Adaptive Flight Control Law Based on Neural Networks and Dynamic Inversion for Micro Aerial Vehicles

Romulus Lungu[1], Mihai Lungu[2]

[1]University of Craiova, Faculty of Electrical Engineering, 107 Decebal Blvd., Craiova, Romania, romulus_lungu@yahoo.com
[2]University of Craiova, Faculty of Electrical Engineering, 107 Decebal Blvd., Craiova, Romania, Lma1312@yahoo.com

The paper presents two new adaptive systems, for the attitude's control of the micro aerial vehicles (MAVs) – insect type. The dynamic model describing the motion of MAVs with respect to the Earth tied frame is nonlinear and the design of the new adaptive control system is based on the dynamic inversion technique. The inversion error is calculated with respect to the control law and two matrices (inertia and dynamic damping matrices) which express the deviation of the estimated matrices relative to the calculated ones (the matrices from the nonlinear dynamics of MAVs) in conditions of absolute stability in closed loop system by using the Lyapunov theory. To completely compensate this error, an adaptive component (output of a neural network) is added in the control law. The system also includes a second order reference model which provides the desired attitude vector and its derivative. The two variants of the new adaptive control system are validated by complex numerical simulations.

Key words: MAV, Dynamic inversion, Neural network

1. Introduction

The results of the studies and research regarding the aerodynamics, the flight dynamics and the automatic control of the MAVs (Micro Aerial Vehicles) are numerous. Very important works for this research area have been obtained in [1] and [2]. Important results have been reported in the aerospace domain or in other related areas: mechatronics, automation, or electronics relative to the MAVs' modeling, stabilization of the attitude, and the flight control [2-12]. The micro aerial vehicles can be regarded sometimes as the physical models of the insects. Such mechanisms generally consist of three subsystems: a command subsystem (electrical engine or piezoelectric actuator), a wing actuation equipment (the cinematic mechanism), and a controller. Consisting of such advantages as limited volume, low mass, reasonable cost and rapid transportation, MAVs have demonstrated successful application in the military and civilian fields [13].

Progress has been also made towards: 1) the modeling and the manufacturing of the miniaturized highly performing servo-actuators (thorax of the MAVs - insect type) to achieve the wing's beat motion in the case of flying mini-robots and to command the MAVs' motion through the modification of the wing's beat and attack angles; 2) the manufacturing of miniaturized sensors and transducers which merge the signals provided by the accelerometers, gyros, optical or magnetic sensors; 3) the design and software implementation of the linear or nonlinear observers which serve the navigation and flight control systems [3, 6, 14-16].

Because the atmospheric conditions and the dynamics of MAVs are changing during the flight, it is difficult to use conventional controllers. To design perfect conventional controllers, one has to know the precise mathematical model of the system to be controlled. Furthermore, the MAV's dynamics may vary with respect to the altitude and the flight conditions; therefore, the adaptive controllers are better choices. The adaptive control algorithms for a large class of aerospace vehicles can be modified and adapted to the stabilization and the flight control of the micro aerial vehicles; some of them are based on the usage of the dynamic inversion and neural networks techniques [12, 17-19]. Generally, the dynamic inversion relies on the philosophy of feedback linearization; the plant nonlinearities are canceled and the closed loop plant behaves like a stable linear system. The method is characterized by simplicity in the control structure, ease of implementation, global exponential stability of the tracking error and so on. On the other hand, the strong point of the neural networks (NNs) is their approximation ability, these being capable of approximating the dynamics of unknown systems through learning.

The aim of the flight control system is to attain the commands generated by the guidance system and to maintain steady conditions during flight [20]. A MAV– insect type is highly susceptible to atmospherics disturbances; in addition, the dynamics is highly nonlinear and time varying. Hence, neural networks based adaptive flight controllers with the ability to achieve desired performance are essential for automating the MAV – insect type. In order to achieve this autonomous flight, adaptive flight controllers with the ability to adapt to nonlinear dynamics of the MAV – insect type are necessary. Thus the main objective of the paper is the design and software implementation of such adaptive controller using either the quaternion vector or the vector of MAV's attitude; the design of the control law is based on the dynamic inversion technique and Lyapunov theory, this process involving the calculation of the controllers' coefficients in conditions of stability

Feedback linearization, in its various forms, is perhaps the most commonly employed nonlinear control method; to use it, all parametric plant uncertainties must appear in the same equation of the state-space representation as the control (the main disadvantage of the feedback linearization method). Feed-forward neural networks based on the back propagation learning

algorithm have been used in [21]; the main disadvantage is that the neural networks require a priori training on normal and faulty operating data. Other approaches involves the usage of the time delay neural networks; a controller based on this type of neural networks has been designed in [22], but its main drawback is related to the flight path track accuracy and to the fact that it is enable only under limited conditions. Several neural network control approaches have been proposed based on Lyapunov stability theory [22, 23]. The main advantage of these control schemes is that the adaptive laws were obtained from the Lyapunov synthesis and, therefore, guarantees the system's stability; the disadvantage is that some conditions should be assumed; these requirements are not easy to satisfy in practical control application [24]. Juang designed a new learning technique using a time delay network or networks with back-propagation through time algorithms to control the landing [25]; the main drawbacks are: 1) the number of hidden units was determined by trial; 2) the convergence time is high. Gain scheduling for PID controllers is the common adaptive control technique used in [26]. An advantage of the gain scheduling technique is that it allows the parameters to be changed quickly with change in the plant dynamics. However, the application of the technique is limited due to the absence of a learning process.

The number of neural network based adaptive controllers designed for MAVs' flight control is extremely small; one of the few such neural network based controller is presented in [20], but the real-time implementation of this controller architecture is not practical due to the following reasons: 1) the feedback from the identifier model for the iterative change in the controller outputs requires at least one sample time and 2) the computation time to obtain the new set of plant inputs is considerably high for multiple iterations of training. In the time being, none of the neural network based adaptive controllers for MAVs' flight control was designed for the attitude's control of the MAVs – insect type; this motivates our work. The recent studies and research have culminated in the development of efficient flying robots, but, from our information, none of them has adaptive controllers based on the attitude or the quaternion vectors, Lyapunov theory, dynamic compensators, reference models, and neural networks; this is achieved in this paper, being interesting to see if such adaptive controllers can guarantee the control of MAVs' attitude. Thus, our goal is to design new adaptive systems for the control of the micro aerial vehicles – insect type by using the attitude vector or the quaternion one. Taking into account the advantages of these elements (NNs, dynamic inversion, reference models etc.) and the fact that, till now, no paper deals with the control of MAVs by means of neural networks, dynamic inversion concept, linear dynamic compensators, reference models, and Lyapunov theory, the present paper represents an absolute novelty in the search area of controllers' design for the MAVs' flight.

The structure of the paper is the following one: the MAV's nonlinear dynamics is presented in the paper's second section; the design of the new adaptive control system is achieved in section 3; in the next section, complex simulations to validate the new designed adaptive control system have been performed and the obtained results are analyzed; finally, some conclusions are shared in section 5 of the paper.

2. The Dynamics of the Micro Aerial Vehicles

2.1. The general dynamics of the Micro Aerial Vehicles

The dynamics of MAVs is generally nonlinear; in the case of the gliding flight, the dynamics is linear. The uncertainties cased by the MAV dynamics' incomplete knowing lead to the necessity of a robust controller. In these circumstances, the Lyapunov direct method is used.

Let us denote with $\boldsymbol{\Theta}=\left[\varphi\ \theta\ \psi\right]^{\mathrm{T}}$ – the attitude vector of the MAV, where φ is the roll angle, θ – the pitch angle, and ψ – the flight direction angle. These angles express the angular position of the MAV (of the MAV's body tied frame $oxyz$, ox – the longitudinal axis (pointing the flight direction), oy – the axis oriented towards the right wing, and oz – the axis perpendicular to the plane oxy and downward oriented with respect to the Earth tied frame (Darboux frame - $OXYZ$) having the axis OX tangent to the parallel and East oriented, OY – tangent to the meridian and North oriented, and OZ – the locus vertical line (Zenith oriented).

Let us also consider $\boldsymbol{\omega}_p=\dot{\boldsymbol{\Theta}}=\left[\dot{\varphi}\ \dot{\theta}\ \dot{\psi}\right]^{\mathrm{T}}$ – the angular rates' vector of the MAV relative to the Earth tied frame. We denote with $\boldsymbol{\omega}_b=\left[\omega_x\ \omega_y\ \omega_z\right]^{\mathrm{T}}$ – the vector of MAV's angular rates with respect to the axes of the $oxyz$ frame. The connection between the above presented vectors is: $\boldsymbol{\omega}_p=\dot{\boldsymbol{\Theta}}=W^{-1}\boldsymbol{\omega}_b$, $\boldsymbol{\omega}_b=W\boldsymbol{\omega}_p=W\dot{\boldsymbol{\Theta}}$, where the matrix W has the form:

$$W=\begin{bmatrix}1 & 0 & -\sin\theta\\ 0 & \cos\varphi & \sin\varphi\cos\theta\\ 0 & -\sin\varphi & \cos\varphi\cos\theta\end{bmatrix}. \tag{1}$$

The vector of the resultant moments acting upon the micro aerial vehicle is [14]:

$$\vec{\boldsymbol{M}}_b=\vec{\boldsymbol{M}}_b^a+\vec{\boldsymbol{M}}_b^g+\vec{\boldsymbol{M}}_b^v\cong\vec{\boldsymbol{M}}_b^a, \tag{2}$$

where $\vec{\boldsymbol{M}}_b^a,\vec{\boldsymbol{M}}_b^g,\vec{\boldsymbol{M}}_b^v$ are the aerodynamic, the weight, and the dynamic damping components, respectively; because $\vec{\boldsymbol{M}}_b^g$ and

$\vec{\boldsymbol{M}}_b^v$ have very small values, we can make the approximation $\vec{\boldsymbol{M}}_b \cong \vec{\boldsymbol{M}}_b^a$. We denote with $\vec{\boldsymbol{F}}_b^{al}$ and $\vec{\boldsymbol{F}}_b^{ar}$ – the vectors of the aerodynamic forces produced by the left and the right wings and with $\vec{r}_l$ and $\vec{r}_r$ – the position vectors of the left and right wings' pressure centers with respect to the MAV's mass center (point "*o*"); the following equation connects these variables:

$$\vec{\boldsymbol{F}}_b^a = \vec{\boldsymbol{F}}_b^{al} + \vec{\boldsymbol{F}}_b^{ar}, \vec{\boldsymbol{M}}_b^a = \vec{r}_l \times \vec{\boldsymbol{F}}_b^{al} + \vec{r}_r \times \vec{\boldsymbol{F}}_b^{ar}. \tag{3}$$

