] is in the structure of the NN implemented where by opposition to the RBF NN the use of a SHL introduces complications in the proof. The second difference is in the adaptations laws; in this paper a new algorithm was proposed in order to get improved results in tracking errors compared to those obtained in [16] without having any degradation in weight’s term.

This paper is organized as follows. In Section 2, the dynamical model of the Quadrotor will be presented. In section 3, Problem formulation is described. In Section 4, neural network adaptive controller based on $e$-modification technique will be developed. In section 5 we propose a new adaptive control based on the use of two SHL neural networks at the same control loop. Section 6, simulation results are given to show the effectiveness and feasibility of the proposed control and the improvements given by the use of the proposed adaptations laws. The conclusion is given in section 7.

2. QUADROTOR DYNAMICS

The quadrotor helicopter conventional structure is composed of four propellers in cross configuration where the pairs of rotors (1,3) and (2,4), turn in opposite directions in order to prevent the device from turning on her (Figure. 1).
Figure 1. Model of conventional Quadrotor.

Flights mode in quadrotor are defined according to the direction and the velocity of each rotor. Vertical ascending (descending) flight is created by increasing (decreasing) thrust forces. Applying speed difference between front and rear rotors we will have pitching motion which is defined as a rotation motion around Y axis coupled with a translation motion along X axis. The same analogy is applied to obtain rolling motion, but by changing the side motors speed this time and as result we will have a rotation motion around X axis coupled with a translation motion along Y axis. The last flight mode is yaw motion; this one is obtained while increasing (decreasing) speed of motors (1.3) compared to (2.4) motors speed. Unlike pitch and roll motions, yaw rotation is the result of reactive torques produced by rotors rotation.

By using the formalism of Newton-Euler [2, 3, 6, 9] the dynamic model of a Quadrotor can be expressed as:

\[
\begin{align*}
\dot{\phi} &= \frac{U_\phi(t)}{I_x} + \left( \frac{I_y - I_z}{I_x} \right) \dot{\psi} \dot{\theta} - \frac{J \Omega_x(t)}{I_x} \dot{\phi} - \frac{k_{frx}}{I_x} \dot{\psi}^2 + d_\phi \\
\dot{\theta} &= \frac{U_\theta(t)}{I_y} + \left( \frac{I_z - I_x}{I_y} \right) \dot{\phi} \dot{\psi} + \frac{J \Omega_y(t)}{I_y} \dot{\psi} - \frac{k_{frz}}{I_y} \dot{\theta}^2 + d_\theta \\
\dot{\psi} &= \frac{U_\psi(t)}{I_z} + \left( \frac{I_x - I_y}{I_z} \right) \dot{\phi} \dot{\theta} - \frac{k_{frz}}{I_z} \dot{\psi}^2 + d_\psi \\
\dot{x} &= \frac{1}{m} \left( \left( C_{x\phi} S_\phi C_\psi + S_\phi S_{x\psi} \right) U_1(t) \right) - k_{fx} \dot{x} + d_x \\
\dot{y} &= \frac{1}{m} \left( \left( S_\psi S_\phi C_\phi - S_\phi S_{x\psi} \right) U_1(t) \right) - k_{fy} \dot{y} + d_y \\
\dot{z} &= \frac{1}{m} \left( C_\phi C_\psi U_1(t) - k_{fz} \dot{z} \right) - g + d_z 
\end{align*}
\]

Where: \( m \): Quadrotor mass, \( k_\phi \): Thrust factor, \( k_d \): drag factor, \( \omega_1 \): angular rotor speed, \( J=\text{diag}(I_x, I_y, I_z) \): inertia matrix, \( K_d=\text{diag}(k_{fx}, k_{fy}, k_{fz}) \): drag translation matrix, \( K_\phi=\text{diag}(k_{frx}, k_{frz}, k_{frw}) \): friction
aerodynamic coefficients, $\xi = [x \ y \ z]$: position vector, $\eta = [\phi \ \theta \ \psi]$ represents the angles of roll, pitch and yaw, $d_{x,y,z,\phi,\theta,\psi}$ disturbance and $\Omega = \sum_{i=1}^{4} (-1)^{i+1} \omega_i$.

