THE PUBLISHING HOUSE
OF THE ROMANIAN ACADEMY

PROCEEDINGS OF THE ROMANIAN ACADEMY, Series A,
Volume 12, Number 2/2011, pp. 133–142

NON-LINEAR ADAPTIVE SYSTEM FOR THE COMMAND OF THE HELICOPTERS PITCH'S ANGLE

Romulus LUNGU[1], Mihai LUNGU[1], Constantin ROTARU[2]

[1]University of Craiova, Faculty of Electrical Engineering, Avionics Department, Blv. Decebal, No.107, Craiova, Romania
[2]Military Technical Academy, Department of Aviation Integrated Systems, George Cosbuc Blv., no. 81-83, Bucharest, Romania
E-mail: Lma1312@yahoo.com, mlungu@elth.ucv.ro

The paper presents a new complex adaptive non-linear system with one input and one output (SISO) which is based on dynamic inversion. The system consists of a dynamic compensator, an adaptive controller and a reference model. Linear dynamic compensator makes the stabilization command of the linearised system using as input the difference between closed loop system's output and the reference model's output. The state vector of the linear dynamic compensator, the output and other state variables of the control system are used for adaptive control law's obtaining; this law is modeled by a neural network. The aim of the adaptive command is to compensate the dynamic inversion error. Thus, the command law has two components: the command given by the linear dynamic compensator and the adaptive command given by the neural network. As control system one chooses the non-linear model of helicopter's dynamics in longitudinal plain. The reference model is linear. One obtains the structure of the adaptive control system of the pitch angle and Matlab/Simulink models of the adaptive command system's subsystems. Thus, characteristics that describe the adaptive command system's dynamics with linear or non-linear actuator are obtained.

Key words: Dynamic inversion; Neural network; Adaptive command; Helicopter; Pitch angle.

1. INTRODUCTION

The complexity and incertitude that appear in the non-linear and instable phenomena are the main reasons that require the projecting of non-linear adaptive structures for control and stabilization; in these cases the linear models are far from a good description of the flying objects' dynamic. Another reason is the non-linear character of the actuators. The observers must be easily adaptable and their project algorithms must allow the state's estimation of the flying object even in the case of their failure or no use of the damaged sensors' signals. In these situations, it's good to use the real time adaptive control based on neural networks and dynamic inversion of the unknown or partial known nonlinearities from the dynamic model of the flying object [1]. The neural network's training is based on the signals from state observers; these observers get information about the control system's error [2, 3, 4].

2. ADAPTIVE COMMAND BASED ON DYNAMIC INVERSION

Let's consider the dynamic system (A) with single input and single output described by the equations

$$\dot{x} = f(x,u),\ y = h(x), \tag{1}$$

with $x(n\times 1)$, $n-$ known, f and $h-$ unknown non-linear functions, u and $y-$ measurable.

One projects an adaptive control law v in rapport with the output using a neural network (NN); NN models a function that depends on the values of input and output of the system (A) at different time moments so that $y(t)$ follows the bordered signal $\bar{y}(t)$. The feedback's linearization may be made by transformation [5]

$$v = \hat{h}_r(y,u), \tag{2}$$

where v is the pseudo-command signal and $\hat{h}_r(y,u)$ – the best approximation of $h_r(x,u) = h_r(x(y),\, u)$. The equation (2) is equivalent with the following one

$$u = \hat{h}_r^{-1}(y,v)\,. \tag{3}$$

If $\hat{h}_r \equiv h_r$, one yields $y^{(r)} = v$; otherwise $\left(\hat{h}_r \neq h_r\right)$

$$y^{(r)} = v + \varepsilon\,; \tag{4}$$

$$\varepsilon = \varepsilon(x,u) = h_r(x,u) - \hat{h}_r(y,u) \tag{5}$$

is the approximation of function h_r (inversion's error). Assessing y to follow $\bar{y}$, v has the form [5, 6, 7]

$$v = \bar{y}^{(r)} + v_{pd} - v_a + \bar{v}, \tag{6}$$

where v_{pd} is the output of the dynamic linear compensator for stabilization, used for the liniarised dynamic (4), with $\varepsilon = 0, v_a$ – the adaptive command that must compensate ε and $\bar{v}$ has, for example, the form [8], [9]

$$\bar{v} = k_z\left(\left\|\hat{Z}\right\|_F + \bar{Z}\right)\frac{\bar{\boldsymbol{E}}}{\left\|\bar{\boldsymbol{E}}\right\|}\left\|\hat{\boldsymbol{E}}\right\| + k_v\bar{\boldsymbol{E}}, \tag{7}$$

with $k_z, k_v > 0$ gain constants, $\left\|\hat{Z}\right\|_F$ – the Frobenius norm of matrix $\hat{Z}, \bar{Z}$ – the ideal matrix of the neural network and $\bar{\boldsymbol{E}} = \hat{\boldsymbol{E}}P\bar{B}$, with $P, \bar{B}$ matrices and $\hat{\boldsymbol{E}}$ – vector. The derivative $\bar{y}^{(r)}$ is introduced for the conditioning of the dynamic error $\tilde{y} = \bar{y} - y$. This derivative is given by a reference model (command filter) [5]. $\bar{y}^{(r)}$ may be cumulated with other signals and it results the component v_r of form (11).