The vectors of the aerodynamic forces and moments produced by the two wings have the expressions [1]:

$$\boldsymbol{F}_a^a = \begin{bmatrix} \boldsymbol{F}_d^l \cos\phi_L + \boldsymbol{F}_d^r \cos\phi_R \\ \boldsymbol{F}_d^l \cos\phi_L - \boldsymbol{F}_d^r \cos\phi_R \\ \boldsymbol{F}_l^l + \boldsymbol{F}_l^r \end{bmatrix}, \boldsymbol{M}_a^a = r_a \begin{bmatrix} -\boldsymbol{F}_d^l \cos\phi_L + \boldsymbol{F}_d^r \cos\phi_R \\ -\boldsymbol{F}_d^l \sin\phi_L - \boldsymbol{F}_d^l \cos\phi_R \\ \boldsymbol{F}_d^l - \boldsymbol{F}_d^r \end{bmatrix}, \tag{4}$$

with r_a – the position vector of the two wings' resultant aerodynamic force $\vec{\boldsymbol{F}}_a^a$, $\vec{\boldsymbol{F}}_l^l, \vec{\boldsymbol{F}}_l^r$ – the lift forces, $\vec{\boldsymbol{F}}_d^l, \vec{\boldsymbol{F}}_d^r$ – the wings' drag forces, ϕ_L and ϕ_R – the angles of the wings in the beat plane ("*l*", "*L*" – left, "*r*", "*R*" – right). Using equation (4), one gets [1]: $\boldsymbol{M}_b^a = -\mathcal{M}_\phi r_{ab} \boldsymbol{F}_a^a + \mathcal{M}_\phi \boldsymbol{M}_a^a$, where $\mathcal{M}_\phi$ is the transformation (rotation) matrix of the wing in the beat plane, while r_{ab} is the position vector of the force $\vec{\boldsymbol{F}}_a^a$ relative to the mass center of the micro aerial vehicle.

The dynamics of the MAV's rotation is described by the equation [14]:

$$\frac{\partial \vec{\boldsymbol{K}}_b}{\partial t} + \vec{\boldsymbol{\omega}}_b \times \vec{\boldsymbol{K}}_b = \vec{\boldsymbol{M}}_{\boldsymbol{b}}^{\boldsymbol{a}}, \tag{5}$$

with $\boldsymbol{K}_b = J_b \boldsymbol{\omega}_b$ – the rotation kinetic moment and J_b – the inertia moment matrix with respect to the axes of the MAV's tied frame:

$$J_b = \begin{bmatrix} J_{xx} & 0 & -J_{xz} \\ 0 & J_{yy} & 0 \\ -J_{xz} & 0 & J_{zz} \end{bmatrix}. \tag{6}$$

The dynamics of the MAV's rotation (equation (5)) may be also written as:

$$\dot{\boldsymbol{K}}_b + \boldsymbol{\omega}_b^{\times} \boldsymbol{K}_b = \boldsymbol{M}_{\boldsymbol{b}}^{\boldsymbol{a}}, \tag{7}$$

with $\boldsymbol{K}_b = J_b \boldsymbol{\omega}_b = J_b W \dot{\boldsymbol{\Theta}}, W$ – matrix whose form will be presented later, while $\boldsymbol{\omega}_b^{\times}$ has the form in [1], i.e.:

$$\boldsymbol{\omega}_b^{\times} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} = W \dot{\boldsymbol{\Theta}}^{\times}. \tag{8}$$

The second term from the left side of equation (7) is negligible; equation (7) becomes: $(J_b W)\ddot{\boldsymbol{\Theta}} + (J_b \dot{W})\dot{\boldsymbol{\Theta}} = \boldsymbol{M}_b^{\alpha}$ or

$$m(\boldsymbol{\Theta})\ddot{\boldsymbol{\Theta}} + n(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}})\dot{\boldsymbol{\Theta}} = \boldsymbol{M}_b^{\alpha}, \tag{9}$$

with

$$m(\boldsymbol{\Theta}) = J_b W(\boldsymbol{\Theta}), n(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}}) = J_b \dot{W}(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}}), \tag{10}$$

and $\dot{W}(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}}) = \begin{bmatrix} 0 & 0 & -\dot{\theta}\cos(\theta) \\ 0 & -\dot{\varphi}\sin(\varphi) & \dot{\varphi}\cos(\varphi)\cos(\theta) - \dot{\theta}\sin(\varphi)\sin(\theta) \\ 0 & -\dot{\varphi}\cos(\varphi) & -\dot{\varphi}\sin(\varphi)\cos(\theta) - \dot{\theta}\cos(\varphi)\sin(\theta) \end{bmatrix}$.

2.2. The dynamics of the Micro Aerial Vehicles using the quaternion theory

Now, by using the quaternion theory (Euler rotational theory), the MAV's attitude with respect to the Earth tied frame $(OXYZ)$ may be modified through the rotation of MAV (considered to be rigid solid) around a proper axis (Euler axis) with the angle γ; the components of the quaternion vector $\tilde{\boldsymbol{q}}$ are [4]:

$$q_1 = \varepsilon_1 \sin\frac{\gamma}{2}, q_2 = \varepsilon_2 \sin\frac{\gamma}{2}, q_3 = \varepsilon_3 \sin\frac{\gamma}{2}, q_4 = \cos\frac{\gamma}{2}, \tag{11}$$

where $\varepsilon_1, \varepsilon_2, \varepsilon_3$ are the directory cosines of the vector $\boldsymbol{\varepsilon} = [\varepsilon_1 \quad \varepsilon_2 \quad \varepsilon_3]^{\mathrm{T}}$ relative to the inertial frame; the following equations are verified: $\varepsilon_1^2 + \varepsilon_2^2 + \varepsilon_3^2 = 1, q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1.$ The first three components (q_1, q_2, q_3) of the vector $\tilde{\boldsymbol{q}}$ form the quaternion vector $\boldsymbol{q}$ which verifies the equations: $\boldsymbol{q} = [q_1 \ q_2 \ q_3]^{\mathrm{T}} = \boldsymbol{\varepsilon} \sin\frac{\gamma}{2}$ and $\dot{\boldsymbol{q}} = \boldsymbol{\omega}^{\mathrm{x}} \boldsymbol{q}$, $\boldsymbol{\omega}^{\mathrm{x}} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}$, with $\boldsymbol{\omega}_p = \dot{\boldsymbol{\Theta}} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}$.

Another form for the differential equation of the quaternion's kinematics is [5]:

$$\dot{\boldsymbol{q}} = F(\boldsymbol{q})\boldsymbol{\omega}, \tag{12}$$

where

$$F(\boldsymbol{q}) = \frac{1}{2}\left\{ I_3 + \boldsymbol{q}^{\mathrm{x}} + \boldsymbol{q}\boldsymbol{q}^T - \frac{1}{2}\left[1 + \boldsymbol{q}^{\mathrm{T}}\boldsymbol{q}\right] I_3 \right\}, \boldsymbol{q}^{\mathrm{x}} = \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix}; \tag{13}$$

I_3 is the (3x3) identity matrix.

By time derivation of the equation (12), we get:

$$\ddot{\boldsymbol{q}} = F(\boldsymbol{q})\dot{\boldsymbol{\omega}} + \dot{F}(\boldsymbol{q}, \dot{\boldsymbol{q}})\boldsymbol{\omega}, \tag{14}$$

where $\dot{F}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is obtained by time derivation of equation (13), i.e.:

$$\dot{F}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \frac{1}{2}\left[\dot{\boldsymbol{q}}^{\mathrm{x}} + 2\dot{\boldsymbol{q}}\boldsymbol{q}^{\mathrm{T}} - \dot{\boldsymbol{q}}^T\boldsymbol{q} I_3\right]. \tag{15}$$

By elimination of the vector $\boldsymbol{\omega}$ between equations (12) and (14), we obtain:

$$\ddot{\boldsymbol{q}} = F(\boldsymbol{q})\dot{\boldsymbol{\omega}} + \dot{F}(\boldsymbol{q}, \dot{\boldsymbol{q}})F^{-1}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{16}$$

By left multiplying the equation (16) with $m(\boldsymbol{\Theta})F^{-1}(\boldsymbol{q})$, the following equation results:

$$m(\boldsymbol{\Theta})F^{-1}(\boldsymbol{q})\ddot{\boldsymbol{q}} = m(\boldsymbol{\Theta})\,\dot{\boldsymbol{\omega}} + m(\boldsymbol{\Theta})F^{-1}(\boldsymbol{q})\dot{F}(\boldsymbol{q}, \dot{\boldsymbol{q}})F^{-1}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{17}$$

or, by elimination of $m(\boldsymbol{\Theta})\,\dot{\boldsymbol{\omega}} = m(\boldsymbol{\Theta})\,\ddot{\boldsymbol{\Theta}}$ between equation (9) and (17), having in mind that $\dot{\boldsymbol{\Theta}} = \boldsymbol{\omega} \overset{(12)}{=} F^{-1}(\boldsymbol{q})\dot{\boldsymbol{q}}$, we get:

$$M(\boldsymbol{q})\ddot{\boldsymbol{q}} + N(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} = \boldsymbol{M}_b^a, \tag{18}$$

where

$$M(\boldsymbol{q}) = m(\boldsymbol{\Theta})F^{-1}(\boldsymbol{q}), N(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \left[n(\dot{\boldsymbol{\Theta}}, \dot{\boldsymbol{\Theta}}) - M(\boldsymbol{q})\dot{F}(\boldsymbol{q}, \dot{\boldsymbol{q}})\right]F^{-1}(\boldsymbol{q}). \tag{19}$$

Concluding, the dynamics of the MAV is described either by the equation (9) or the equation (18). Generalizing, the nonlinear dynamics of the MAV may be described by the equation:

$$C(\boldsymbol{y})\ddot{\boldsymbol{y}} + D(\boldsymbol{y}, \dot{\boldsymbol{y}}) = \boldsymbol{u}, \tag{20}$$

with $\boldsymbol{u}-$ the input vector and $\boldsymbol{y}-$ the output vector. Thus,

$$\boldsymbol{u} = M_b^a, \boldsymbol{y} = \boldsymbol{\Theta}, C(\boldsymbol{y}) = C(\boldsymbol{\Theta}) = m(\boldsymbol{\Theta}), D(\boldsymbol{y}, \dot{\boldsymbol{y}}) = D(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}}) = n(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}})\dot{\boldsymbol{\Theta}} \tag{21}$$

or

$$\boldsymbol{u} = M_b^a, \boldsymbol{y} = \boldsymbol{q}, C(\boldsymbol{y}) = C(\boldsymbol{q}) = M(\boldsymbol{q}),\ D(\boldsymbol{y}, \dot{\boldsymbol{y}}) = D(\boldsymbol{q}, \dot{\boldsymbol{q}}) = N(\boldsymbol{q}, \boldsymbol{q})\dot{\boldsymbol{q}}, \tag{22}$$

with $m(\boldsymbol{\Theta})$ and $n(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}})$ having the forms (10) and $M(\boldsymbol{q}), N(\boldsymbol{q}, \boldsymbol{q})$ of forms (19).