The control inputs according to the angular velocities of the four rotors are given by:

$$
\begin{bmatrix}
U_1(t) \\
U_x(t) \\
U_y(t) \\
U_z(t)
\end{bmatrix} =
\begin{bmatrix}
k_p & k_p & k_p & k_p \\
0 & -lk_p & 0 & lk_p \\
-lk_p & 0 & lk_p & 0 \\
k_d & -k_d & k_d & -k_d
\end{bmatrix}
\begin{bmatrix}
\omega_x^2 \\
\omega_y^2 \\
\omega_z^2 \\
\omega_r^2
\end{bmatrix}
$$

(2)

2.2. Virtual control

In this section, three virtual control inputs will be defined to ensure that the Quadrotor follows a specified trajectory. Those virtual controls are [9]:

$$
\begin{align*}
U_1(t) &= (C_x \ S_y \ C_{\phi} + S_x \ S_y \ S_{\phi}) U_{i1}(t) \\
U_x(t) &= (S_x \ S_{\phi} \ C_{\phi} - S_x \ S_{\phi} \ S_{\phi}) U_{i1}(t) \\
U_y(t) &= (C_x \ S_{\phi}) U_{i1}(t)
\end{align*}
$$

(3)

The physical interpretation of these virtual control, means that the control of translation motion depends on three common inputs are: $\theta$, $\phi$ and $U_{i1}(t)$. This requires that the rolling and pitching motions must take a desired trajectory to guarantee the control task of translation motion. Using equation (3) the desired trajectories in rolling and pitching are defined as follow [9]:

$$
\begin{align*}
\varphi_d &= \arcsin\left(\frac{U_x(t) S_{\phi_d} - U_y(t) C_{\phi_d}}{\sqrt{U_x(t)^2 + U_y(t)^2 + U_z(t)^2}}\right) \\
\theta_d &= \arctan\left(\frac{C_{\phi_d} U_x(t) + S_{\phi_d} U_y(t)}{U_z(t)}\right)
\end{align*}
$$

(4)

with $\varphi_d$, $\theta_d$ and $\psi_d$ are the desired trajectories in roll, pitch and yaw respectively.

3. Problem Formulation

From equation (1) and after using virtual control input giving by (3), we can consider that the dynamic model of the quadrotor is the nonlinear time-varying system described by two differentials equations with the following structure for each one:

$$
\ddot{y} = F(x) + G(x) U
$$

(5)

where: $x = [y_1 \ y_2 \ y_3 \ y_4 \ y_5 \ y_6 \ y_7 \ y_8 \ y_9 \ y_{10} \ y_{11} \ y_{12}]$ is the output which is assumed available for measurement, $U = [u_1 \ \cdots \ u_p]^T$ is the input vector, $Y = [y_1 \ \cdots \ y_p]^T$ is the output vector and $F(X), G(X)$ are smooth nonlinear functions.
Assumption.1: The desired output trajectory $Y_d(t)$ and its first derivative are smooth and bounded.

Assumption.2: The gain $G(X)$ is bounded, positive definite and slowly time varying.

Let’s define the tracking error as:

$$e(t) = Y_d(t) - Y(t)$$

(6)

and the filtered tracking errors by [17]:

$$S = \dot{e} + \lambda e$$

(7)

Where: $S = \begin{bmatrix} s_1 & \cdots & s_p \end{bmatrix}^T$ and $\lambda \in \mathbb{R}^{p \times p} > 0$ is a diagonal matrix.

From (7), the convergence of $S$ to zero implies convergence of the tracking error $e(t)$ and its derivative $\dot{e}(t)$ to zero [17]. So, the objective of control is to synthesize a control law that allows the convergence to zero of the filtered error.