3. Design of the Adaptive Control System

3.1. The design of the general control law and of the adaptive control system

To compensate the influence of the unknown or partially known nonlinearities from the dynamics of MAVs, the conventional controllers should be replaced with adaptive architectures based on dynamic inversion and neural networks. The design of the general control law and of the neural network based control system (two variants) is presented in this section of the paper.

In order to obtain a linear control system of the MAV's attitude, we choose the control law [15]:

$$\boldsymbol{u} = C(\boldsymbol{y})\boldsymbol{v} + D(\boldsymbol{y}, \dot{\boldsymbol{y}}), \tag{23}$$

with a proportional-derivative (P.D.) pseudo-cotrol law – provided by a linear dynamic compensator, i.e.:

$$\boldsymbol{v} = k_p \boldsymbol{e} + k_d \dot{\boldsymbol{e}} + \hat{\boldsymbol{v}}_r, \hat{\boldsymbol{v}}_r = \ddot{\boldsymbol{y}}_d, \tag{24}$$

with $\boldsymbol{e} = \boldsymbol{y}_d - \boldsymbol{y}$, while the variables having the index "*d*" are the imposed (desired) ones; the component $\hat{\boldsymbol{v}}_r = \ddot{\boldsymbol{y}}_d$ leads to the convergence $\ddot{\boldsymbol{y}} \to \ddot{\boldsymbol{y}}_d$, issue that will be justified later; k_p and k_d are diagonal matrices having on their principal diagonal the positive elements $k_{pi}, k_{di}, i = \overline{1,3}$.

The system (20) may be described by the nonlinear input-state and state-output equations [12]:

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u}), \boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}), \tag{25}$$

where $\boldsymbol{x}$ is the system's state vector satisfying the conditions of the hypothesis 1 in [17], i.e.: $\boldsymbol{y}^{(r)} = \boldsymbol{h}_r(\boldsymbol{y}, \dot{\boldsymbol{y}}, \boldsymbol{u})$, $\boldsymbol{h}_r \overset{\Delta}{=} \frac{\mathrm{d}^r \boldsymbol{h}}{\mathrm{d}t^r}$ and $\frac{\partial \boldsymbol{h}_i}{\partial \boldsymbol{u}} = 0, i = \overline{0,(r-1)}$, $\frac{\partial \boldsymbol{h}_r}{\partial \boldsymbol{u}} \neq 0$, with $r = 2-$ the relative degree of the system.

According to (20) – equivalent to the system described by (25) and to (23), it yields:

$$\ddot{\boldsymbol{y}} = \boldsymbol{y}^{(r)} = \boldsymbol{h}_r(\boldsymbol{y}, \dot{\boldsymbol{y}}, \boldsymbol{u}) = \boldsymbol{v}. \tag{26}$$

From (26) one obtains:

$$\boldsymbol{u} = \boldsymbol{h}_r^{-1}(\boldsymbol{y}, \dot{\boldsymbol{y}}, \boldsymbol{v}). \tag{27}$$

The automatic control of the vector $\boldsymbol{y}$ (based on the dynamic inversion technique) is achieved by means of the equations (20), (24), and (27); the block diagram of the resulting control system is presented in Fig. 1.

In order to effectively implement the control law (23), $C(\boldsymbol{y})$ and $D(\boldsymbol{y}, \dot{\boldsymbol{y}})$ must be known with high precision; in reality, these functions are not fully known. Let us consider that $\hat{C}(\boldsymbol{y})$ and $\hat{D}(\boldsymbol{y}, \dot{\boldsymbol{y}})$ are the estimates of the matrices $C(\boldsymbol{y})$ and $D(\boldsymbol{y}, \dot{\boldsymbol{y}})$ with the approximation errors:

$$\tilde{C}(\boldsymbol{y}) = \hat{C}(\boldsymbol{y}) - C(\boldsymbol{y}), \tilde{D}(\boldsymbol{y}, \dot{\boldsymbol{y}}) = \hat{D}(\boldsymbol{y}, \dot{\boldsymbol{y}}) - D(\boldsymbol{y}, \dot{\boldsymbol{y}}). \tag{28}$$

As a consequence, the control law (23) becomes:

$$\hat{\boldsymbol{u}} = \hat{C}(\boldsymbol{y})\hat{\boldsymbol{v}} + \hat{D}(\boldsymbol{y}, \dot{\boldsymbol{y}}) = \hat{\boldsymbol{h}}_r^{-1}(\boldsymbol{y}, \dot{\boldsymbol{y}}, \hat{\boldsymbol{v}}), \tag{29}$$

with $\hat{\boldsymbol{v}}$ of form (24). By dynamic inversion, the equation (23) gets the form:

$$\ddot{\boldsymbol{y}} = \boldsymbol{v}, \ \boldsymbol{v} = \hat{\boldsymbol{v}} + \boldsymbol{\varepsilon}, \tag{30}$$

where $\boldsymbol{\varepsilon}$ is the vector of dynamic approximation error (error resulting from the approximation of the function $\boldsymbol{h}_r(\boldsymbol{y}, \dot{\boldsymbol{y}}, \boldsymbol{u})$); $\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{\varepsilon}(\boldsymbol{x}(\boldsymbol{y}), \boldsymbol{u}) = \boldsymbol{\varepsilon}(\boldsymbol{y}, \dot{\boldsymbol{y}}, \boldsymbol{u}) = \boldsymbol{h}_r(\boldsymbol{y}, \dot{\boldsymbol{y}}, \boldsymbol{u}) - \hat{\boldsymbol{h}}_r(\boldsymbol{y}, \dot{\boldsymbol{y}}, \hat{\boldsymbol{u}})$. In steady regime, imposing that $\ddot{\boldsymbol{y}} \to \ddot{\boldsymbol{y}}_d \ (\boldsymbol{\varepsilon} \to 0)$, we obtain the component $\hat{\boldsymbol{v}}_r = \ddot{\boldsymbol{y}}_d$.

To deduce the dependency between $\boldsymbol{u}$ and $\hat{\boldsymbol{u}}$, we use the Taylor series expansion of the function (27) around the set of variables $(\boldsymbol{y}, \dot{\boldsymbol{y}}, \hat{\boldsymbol{v}})$; we get: $\boldsymbol{u} = \boldsymbol{h}_r^{-1}(\boldsymbol{y}, \dot{\boldsymbol{y}}, \boldsymbol{v}) \cong \hat{\boldsymbol{h}}_r^{-1}(\boldsymbol{y}, \dot{\boldsymbol{y}}, \hat{\boldsymbol{v}}) + \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{v}} \boldsymbol{h}_r^{-1}(\boldsymbol{y}, \dot{\boldsymbol{y}}, \boldsymbol{v})\Big|_{\boldsymbol{v}=\hat{\boldsymbol{v}}} (\boldsymbol{v} - \hat{\boldsymbol{v}}) = = \hat{\boldsymbol{u}} + \frac{\mathrm{d}}{\mathrm{d}\hat{\boldsymbol{v}}} \left(\hat{\boldsymbol{h}}_r^{-1}(\boldsymbol{y}, \dot{\boldsymbol{y}}, \hat{\boldsymbol{v}})\right) \cdot \boldsymbol{\varepsilon}$.

To calculate the error $\boldsymbol{\varepsilon}$, buy using the equations (23) and (30), we get:

$$C(\boldsymbol{y})\ddot{\boldsymbol{y}} = C(\boldsymbol{y})\boldsymbol{v} = C(\boldsymbol{y})(\hat{\boldsymbol{v}} + \boldsymbol{\varepsilon}) \cong \hat{\boldsymbol{u}} - D(\boldsymbol{y}, \dot{\boldsymbol{y}}). \tag{31}$$

Replacing $C(\boldsymbol{y})$ and $D(\boldsymbol{y}, \dot{\boldsymbol{y}})$ – forms (28) into equation (31), we obtain:

$$\boldsymbol{\varepsilon} = C^{-1}(\boldsymbol{y})\left[\tilde{C}(\boldsymbol{y})\hat{\boldsymbol{v}} + \tilde{D}(\boldsymbol{y},\dot{\boldsymbol{y}})\right]. \tag{32}$$

With these, we obtain the system in Fig. 2 – equivalent to the one in Fig. 1. To calculate the matrices $\tilde{C}$ and $\tilde{D}$, the method of Lyapunov functions can be used. The system in Fig. 2 (the component $\hat{\boldsymbol{v}}_a$ in this stage of design is neglected) has the input $\boldsymbol{\varepsilon}$ and the output $\boldsymbol{y}$, being described by the state equation:

$$\dot{\mathbf{E}} = \bar{A}\mathbf{E} + B\boldsymbol{\varepsilon}, \mathbf{E} = \left[\mathbf{E}_1 \;\; \mathbf{E}_2\right]^{\mathrm{T}} = \left[\boldsymbol{e} \;\; \dot{\boldsymbol{e}}\right]^{\mathrm{T}}. \tag{33}$$

The variable $\dot{\mathbf{E}}_2$ is successively expressed as: $\dot{\mathbf{E}}_2 = \ddot{\boldsymbol{e}} = \ddot{\boldsymbol{y}}_d - \ddot{\boldsymbol{y}} = \ddot{\boldsymbol{y}}_d - \left(\hat{\boldsymbol{v}} + \boldsymbol{\varepsilon}\right) = \ddot{\boldsymbol{y}}_d - \left(\boldsymbol{v}_{pd} + \ddot{\boldsymbol{y}}_d + \boldsymbol{\varepsilon}\right) \overset{(24)}{=} -k_p\boldsymbol{e} - k_d\dot{\boldsymbol{e}} - \boldsymbol{\varepsilon}$. Thus, the matrices $\bar{A}$ and B have the forms: $\bar{A} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ -I \end{bmatrix} D_c = A + BD_c$, $A = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}$, $B = \begin{bmatrix} 0 \\ -I \end{bmatrix}$, with $D_c = \begin{bmatrix} k_p & k_d \end{bmatrix}$; all the elements of the matrices $\bar{A}$ and B are (3×3) matrices; A is the matrix of the linear system described by equation (26).

We choose the Lyapunov function:

$$V_a = \frac{1}{2}\left(\mathbf{E}^{\mathrm{T}} P\mathbf{E} + \tilde{a}^{\mathrm{T}}\Gamma^{-1}\tilde{a}\right), \tag{34}$$

with P (symmetric and positive defined matrix) – solution of the Lyapunov equation $\bar{A}^{\mathrm{T}}P + P\bar{A} = -Q$; the matrix Q is chosen as a symmetric and positive defined matrix (in particular it can be the unity matrix). Γ is a diagonal matrix whose elements are the gain coefficients, $\tilde{a} = \hat{a} - a$, with $\hat{a}-$ the estimate of the vector a;

$$a = \left[C_{ij} \;\; D_{ij}\right]^{\mathrm{T}}, \hat{a} = \left[\hat{C}_{ij} \;\; \hat{D}_{ij}\right]^{\mathrm{T}}, \tilde{a} = \left[\tilde{C}_{ij} \;\; \tilde{D}_{ij}\right]^{\mathrm{T}}, i, j = \overline{1,3}, \tag{35}$$

where C_{ij} and D_{ij} are the elements of the matrices $C(\boldsymbol{y})$ and $D(\boldsymbol{y},\dot{\boldsymbol{y}})$, $\hat{C}_{ij}$ and $\hat{D}_{ij}$ are the elements of the matrices $\hat{C}(\boldsymbol{y})$ and $\hat{D}(\boldsymbol{y},\dot{\boldsymbol{y}})$, while $\tilde{C}_{ij}$ and $\tilde{D}_{ij}$ are the elements of the matrices $\tilde{C}(\boldsymbol{y})$ and $\tilde{D}(\boldsymbol{y},\dot{\boldsymbol{y}})$, respectively.