The time derivative of the (7) can be written as:

$$\dot{S} = v - F(X) - G(X)U$$

(8)

where:

$$v = \ddot{Y_d} + \lambda \dot{e}$$

(9)

The ideal control law that guarantees the closed-loop performance is defined by the following expression:

$$U(t) = U_q^*(t) + U_{pd}^*(t), i = 1:p$$

(10)

With:

$$U_q^*(t) = G^{-1}(v - F(X))$$

$$U_{pd}^*(t) = KS, K > 0$$

(11)

Substituting (10) into (8), we obtain:

$$\dot{S} = -G(X)K S \Rightarrow S^T \dot{S} \leq 0 \quad i = 1:p$$

(12)

Which implying that $S \to 0$ when $t \to \infty$, and therefore $e, \dot{e} \to 0$ when $t \to \infty$ [17].

According to the above analysis, the control law (10) is easily obtained if the nonlinear functions $F(X)$ and $G(X)$ are known. However, in this paper, these functions are assumed to be unknown, so the above design method cannot be applied directly. To overcome this constraint, two SHL network is used in the control scheme to estimate online the unknown equivalent controller defined in (11).
4. SYNTHESIS OF CONVENTIONAL DIRECT NEURAL ADAPTIVE CONTROL

The adaptive control that will be used in this paper

\[ U(t) = U_{eq}(t) + U_{pd}(t) + U_r(t) \]  \hspace{1cm} (13)

- \( U_{eq}(t) \) : is the neural adaptive control term used to approximate the ideal equivalent control defined in eq. (11),
- \( U_{pd}(t) \) : PD control is given by.

\[ U_{pd}(t) = K S \]  \hspace{1cm} (14)

- \( U_r(t) \) : is used to compensate estimation’s errors its expression is defined by

\[ U_r(t) = S \frac{\dot{\bar{w}}}{\|S\|^2 + \beta} \]  \hspace{1cm} (15)

Where:

\[
\begin{bmatrix}
\dot{\beta} = -\mu \dot{\beta} \\
\dot{\bar{w}} = \gamma \|S\|
\end{bmatrix}
\]  \hspace{1cm} (16)

With : \( \mu > 0, \ \beta(0) > 0, \ \dot{\bar{w}}(0) \geq 0 \) and, And \( \bar{w} \in \mathcal{R} \) is the estimated of \( w \), which will be defined later.

4.1. Classical adaptive control

It has been proved that the SHL neural network is a universal approximator [13]. So we assume that the ANN presented in Figure 2 can approximate the unknown implicit ideal controller of the eq. (11) as follows:

\[ U_{eq}^* = G^{-1}(\nu - F) = W^T \sigma(y^T X) + \varepsilon \]  \hspace{1cm} (17)
Figure 2: SHL NN used in the control schemes.

where: \( ne \) input neurons number, \( nc \) hidden neurons number, \( ns=3 \) output neurons number, \( \chi \in \mathbb{R}^{ne} \) input vector and \( \sigma(.) \) is sigmoid activation function, \( W \in \mathbb{R}^{nc \times ne} \) and \( V \in \mathbb{R}^{nc \times ne} \) are the ideal weight and \( \varepsilon \) is the reconstruction error.

Since the optimal weights, it is necessary to estimate them by an adaptation mechanism so that the output feedback control law can be realized.

\( \hat{W} \) and \( \hat{V} \) are the estimate of \( W \) and \( V \). Thus, the adaptive control approximating the ideal SHLNN output defined in (17) is given by:

\[
U_{eq} = \hat{W}^{T} \sigma(\hat{V}^{T} \chi)
\]  

(18)

**Assumption 3:** \( \|W\|_{F} \leq W_{m}, \|V\|_{F} \leq V_{m} \) with: \( W_{m} \) et \( V_{m} \) are unknown positive constants.

The equivalent control identification error is giving by:

\[
\tilde{U}_{eq} = U_{eq}^{*} - U_{eq} = \tilde{W}^{T} \hat{\sigma} + \hat{W}^{T} \hat{\sigma}^{'} \hat{V}^{T} \chi + w
\]

(19)

Where, \( w \) presents the estimation errors:

\[
w = \varepsilon + \tilde{W}^{T} \hat{\sigma}^{'} \hat{V}^{T} \chi + W^{T} O(\hat{V}^{T} \chi)^{2}
\]

(20)

Where : \( \tilde{W} = W - \hat{W} \) and \( \tilde{V} = V - \hat{V} \) are parameter estimation errors.