By time derivation of the equation (34) and taking into account the equation (33), we get:

$$\dot{V}_a = -\frac{1}{2}\mathbf{E}^{\mathrm{T}}Q\mathbf{E} + \left(\mathbf{E}^{\mathrm{T}}PB\right)\boldsymbol{\varepsilon} + \dot{\tilde{a}}^{\mathrm{T}}\Gamma^{-1}\tilde{a}. \tag{36}$$

By means of the notation $\bar{\mathbf{E}} = \mathbf{E}^{\mathrm{T}}PB$, introducing the parameterization matrix $Y-$ solution of the equation

$$Y\tilde{a} = \mathbf{E}\boldsymbol{\varepsilon}, \tag{37}$$

with $\boldsymbol{\varepsilon}$ of form (32), the equation (36) becomes:

$$\dot{V}_a = -\frac{1}{2}\mathbf{E}^{\mathrm{T}}Q\mathbf{E} + Y\tilde{a} + \dot{\tilde{a}}^{\mathrm{T}}\Gamma^{-1}\tilde{a}. \tag{38}$$

We impose the fulfillment of the condition:

$$Y = -\dot{\tilde{a}}^{\mathrm{T}}\Gamma^{-1}, \tag{39}$$

which, by right multiplying with Γ, gets the form:

$$\dot{\tilde{a}} = -\Gamma Y^{\mathrm{T}}, \tag{40}$$

because $\Gamma = \Gamma^{\mathrm{T}}$. With (39), the equation (38) becomes $\dot{V}_a = -\frac{1}{2}\mathbf{E}^{\mathrm{T}}Q\mathbf{E} \le 0$. Thus, the closed loop system is global asymptotically stable: $\mathbf{E} \to 0$ ($\boldsymbol{e}$ and $\dot{\boldsymbol{e}}$ tend to zero in the same time; $\boldsymbol{y} \to \boldsymbol{y}_d$ and $\dot{\boldsymbol{y}} \to \dot{\boldsymbol{y}}_d$). Also, $\tilde{a} \to 0 (\hat{a} \to a)$.

3.2. The design of the neural network based component of the general control law

From (37), when $\mathbf{E} \to 0 \left(\bar{\mathbf{E}} \to 0\right)$ and $\tilde{a} \to 0$, it results that it is not sure that the approximation error $\boldsymbol{\varepsilon}$ converges to zero and, therefore, it is not sure that $\hat{\boldsymbol{h}}_r \to \boldsymbol{h}_r$. The block diagram of the subsystem for the calculation of the error $\boldsymbol{\varepsilon}$ is presented in Fig. 3. To compensate this error, in the control law an adaptive component $\boldsymbol{v}_a$ provided by a neural network (feed-forward type in this paper) must be added; adding also a robustness component $\bar{\boldsymbol{v}}$, we obtain: $\hat{\boldsymbol{v}}_a = \bar{\boldsymbol{v}} - \boldsymbol{v}_a$. The expression of the component provided by the neural network ($\mathrm{NN_c}$) has been borrowed from [16]:

$$\boldsymbol{v}_a = W^{\mathrm{T}}\sigma\left(V^{\mathrm{T}}\eta\right), \tag{41}$$

with $\sigma-$ the sigmoid function having the form $\sigma(z)=\left(1+e^{-bz}\right)^{-1}, b=\left[b_1\ \ldots\ b_{n_1+1}\right]^{\mathrm{T}}$, n_1- the number of neurons in the neural network's input layer, b_j, $j=\overline{1,n_1+1}-$ the activation potentials of the neurons, $\eta-$ the input vector of $\mathrm{NN_c}$,

$$\eta=\left[1\quad \hat{\boldsymbol{v}}_k^{\mathrm{T}}\quad \boldsymbol{y}_k^{\mathrm{T}}\right]^{\mathrm{T}}=\left[1\quad I_1\quad I_2\quad \ldots\quad I_{n_1}\right]^{\mathrm{T}}, \tag{42}$$

$$\hat{\boldsymbol{v}}_k^{\mathrm{T}}=\left[\hat{\boldsymbol{v}}(t)\ \hat{\boldsymbol{v}}(t-d)\ \hat{\boldsymbol{v}}(t-2d)\right], \boldsymbol{y}_k^{\mathrm{T}}=\left[\boldsymbol{y}(t)\ \boldsymbol{y}(t-d)\ \boldsymbol{y}(t-2d)\right], \tag{43}$$

with I_j, $j=\overline{1,n_1}-$ the inputs of the neural network and $d-$ the simulation step [18]; it can also be regarded as a delay of the signals $\hat{\boldsymbol{v}}(t)$ and $\boldsymbol{y}(t)$; in the simulations area, we considered $d=0.05\mathrm{s}$. The two matrices in the pair (W,V) are the weights of the feed-forward neural network ($\mathrm{NN_c}$) and the solutions of the equations' system [12]:

$$\begin{aligned}\dot{W} &= -k_W\left[2\left(\sigma-\sigma' V^T\eta\right)\bar{\mathbf{E}}+k\left(W-W_0\right)\right], W_0=W(0),\\ \dot{V} &= -k_V\left[2\eta\bar{\mathbf{E}}W^T\sigma'+k\left(V-V_0\right)\right], V_0=V(0),\end{aligned} \tag{44}$$

with $\sigma'=\left.\dfrac{\mathrm{d}\sigma}{\mathrm{dz}}\right|_{z=z_0}$, k_W, k_V and $k-$ positive constants; $\bar{\mathbf{E}}=\mathbf{E}^{\mathrm{T}}PB$ is the training matrix of the $\mathrm{NN_c}$. The elements of the matrix W are the weights of the links between the n_2 neurons of the hidden layer and the n_3 neurons of the output layer; the elements of matrix V are the weights of the links between the n_1 neurons of the input layer and the n_2 neurons of the hidden layer. The dimensions of the two matrices are $(n_2+1)\times n_3$ and $(n_1+1)\times n_2$, respectively.

The robustness component of the control law is calculated by means of the formula borrowed from [19]:

$$\bar{\boldsymbol{v}}^{\mathrm{T}}=k_z\left(\|F\|_f+\bar{F}\right)\|\mathbf{E}\|\frac{\bar{\mathbf{E}}}{\|\bar{\mathbf{E}}\|}+k_e\bar{\mathbf{E}}, \tag{45}$$

with k_z and k_e- positive gains, $F=\begin{bmatrix}W & 0\\ 0 & V\end{bmatrix}$; $\|F\|_f^2=\mathrm{trace}\left\{F^TF\right\}\le\bar{F}$; $\|F\|_f$ is the Frobenius norm of the matrix F, while $\bar{F}$ is the ideal matrix of the neural network. $\bar{\boldsymbol{v}}$ should ensure the boundedness of the neural network's matrices W and V. The robustness component (45) ensures the boundedness of the neural network's matrices W and V. This equation has been deduced by Calise et al. and used in the adaptive control of different aerospace vehicles, using the dynamic inversion and neural networks [18, 27].

The adaptive control law $\boldsymbol{v}_a-$ equation (41), with W and $V-$ solutions of the system (44), together with the robustness component $\bar{\boldsymbol{v}}$, must compensate the approximation error $\boldsymbol{\varepsilon}$ of the function $\boldsymbol{h}_r$. the adaptive control law (41) results from the condition that the closed loop system (having the input $\boldsymbol{v}_a-\left(\bar{\boldsymbol{v}}+\boldsymbol{\varepsilon}+\hat{\boldsymbol{v}}_r\right)$ and the output $\mathbf{E}=\left[\boldsymbol{e}\quad \dot{\boldsymbol{e}}\right]^{\mathrm{T}}$) is global asymptotically stable; for this we have chosen the Lyapunov function $V_d=\dfrac{1}{2}\mathbf{E}^TP\mathbf{E}+f\left(W,V\right)$ and we imposed $\dot{V}_d<0$; this system is described by the equation $\dot{\mathbf{E}}=\bar{A}\mathbf{E}-B\left(\hat{\boldsymbol{v}}_a+\hat{\boldsymbol{v}}_r+\boldsymbol{\varepsilon}\right), \hat{\boldsymbol{v}}_a=\bar{\boldsymbol{v}}-\boldsymbol{v}_a$. Taking into account this equation, with the function $f\left(W,V\right)$ borrowed from [28], and imposing $\dot{f}\left(W,V\right)=0$, we obtained the equations (44) and $\dot{V}_d=-\mathbf{E}^TP\mathbf{E}<0$, P being the solution of the equation $\bar{A}^{\mathrm{T}}P+P\bar{A}=-Q$. Thus, the convergence $\dot{\mathbf{E}}\to 0$ is achieved; $\boldsymbol{e}$ and $\dot{\boldsymbol{e}}$ simultaneously tend to zero ($\boldsymbol{y}\to\boldsymbol{y}_d$ and $\dot{\boldsymbol{y}}\to\dot{\boldsymbol{y}}_d$).

3.3. The design of the reference model

In Fig. 4 we present the block diagram of the adaptive system for the control of the vector $\boldsymbol{y}$, based on the dynamic inversion technique, with linear dynamic compensator (P.D. type), neural network, and reference model (command filter). The reference model is a second order one; it provides the components (signals) $\boldsymbol{y}_d, \dot{\boldsymbol{y}}_d, \ddot{\boldsymbol{y}}_d$; $\boldsymbol{y}, \dot{\boldsymbol{y}}$, and $\ddot{\boldsymbol{y}}$ must converge to $\boldsymbol{y}_d, \dot{\boldsymbol{y}}_d, \ddot{\boldsymbol{y}}_d$. The transfer matrix of the reference model is:

$$H_m(\mathrm{s})=\frac{\omega_0^2}{s^2+2\xi_0\omega_0\mathrm{s}+\omega_0^2}I_3, \tag{46}$$

with $\xi_0=0.7$ and $\omega_0=2\pi f_0$, $f_0=150\,\mathrm{s}^{-1}$.

For the determination of the matrices k_p and k_d, we choose the elements $k_{p1}=k_{p2}=k_{p3}$ and $k_{d1}=k_{d2}=k_{d3}$ as solutions

of the characteristic equations of the liner system in Fig. 4 having the input $\boldsymbol{y}_d$ and the output $\boldsymbol{y}$, with $\hat{\boldsymbol{v}}_a = 0$ and $\boldsymbol{\varepsilon} = 0$;

$$s^2 + k_{di}s + k_{pi} = 0, i = \overline{1,3}. \tag{47}$$

Imposing that these equations have negative roots, the solutions (k_{pi}, k_{di}) result. For example, we can choose $k_{pi} = \omega_0^2$ and $k_{di} = 2\xi_0\omega_0, i = \overline{1,3}$.

In Fig. 5 we present the subsystems for the calculation of the matrices $C(\boldsymbol{y})$ and $D(\boldsymbol{y}, \dot{\boldsymbol{y}})$ using the equations (29) and (30); we will work with the mean values of the states' variations ($\Delta\bar{z}$, $\bar{z}-$ the mean value of z, while $\Delta\bar{z}-$ the variation of the variable $\bar{z}$); we also must remark that in the previous calculations, the signs " Δ " and " $\bar{\bullet}$ " have been omitted. The justification for the usage of the mean values is related to the fact that the physical variables are harmonic functions of period $T = 1/f_0$.

4. Numerical Simulation Results

To study the functionality of the new designed adaptive control systems, we consider a MAV – insect type [2] having the mass $m_0 = 10^{-1}\text{kg}$; the following numerical values also characterizes the MAV:

$$J_{xx} = 2\cdot 10^{-7}\text{Kgm}^2, J_{xz} = 6\cdot 10^{-7}\text{Kgm}^2, J_{yy} = 18\cdot 10^{-7}\text{Kgm}^2, J_{zz} = 8\cdot 10^{-7}\text{Kgm}^2, J_b = \begin{bmatrix} 0.2 & 0 & -0.6 \\ 0 & 1.8 & 0 \\ -0.6 & 0 & 0.8 \end{bmatrix} 10^{-6}\text{kg}\cdot\text{m}^2.$$

Using the formula $k_{pi} = \omega_0^2$ and $k_{di} = 2\xi_0\omega_0, i = \overline{1,3}$, we determine the diagonal matrices k_p and k_d; with these, the matrix D_c is calculated and, then, the matrices $\bar{A}$ and B are determined; then, choosing the matrix $Q = 0.1I_6$, the solution of the Lyapunov equation $\bar{A}^{\mathrm{T}}P + P\bar{A} = -Q$ is obtained. For the neural network (NNc) of the adaptive control systems we have chosen three layers of neurons: an input layer having $n_1 = 6$ neurons, a hidden layer having $n_2 = 7$ neurons, and an output layer having $n_3 = 3$ neurons. The delay d has been considered $d = 0.05\text{s}$. The matrices $W_0 = W(0)$ and $V_0 = V(0)$ are null matrices; the other constants used for the software implementation of the neural network are: $k_W = k_V = 0.5, k = 10, k_z = 0.01, k_e = 0.04$, $\bar{F} = 10, \Gamma = I_{12}$; using these values, we calculate the matrices of the neural network (W and V) –solution of the system (44), the adaptive signal $\boldsymbol{v}_a$ by means of equation (41), with η of form (42), and σ having the form $\sigma(z) = \left(1 + e^{-bz}\right)^{-1}$; the vector of activation potentials is chosen of form: $b = [1 \quad 0.9 \quad 0.8 \quad 0.7 \quad 0.6 \quad 0.5 \quad 0.4]$. The initial values of the vectors $\boldsymbol{y}_d$ and $\dot{\boldsymbol{y}}_d$ are: $\boldsymbol{y}_d(0) = [-2 \quad -3 \quad 2]\text{deg}$ and $\dot{\boldsymbol{y}}_d(0) = [2 \quad 5 \quad 3]\text{deg/s}$; the initial values of the vectors $\boldsymbol{y}$ and $\dot{\boldsymbol{y}}$ are: $\boldsymbol{y}(0) = [-1.3 \quad -5 \quad 2]\text{deg}$ and $\dot{\boldsymbol{y}}(0) = [2 \quad 5 \quad 3]\text{deg/s}$, while the initial quaternion vector is $\boldsymbol{q}^{\mathrm{T}}(0) = \left[-0.55 \quad 0.7 \quad \sqrt{1 - 0.55^2 - 0.7^2}\right]$. We also considered the values $\boldsymbol{y}_c = 0_{3\times 1}$ and $\tilde{a}(0) = [0.0200 \; -0.0575 \; 0.1071 \; 0 \; 0.1580 \; 0.0854 \; -0.1200 \; 0.0384 \; -0.0864 \; 0 \; 0.2106 \; 0.1188]$.

We software implemented in Matlab/Simulink the architecture in Fig. 4 (two variants: a) $\boldsymbol{y} = \Delta\boldsymbol{\Theta}$ and b) $\boldsymbol{y} = \boldsymbol{q}$) and we obtained the time histories in Figs. 6 and 7.

From Fig. 6.a we remark the cancel of the following variables: the variations of the three Euler angles $(\Delta\bar{\varphi}, \Delta\bar{\theta}, \Delta\bar{\psi})$ and of their corresponding angular rates – $\Delta\dot{\bar{\varphi}}, \Delta\dot{\bar{\theta}}, \Delta\dot{\bar{\psi}}$; this means that the Euler angles (roll, pitch, yaw) and their corresponding angular rates tend to their desired values $(\bar{\varphi} \to \bar{\varphi}_d, \bar{\theta} \to \bar{\theta}_d, \bar{\psi} \to \bar{\psi}_d)$. On the other hand, it is confirmed that, the three components of the vectors $\boldsymbol{v}, \hat{\boldsymbol{v}}_{pd}, \hat{\boldsymbol{u}}, \hat{\boldsymbol{v}}, \bar{\boldsymbol{v}}, \boldsymbol{v}_a$ become null in steady regime (in steady regime all the commands are canceled) – Fig. 6.b. Similar conclusions can be drawn for the second variant of the system in Fig. 4 – the characteristics in Fig. 7; beside the variables plotted for the first variant, we also provide here the three components of the quaternion vector $\boldsymbol{q} = [q_1 \quad q_2 \quad q_3]^{\mathrm{T}}$ (Fig. 7.a); the convergences $q_1 \to q_{1d}, q_2 \to q_{2d}, q_3 \to q_{3d}$ are achieved with very small errors.

The adaptive system for the control of MAV's attitude without using the quaternion vector has smaller time regime period than the system for the control of MAV's attitude which uses the quaternion vector. For the first system, the variations of the three Euler angles and of their corresponding angular rates have no overshoots, while, for the second variant, the variations of the angles and angular rates are characterized by overshoots. The characteristics in Figs. 6 and 7 prove the new architecture's stability and its small overshoots.

In our paper, the modeling and the control of the MAV's attitude is achieved either by means of the Euler angles or by the quaternion vector. Comparing our results with the ones in the paper [7], we can remark that in our work the nonlinearities' approximation error $\boldsymbol{\varepsilon}$ and the adaptive command are calculated by using the estimates $\hat{C}(\boldsymbol{y})$ and $\hat{D}(\boldsymbol{y}, \dot{\boldsymbol{y}})$ of the matrices $C(\boldsymbol{y})$ and $D(\boldsymbol{y}, \dot{\boldsymbol{y}})$ or by using $\tilde{C}(\boldsymbol{y})$ and $\tilde{D}(\boldsymbol{y}, \dot{\boldsymbol{y}})$ (the components of the matrix $\tilde{a}$) from equation (20) of the MAV's dynamics; $\tilde{a}$ is

the solution of the equation (40), with $Y-$ the parameterization matrix ensuring that the Lyapunov function (34) is negatively defined (asymptotically stable system). The adaptive command in [7] is modeled by a different neural network (with only one neurons' input layer and one output layer); the output vector consists of the Euler angles. The NN weights' matrices in [7] are not calculated as in this paper with respect to the vector $\overline{\mathbf{E}}$, i.e. with respect to $\mathbf{E}=\left[\boldsymbol{e} \quad \dot{\boldsymbol{e}}\right]^{\mathrm{T}}$, but with respect to the component $\boldsymbol{v}_{pd}$ (the output of a linear dynamic compensator); this leads to higher and faster jumps of the Euler's angles. In our work, the dynamics of the states and command variables are much slower, this being an advantage.

At first sight, we considered in the simulation studies only the initial state errors $\boldsymbol{e}(0)$ and $\dot{\boldsymbol{e}}(0)$. Actually, the disturbances are within $\boldsymbol{e}$ and $\dot{\boldsymbol{e}}$, thus in $\mathbf{E}$ and $\overline{\mathbf{E}}$, i.e. in the weights matrices *W* and *V* – the components of the command (41) and solutions of the equations (44). Also, the disturbances (through $\boldsymbol{e}$ and $\dot{\boldsymbol{e}}$) intervene in the obtaining of the signal $\boldsymbol{\varepsilon}$ (see fig. 3). Therefore, we can conclude that the disturbances affecting the MAV's dynamics are taken into account as well as the uncertainties which can be found in $\boldsymbol{\varepsilon}=\boldsymbol{h}_r-\hat{\boldsymbol{h}}_r$ (with $\boldsymbol{h}_r-$ known or partially known nonlinear function). Because the adaptive control law is designed such that the closed loop system is stable both in the absence and in the presence of the disturbances, the robustness of the system is guaranteed.

5. Conclusions

In this paper we presented two neural network based systems for the adaptive control of the MAV's attitude: by using the attitude vector or the quaternion vector. The dynamic model describing the motion of MAVs with respect to the Earth tied frame is nonlinear and the design of the new adaptive control system (having two variants) is based on the dynamic inversion technique. The inversion error $\boldsymbol{\varepsilon}$ is calculated with respect to the pseudo-control vector $\hat{\boldsymbol{v}}$ and the matrices $\tilde{C}(\boldsymbol{y})$ and $\tilde{D}(\boldsymbol{y},\dot{\boldsymbol{y}})$ which express the deviation of the estimated matrices $\hat{C}(\boldsymbol{y})$ and $\hat{D}(\boldsymbol{y},\dot{\boldsymbol{y}})$ relative to the calculated ones $C(\boldsymbol{y})$ and $D(\boldsymbol{y},\dot{\boldsymbol{y}})$ from the nonlinear description of the MAV, by imposing that the closed loop system is global asymptotically stable (with the usage of the Lyapunov function V_a). The adaptive component $\hat{\boldsymbol{v}}_a$ of the pseudo-control $\hat{\boldsymbol{v}}$ must compensate the error $\boldsymbol{\varepsilon}$. The new adaptive system consists of a linear dynamic compensator, a reference model and a neural network providing the adaptive component $\hat{\boldsymbol{v}}_a$; the neural network is trained by the attitude error vector. We obtained the adaptive system for the attitude's control of the MAVs, its subsystems being the subsystem for the calculation of the approximation error $\boldsymbol{\varepsilon}$ and the subsystem for the calculation of the matrices $C(\boldsymbol{y})$ and $D(\boldsymbol{y},\dot{\boldsymbol{y}})$, in two variants: $\boldsymbol{y}=\Delta\boldsymbol{\Theta}$ and $\boldsymbol{y}=\boldsymbol{q}$. We software implemented in Matlab/Simulink environment all the components of the new adaptive system and we obtained the time histories of the main variables; the obtained characteristics prove the new architecture's correctness, stability, and its small overshoots.