**Assumption 4:** \( \|w\| \leq \overline{w} \). with: \( \overline{w} \) is an unknown positive constant its estimation is given by eq. (16).

To achieve the goal of controlling the weights adaptation laws are defined by:
\[
\begin{align*}
\dot{W} &= F_W \left( \hat{\sigma} \hat{S}^T - \kappa \|S\| \hat{W} \right) \\
\dot{V} &= F_V \left( \hat{X} S^T \hat{W}^T \hat{\sigma}' - \kappa \|S\| \hat{V} \right)
\end{align*}
\]

(21)

with : \(F_W > 0, F_V > 0\) are the adaptive gains and \(\kappa\) is a positive constant.

The additional term \(e\)-modification is used to prevent the explosion of the weights [14].

5. PROPOSED ADAPTIVE CONTROL

The traditional adaptations laws given by eq. (21) may not perform satisfactorily especially in the presence of oscillating wind disturbance as will be demonstrated in the result section. For this reason, a second SHL NN (alternate network) has the same structure as the first one (controller network) will be added to the adaptive scheme in order to prevent the weight drift and having a high level of performance at the time in other word, solving the dilemma between weight drifting and the tracking error.

The expression of alternate SHL NN is giving by:

\[\hat{U}_{af} = \hat{P}_W \sigma(\hat{P}_V X)\]

(22)

Where: \(\hat{P}_W\) and \(\hat{P}_V\) are the alternate weights.

The main idea is that an alternate SHL NN can uniformly approximate the same nonlinear unknown function as the control SHL NN. The control weights, tend to drift since they are updated using the state error. In contrast, the alternate weights are undergoing supervised learning as is shown in Figure.3 and so the weight drift can easily be stopped. So that the idea is to keep the values of control weight close to the alternate weight values in order to get :

\[\hat{W}^T \sigma(\hat{W}^T X) = \hat{P}_W^T \sigma(\hat{P}_V^T X)\]

Let define the error between the two network’s outputs by:

\[\delta_R = \hat{W}^T \sigma(\hat{W}^T X) - \hat{P}_W^T \sigma(\hat{P}_V^T X)\]

(23)

And the error \(\delta_c\) by:

\[\delta_c = (\hat{V}^T - \hat{P}_V^T) X\]

(24)
In order to prevent weight drift and having high level of performance, the adaptive parameters \( \hat{W}, \hat{V}, \hat{P}_w \) and \( \hat{P}_v \) are updated by the proposed adaptive laws:

### 5.1. Controller weights

\[
\begin{align*}
\dot{V} &= F_v \left( \chi S^T \Sigma \hat{W}^T \dot{\sigma} - \alpha_v \chi \delta_c^T + \kappa_1 \left( \hat{P}_v - \hat{V} \right) \right) \\
\dot{W} &= F_w \left( \dot{\sigma} S^T \Sigma - \alpha_w \dot{\sigma} \delta_c^T - \nu_w \dot{\sigma} \delta_c^T \hat{W} + \kappa_w \left( \hat{P}_w - \hat{W} \right) \right)
\end{align*}
\]

(25)

### 5.2. Alternate weights:

\[
\begin{align*}
\dot{\hat{P}}_v &= F_v \left( \alpha_v \chi \delta_c^T - \mu_v \hat{P}_v \right) \\
\dot{\hat{P}}_w &= F_w \left( \alpha_w \dot{\sigma} \delta_c^T - \mu_w \hat{P}_w \right)
\end{align*}
\]

(26)

with: \( F_v > 0, \ F_w > 0, \ F_{P_v} > 0, \ F_{P_w} > 0 \) and \( \alpha_v, \alpha_w, \kappa_v, \kappa_w, \mu_v, \mu_w, \zeta_R, \zeta_C \) are positive constants and \( \Sigma \) is a diagonal, positive definite matrix.