Acknowledgements

This work was partially supported by the project "*Computational Methods in Scientific Investigation of Space*", project no. 72/29.11.2013, of the Romanian National Authority for Scientific Research, Program for Research - Space Technology and Advanced Research – STAR. It is also supported by the grant no. 89/1.10.2015 of the Romanian National Authority for Scientific Research and Innovation, CNCS – UEFISCDI, project code PN-II-RU-TE-2014-4-0849.

References

[1] Deng, X., Schenato, L., Chung, W.C., Sastry, S., Flapping Flight for Biomimetic Robotic Insects: Part. I – System Modeling. IEEE Transactions on Robotics, vol. 22, 2006, pp. 776-786.

[2] Deng, X., Schenato, L., Chung, W.C., Sastry, S., Flapping Flight for Biomimetic Robotic Insects: Part. II – System Modeling. IEEE Transactions on Robotics, vol. 22, 2006, pp. 789-800.

[3] Fei, J., Batur, C., Robust Adaptive Control for MEMS Vibratory Gyroscope. International Journal of Advanced Robotic Systems, vol. 7, no. 4, 2010, pp. 77-82.

[4] Wie, J., Lu, B., Feedback Control Logic for Spacecraft Eigenaxis Rotations under Slew Rate and Control Constrains. Journal of Guidance, Control and Dynamics, vol. 18, no.6, 1995.

[5] Tsiotras, P., Stabilization and Optimality Results for the Attitude Control Problem. Journal of Guidance, Control and Dynamics, vol. 19, no. 14, 1996, pp. 772-779.

[6] Deng, X., Schenato, L., Sastry, S., Model Identification and Attitude Control for a Micromechanical Flying Insect Including Thorax and Sensors Models. IEEE Int. Conf. on Robotics and Automation, Taipei, Taiwan, 2003, pp. 1152-1157.

[7] Cheng, B., Deng, X., A Neural Adaptive Controller in Flapping Flight. Journal of Robotics and Mechatronics, vol. 24, no. 4, 2012, pp. 602-611.

[8] Chirarattananon, P., Perey-Arancibia, N.O., Wood, R.J., Wing Trajectory Control for Flapping-Wing Micro robots Using Combined Repetitive and Minimum-Variance Adaptive Methods. American Control Conference Fairmout Queen Elizabeth, Montreal, Canada, 2012, pp. 3831-3838.

[9] Chirarattananon, P., Ma, K.Y., Wood, R.J., Adaptive Control for Takeoff, Hovering and Landing of a Robotic Fly. International

Conference on Intelligent Robots and Systems, Tokyo, Japan, 2013, pp. 3808-3815.
[10] Mahjoubi, H., Byl, K., Steering and Horizontal Motion Control in Insect-Inspired Flapping-Wing MAVs: The Tunable Impedance Approach. American Control Conference Fairmout Queen Elizabeth, Montreal, Canada, 2012, pp. 901-908.
[11] Rifai, H., Marcand, N., Poulin, G., Attitude and Position Control of a Flapping Micro Air Vehicle. In book: " Aerial Vehicle", 2009, pp. 555-580.
[12] Lungu, M., Sisteme de conducere a zborului (Flight control systems), Sitech Publisher, Craiova, 2008.
[13] Miao, C.X., Fang, J.C., An Adaptive Three-Dimensional Nonlinear Path Following Method for a Fix-Wing Micro Aerial Vehicle, International Journal of Advanced Robotic Systems, vol. 9, no. 206, 2012, pp. 1-8.
[14] Lungu, R., Sepcu, L., Lungu, M., Four-bar mechanism's PD and neural adaptive control for the thorax of the micromechanical flying insects. Journal of Dynamic Systems, Measurement and Control (Transactions of ASME), vol. 137, no. 5, 2015.
[15] Murphy, I.P., Modeling and Control of Flapping Wing Robots. Master of Science Thesis, Virginia Polytechnic Institute University, 2013.
[16] Sitti, M., Piezoelectrically Actuated Four-Bar Mechanism with Two Flexible Links for Micromechanical Flying Insect Thorax, IEEE/ASME Transactions on Mechatronics, vol. 8, no. 1, 2003, pp. 26-36.
[17] Isidori, A., Nonlinear Control Systems. Springer, Berlin, 1995.
[18] Calise, A.J., Johnson, E.N., Johnson, M.D., Corban, J.E., Applications of Adaptive Neural-Networks Control to Unmanned Aerial Vehicles. Journal of Harbin Institute of Technology, vol. 3, no. 11, 2006, pp. 1865-1869.
[19] Wang, Q., Zhang, B., Liu, R., Nonlinear Piezoelectric Behavior of Ceramic Bending Mode Actuators under Strong Electric Fields, Journal of Applied Physics, vol. 86, no. 6, 1999, pp. 3352-3360.
[20] Li, D., Lu, Z., Chen, X., Modeling and Design Adaptive Double Neural Network Controller for Eight-Rotor Micro Aircraft Vehicle, International Journal of Control and Automation, vol. 7, no. 11, 2014, pp. 169-180.
[21] Wagner, T., Valasek, J., Digital Autoland Control Laws Using Quantitative Feedback Theory and Direct Digital Design, Journal of Guidance, Control, and Dynamics, vol. 30, no. 5, 2007, pp. 1399-1413.
[22] Jang, J.O., Jeon, G.J., A parallel neuro-controller for DC motors containing nonlinear friction, Neurocomputing, vol. 30, 2000, pp. 233-248.
[23] Seshagiri, S., Khalil, H.K., Output feedback control of nonlinear systems using RBF neural network, IEEE Trans. Neural Networks, vol. 11, 2000, pp. 69-79.
[24] Wai, R.J., Development of New Training Algorithms for Neuro-Wavelet Systems on the Robust Control of Induction Servo Motor Drive, IEEE Transactions on Industrial Electronics, vol. 49, no. 6, 2002, pp. 1323-1341.
[25] Juang, J.G., Cheng, K.C., Application of neural networks to disturbances encountered landing control," IEEE Trans. Intell. Transp. Syst., vol. 7, no. 4, 2006, pp. 582-588.
[26] Bouabdallah, S., Noth, A. Siegwart R.PID vs. LQ Control Techniques Applied to an Indoor MicroQuadrotor, IEEE International Conference on Intelligent Robots and Systems, 2004, pp. 546-551.
[27] Johnson, E.N., Calise, A.J., Pseudo-Control Hedging: A New Method for Adaptive Control. Navigation Guidance and Control Technology Workshop, November, 1-2, 2000.
[28] Mori, R., Suzuki, S., Neural Network Modeling of Lateral Pilot Landing Control. Journal of Aircraft, no. 46, 2009, pp. 1721-1726.

Figures

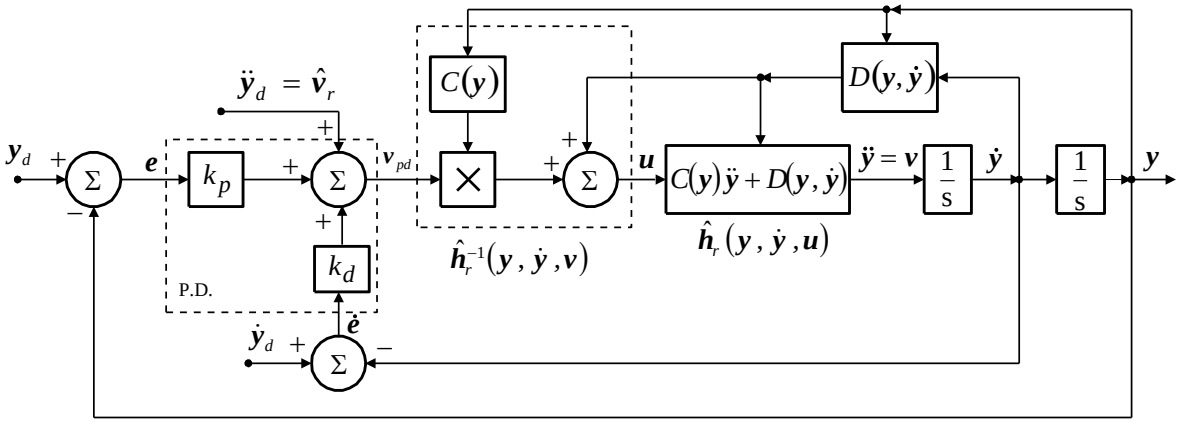


Fig. 1 The automatic system for the control of the vector *y* by using the dynamic inversion technique

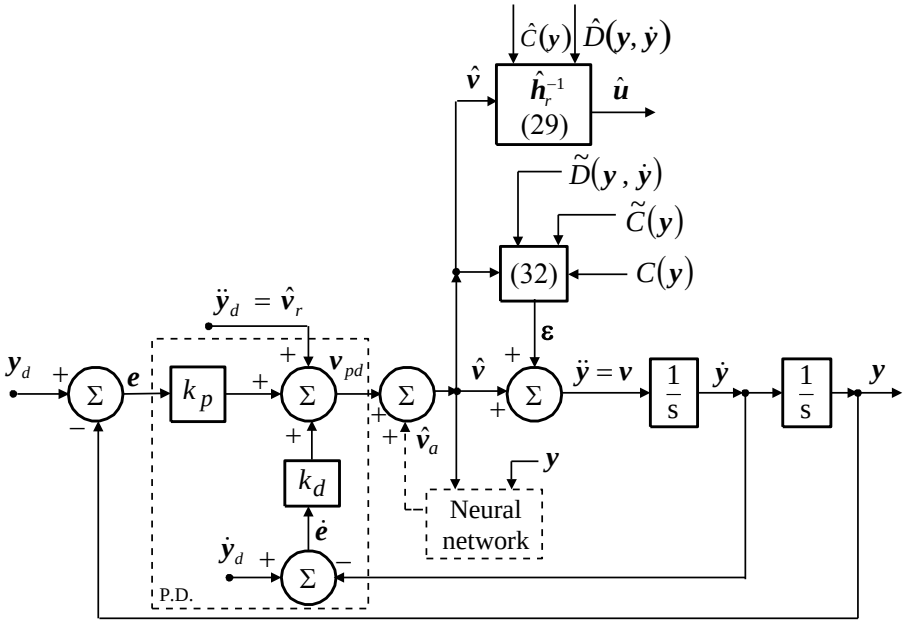


Fig. 2 The equivalent block diagram for the system in Fig. 1

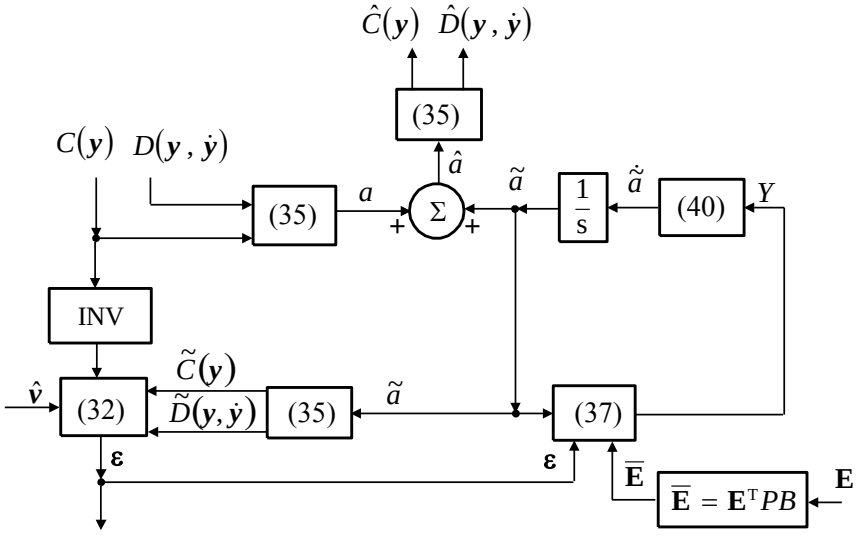


Fig. 3 The block diagram of the subsytem for the calculation of the error ε

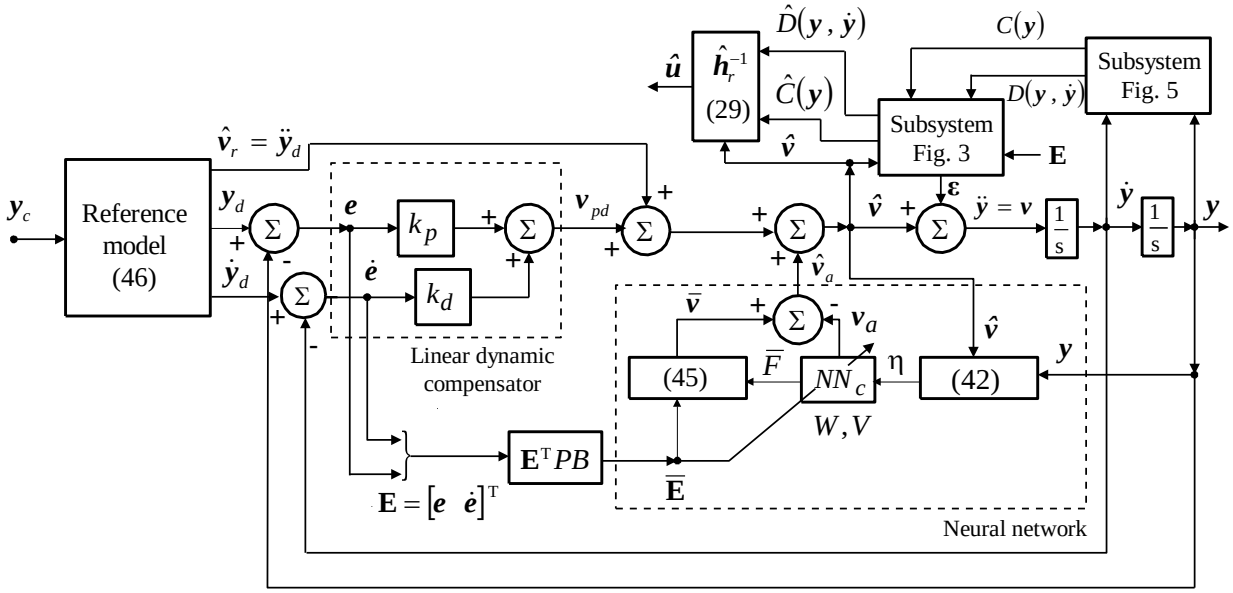
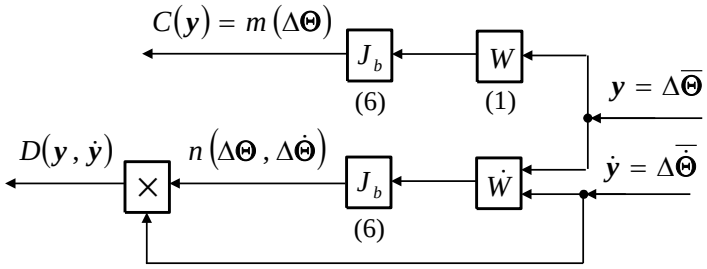
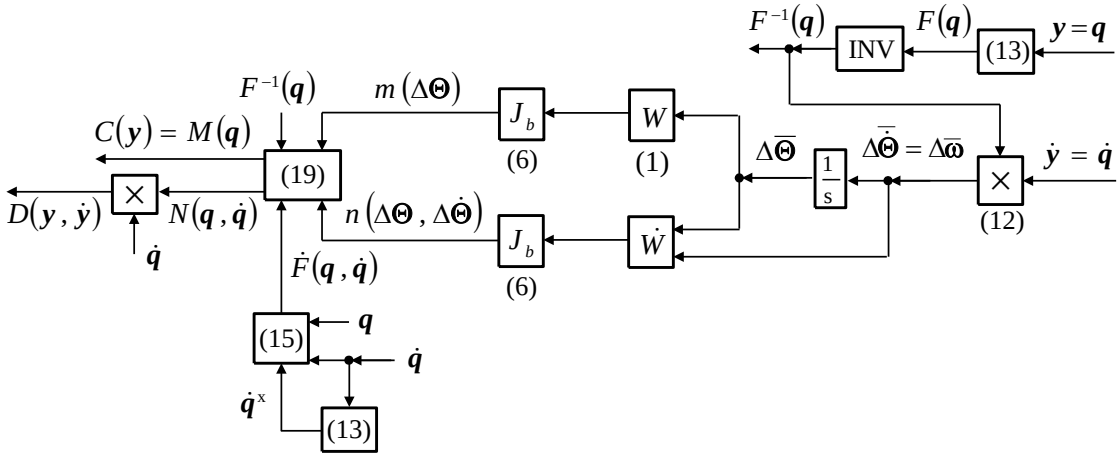


Fig. 4 The block diagram of the adaptive system for the control of the vector y, based on the dynamic inversion technique, with linear dynamic compensator, neural network, and reference model

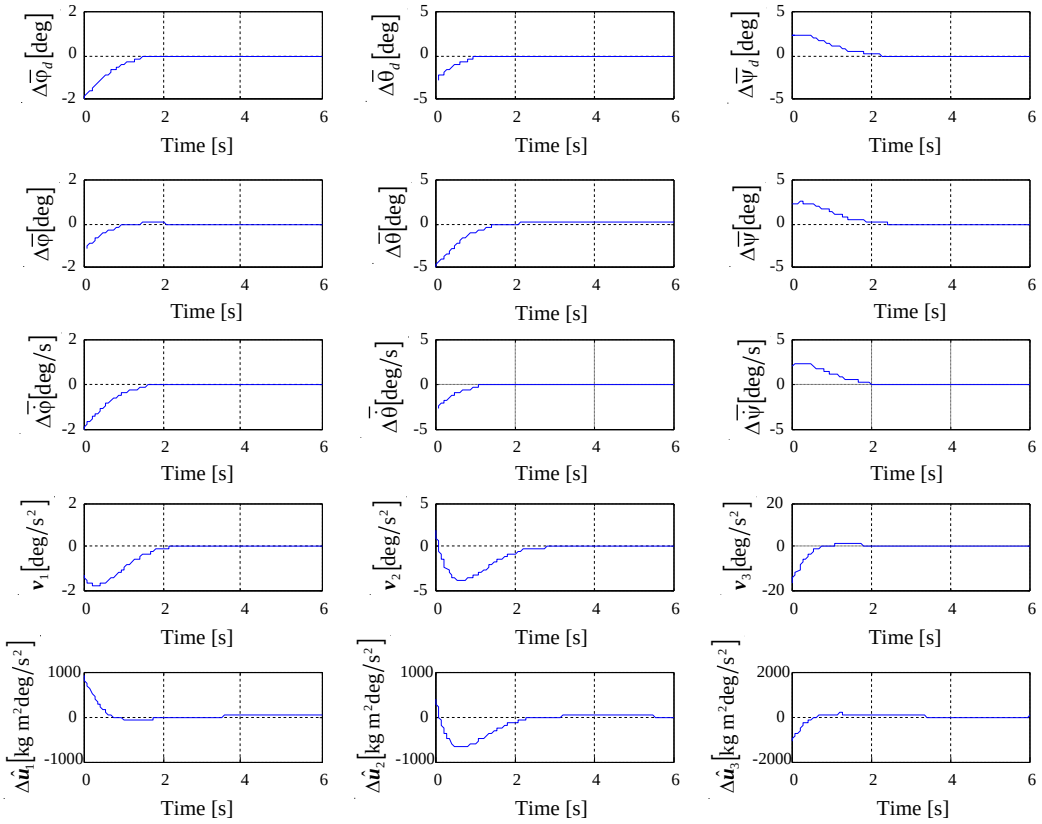


a.

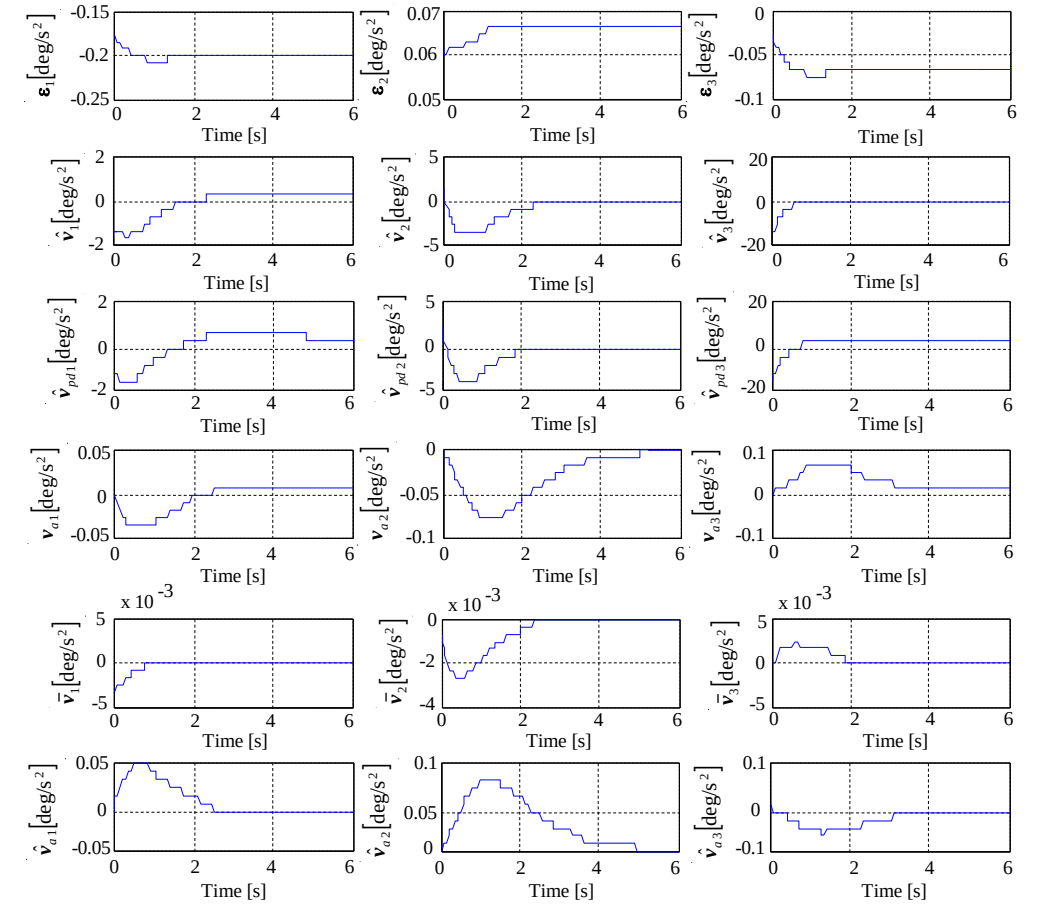


b.