### 5.3. Proof

Consider the following Lyapunov function candidate:

\[
L = \frac{1}{2} S^T \Sigma^{-1} S + \frac{1}{2} \text{tr} \left( \hat{W}^T F_w^{-1} \hat{W} \right) + \frac{1}{2} \text{tr} \left( \hat{V}^T F_v^{-1} \hat{V} \right) + \frac{1}{2} \text{tr} \left( \hat{P}_w F_{P_w}^{-1} \hat{P}_w \right) + \frac{1}{2} \text{tr} \left( \hat{P}_v F_{P_v}^{-1} \hat{P}_v \right) + \frac{1}{2} \text{tr} \left( \hat{P}_w F_{P_w}^{-1} \hat{P}_w \right)
\]

(27)

The first derivative of (27) with respect to time is given by:

\[
\dot{L} = S^T \Sigma^{-1} \dot{S} + \text{tr} \left( \hat{W}^T F_w^{-1} \dot{\hat{W}} \right) + \text{tr} \left( \hat{V}^T F_v^{-1} \dot{\hat{V}} \right) + \text{tr} \left( \hat{P}_w F_{P_w}^{-1} \dot{\hat{P}_w} \right) + \text{tr} \left( \hat{P}_v F_{P_v}^{-1} \dot{\hat{P}_v} \right) + \frac{1}{2} \frac{\mu}{\gamma} \hat{\Sigma}^T \hat{\Sigma} \hat{\Sigma} + \frac{1}{\mu} \beta \]

(28)

Substituting (13), (16), (25) and (26) in (28) Provide:

\[
\dot{L} \leq -S^T \Sigma \kappa S + \mu_w \left\| \hat{W} \right\|_F \left\| \hat{P}_w \right\|_F + \mu_v \left\| \hat{V} \right\|_F \left\| \hat{P}_v \right\|_F
\]

(29)

By choosing \( \mu_w > \frac{\kappa_w}{4} \) and \( \mu_v > \frac{\kappa_v}{4} \) we can guarantee that \( \dot{L} \leq 0 \) if:

\[
S^T \Sigma \kappa S > \mu_w \left\| \hat{V} \right\|_F \left\| \hat{P}_v \right\|_F + \mu_v \left\| \hat{V} \right\|_F \left\| \hat{P}_v \right\|_F .
\]

So the filtered error is bounded and therefore the loop system signals are bounded.
6. SIMULATION RESULTS

The simulation experiments are performed to compare the conventional adaptive control and the proposed technique. Figures 4, 5, 6, 7 and 8 show the simulation results after application to quadrotor actuated part with wind disturbance of form (30) and 50% parametric variation from time \( t = 10 \text{sec} \).

\[
d_{\varphi} = d_{\theta} = d_{\psi} = 2 \sin(2\pi t) + 2
\]  

(30)

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Value</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( m )</td>
<td>0.486 (Kg)</td>
<td>( I_z )</td>
<td>7.65e-3 (KgM²)</td>
</tr>
<tr>
<td>( l )</td>
<td>0.25 (m)</td>
<td>( g )</td>
<td>9.81 (M/s²)</td>
</tr>
<tr>
<td>( k_d )</td>
<td>3.23e-7 (NM/s²)</td>
<td>( k_{p_{\theta}}, k_{p_{\psi}} )</td>
<td>5.56e-4 (Ns/M)</td>
</tr>
<tr>
<td>( k_p )</td>
<td>2.98e-5 (N/s²)</td>
<td>( k_{p_{\varphi}}, k_{p_{\theta}} )</td>
<td>6.35e-4 (Ns/M)</td>
</tr>
<tr>
<td>( J_r )</td>
<td>2.83e-5 (KgM²)</td>
<td>( k_{p_{\phi}}, k_{p_{\theta}} )</td>
<td>5.56e-4 (Ns/r)</td>
</tr>
<tr>
<td>( I_\phi )</td>
<td>3.82e-3 (KgM²)</td>
<td>( k_{p_{\phi}} )</td>
<td>6.35e-4 (Ns/r)</td>
</tr>
<tr>
<td>( I_\theta )</td>
<td>3.82e-3 (KgM²)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Quadrotor Parameters.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Value</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \lambda_{\eta} )</td>
<td>diag(1, 1, 1)</td>
<td>( \lambda_{\eta} )</td>
<td>diag(1, 1, 1)</td>
</tr>
<tr>
<td>( K_{\eta} )</td>
<td>diag(0.1, 0.1, 0.2)</td>
<td>( K_{\eta} )</td>
<td>diag(0.1, 0.1, 0.2)</td>
</tr>
<tr>
<td>( F_{W_{\eta}} )</td>
<td>( I_{3,3} )</td>
<td>( F_{W_{\eta}}, F_{W_{\theta}} )</td>
<td>( I_{3,3} )</td>
</tr>
<tr>
<td>( F_{V_{\eta}} )</td>
<td>( I_{15,15} )</td>
<td>( F_{V_{\eta}}, F_{V_{\theta}} )</td>
<td>( I_{15,15} )</td>
</tr>
<tr>
<td>( \epsilon )</td>
<td>0.1</td>
<td>( \epsilon )</td>
<td>0.1</td>
</tr>
<tr>
<td>( \eta )</td>
<td>0.01</td>
<td>( \eta )</td>
<td>0.01</td>
</tr>
<tr>
<td>( \beta_{\varphi}(0) )</td>
<td>1</td>
<td>( \beta_{\varphi}(0) )</td>
<td>1</td>
</tr>
<tr>
<td>( \gamma_{\eta} )</td>
<td>0.1</td>
<td>( \gamma_{\eta} )</td>
<td>0.1</td>
</tr>
<tr>
<td>( \Sigma )</td>
<td>3</td>
<td>( I_{3,3} )</td>
<td></td>
</tr>
<tr>
<td>( \nu )</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Controller Parameters.

The simulation results present a comparative study between \textit{e-modification}, the presented method in [16] and the proposed technique.

For \textit{e-modification} technique, the value of \( \kappa \) must be very large in order to halt weight drift (Figure 4.b) but we had degradations in performances as results as with \( \kappa=100 \) (Figure 4.a). Otherwise using small value for the parameter \( \kappa \), the system achieves an optimum level performance in the short term, but the weights eventually drift to large magnitudes which could create instability of the system over a long period.

The proposed method achieves high performance while removing weight drift. Where we can see that the same result gotten by \textit{e-modification} technique for weight by using large values is obtained with proposed method by using small values (\( \kappa_{w}<5 \)) and as result we had well performances in terms of pursuing (Figure 6).

- For \( \kappa=100 \) in \textit{e-modification} technique we had \( \int W_{\eta} = 1.72 \) and \( \int S_{\eta} = 8.29 \).
- For \( \kappa_{w}=5 \) in the developed method in [16] we got \( \int W_{\eta} = 1.69 \) and \( \int S_{\eta} = 6.93 \).
- and with \( \kappa_{w}=5 \) in the proposed technique we obtained : \( \int W_{\eta} = 1.67 \) and \( \int S_{\eta} = 3.57 \).
So for the same parameter value, the proposed method shows the improvements provided on performance compared to the developed method in [16] without any weight drift as is shown in Figure 6.

Figure 4: $e$-modification technique with sinusoidal disturbance

Figure 5: Comparing performance and weight drift. (Solid line: proposed technique, dashed line: proposed technique in [16]).

Figure 6: Filtered error (Solid line: proposed technique $k_w=5$, dashed line: proposed technique in [16] $k_w=5$, dotted line: $e$-modification $k=100$).
CONCLUSION

A new adaptive neural network was proposed for an under actuated Quadrotor UAV. The problem treated in this work is the dilemma between weight drift and tracking error in conventional adaptive control as \( e \)-modification technique. As solution we proposed to implement two parallel SHL in the adaptive control scheme for estimating the same unknown part in the dynamic model. The uniformly ultimately boundedness of the tracking error and all signals in the overall closed-loop system is proved using Lyapunov's direct method. The performances of the control law are validated by simulation of a Quadrotor control, a high level of performance was obtained without having weights drift.
REFERENCES