Fig. 5 The subsystems for the calculation of the matrices $C(y)$ and $D(y,\dot{y})$ for: a) $y=\Delta\Theta$; b) $y=q$.

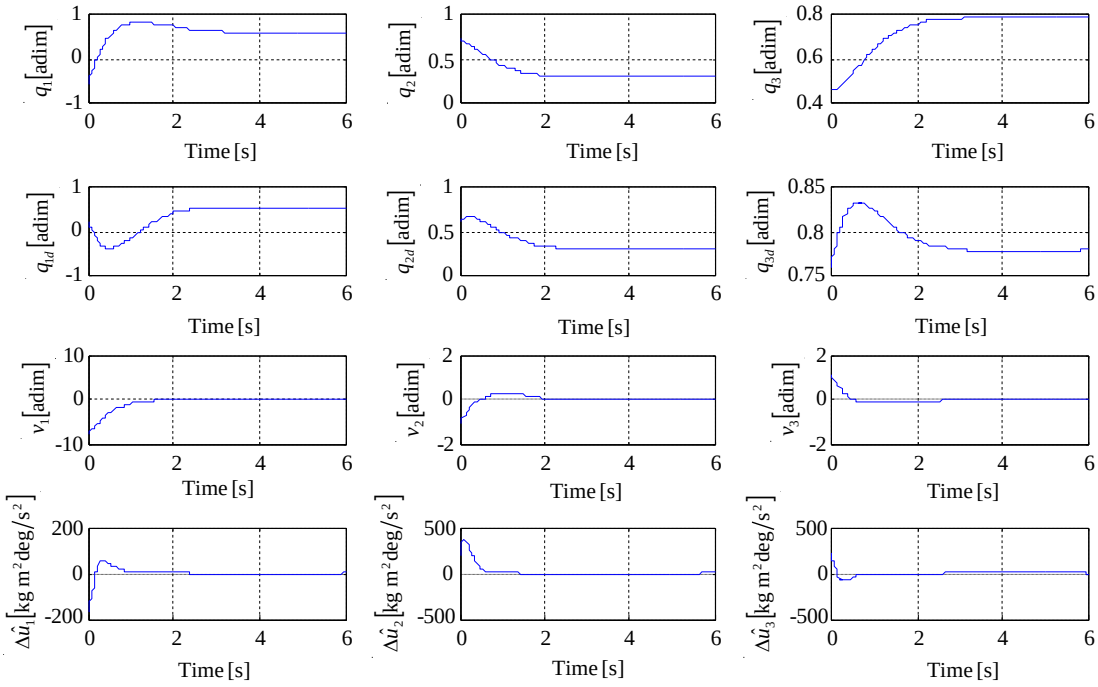


a.

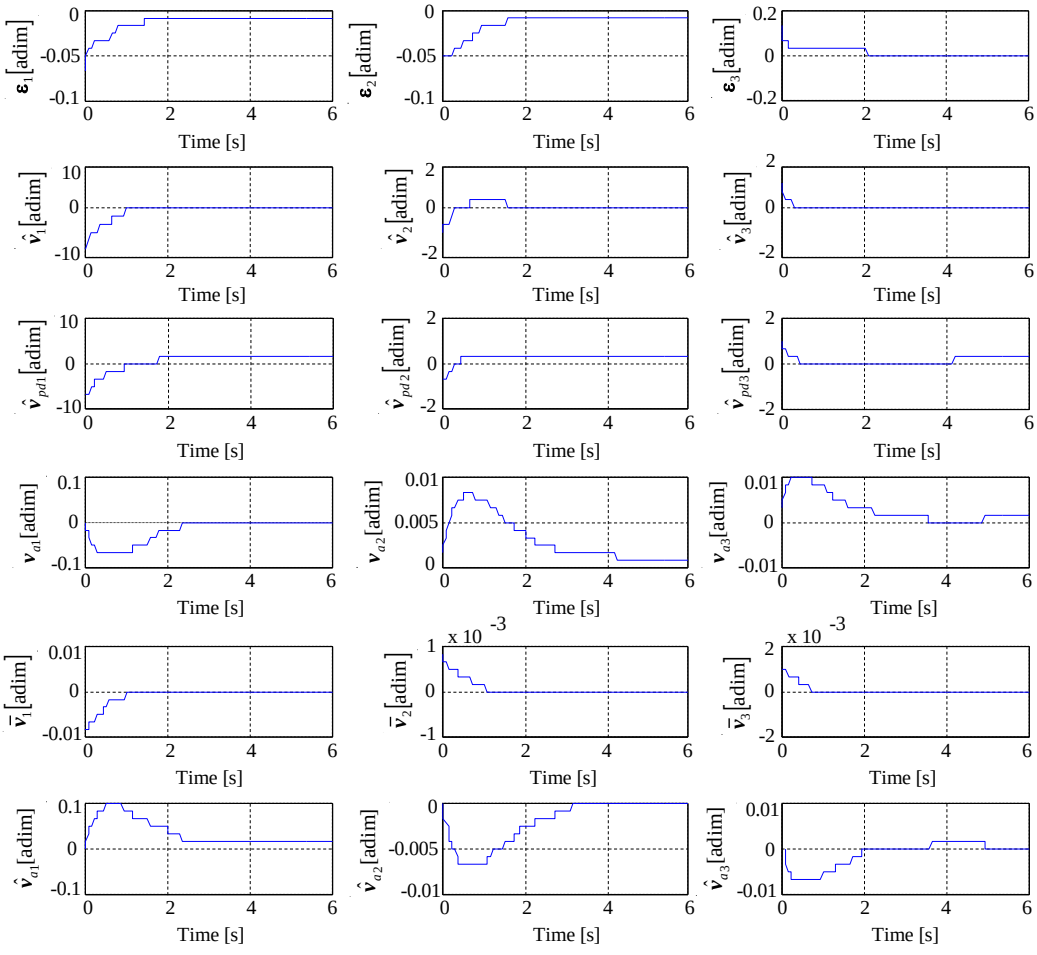


b.

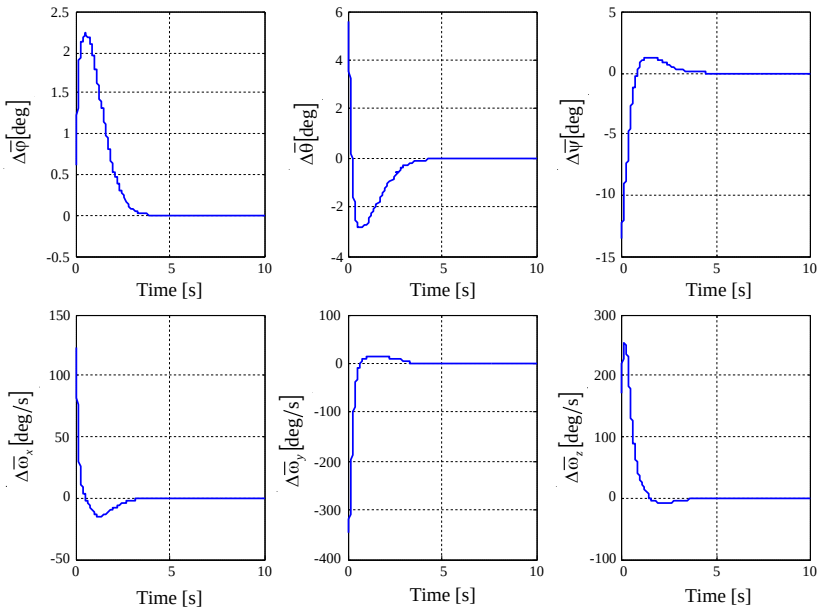
Fig. 6 The time histories of the variables in Fig. 4 for the control of the MAV's attitude $\Delta\Theta$


q_1[adim]
q_2[adim]
q_3[adim]
q_{1d}[adim]
q_{2d}[adim]
q_{3d}[adim]
v_1[adim]
v_2[adim]
v_3[adim]
$\Delta\hat{u}_1$[kg m^{2}deg/s^2]
$\Delta\hat{u}_2$[kg m^{2}deg/s^2]
$\Delta\hat{u}_3$[kg m^{2}deg/s^2]
Time [s]

a.


$\boldsymbol{\varepsilon}_1$[adim]
$\boldsymbol{\varepsilon}_2$[adim]
$\boldsymbol{\varepsilon}_3$[adim]
$\hat{\boldsymbol{v}}_1$[adim]
$\hat{\boldsymbol{v}}_2$[adim]
$\hat{\boldsymbol{v}}_3$[adim]
$\hat{\boldsymbol{v}}_{pd1}$[adim]
$\hat{\boldsymbol{v}}_{pd2}$[adim]
$\hat{\boldsymbol{v}}_{pd3}$[adim]
$\boldsymbol{v}_{a1}$[adim]
$\boldsymbol{v}_{a2}$[adim]
$\boldsymbol{v}_{a3}$[adim]
$\bar{\boldsymbol{v}}_1$[adim]
$\bar{\boldsymbol{v}}_2$[adim]
$\bar{\boldsymbol{v}}_3$[adim]
$\hat{\boldsymbol{v}}_{a1}$[adim]
$\hat{\boldsymbol{v}}_{a2}$[adim]
$\hat{\boldsymbol{v}}_{a3}$[adim]
x 10 -3
Time [s]

b.



c.

Fig. 7 The time histories of the variables in Fig. 4 for the control of the MAV's attitude by using the quaternion vector q