Let's consider $H_d(\mathrm{s})$ – the transfer function of the linear subsystem of A (flying object) with the input u_n and the output y, having to the numerator a p order polynomial and at the denominator a r order polynomial; $p \le r-1$. For this system the paper's authors propose the command structure from Fig. 1, with the linear part described by the equations (8) ÷ (10).

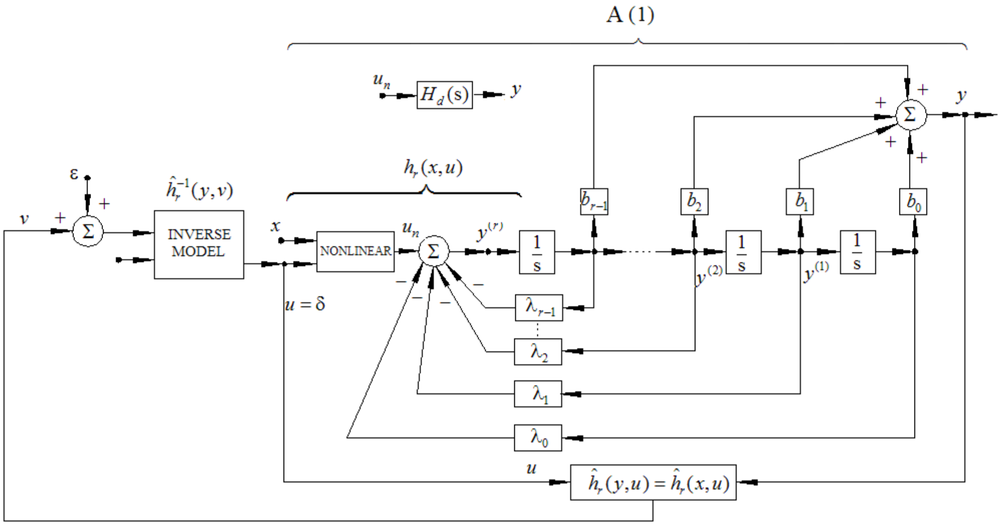

Fig. 1 – The block diagram of the adaptive command system based on dynamic inversion.

Considering

$$Y^{\mathrm{T}} = \left[y\dot{y}\ldots y^{(r-1)}\right],\quad Z^{\mathrm{T}} = \left[v\dot{v}\ldots v^{(p)}\right],\quad \lambda^{\mathrm{T}} = \left[\lambda_0\lambda_1\ldots y_{(r-1)}\right],\quad b^{\mathrm{T}} = \left[b_0 b_1\ldots b_{(p)}\right], \tag{8}$$

with $b_i, i = \overline{0, p}, \lambda_j, j = \overline{0, r-1}$ – the coefficients of the transfer function's numerator and denominator for the system with input u_n and output $y,$ the linear system with input v and output y is described by equation

$$y^{(r)} = -\lambda^{\mathrm{T}} Y + b^{\mathrm{T}} Z + \varepsilon. \tag{9}$$

If $p = 0,$ then $Z = v, b = b_0$ and the previous equation becomes

$$y^{(r)} = -\lambda^{\mathrm{T}} Y + b_0 v + \varepsilon. \tag{10}$$

In the particular case $y^{(r)} = \overline{y}^{(r)}$, one obtains

$$v_r = \frac{1}{b_0}\left(\overline{y}^{(r)} + \lambda^T \overline{Y}\right). \tag{11}$$

The compensator may be described by the state equations

$$\dot{\varsigma} = A_c\varsigma + b_c e, v_{pd} = c_c\varsigma + d_c e, \tag{12}$$

where ς has at least the dimension $(r-1)$,

$$e = \widetilde{y} = c\boldsymbol{e}, \boldsymbol{e}^{\mathrm{T}} = \left[e\ \dot{e}\ \cdots\ e^{(r-1)}\right], c = \left[1\ 0\ 0\ \cdots\ 0\right]_{1\times r}. \tag{13}$$

The state equation of the linear subsystem with the input $(v + \varepsilon)$ and the output y from Fig. 1 is

$$\dot{x} = Ax + b(v + \varepsilon), v = v_{pd} - v_a + \overline{v}, \tag{41}$$

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}_{r\times r},\quad b = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}_{r\times 1}. \tag{15}$$

The stable state $\overline{x}\left(\dot{\overline{x}} = v = \varepsilon = 0\right)$ verifies the equation $A\overline{x} = 0$ and, taking into account equation (14), leads to the equation of the error vector $\boldsymbol{e} \equiv \widetilde{x} = \overline{x} - x$,

$$\dot{\boldsymbol{e}} = A\boldsymbol{e} - bv_{pd} + b\left(v_a - \overline{v} - \varepsilon\right). \tag{16}$$

Introducing in the block diagram from fig. 1 a linear dynamic compensator, a reference model and a non-linear adaptive controller with neural network, one obtains the block diagram from Fig. 2, equivalent with the one from [5].

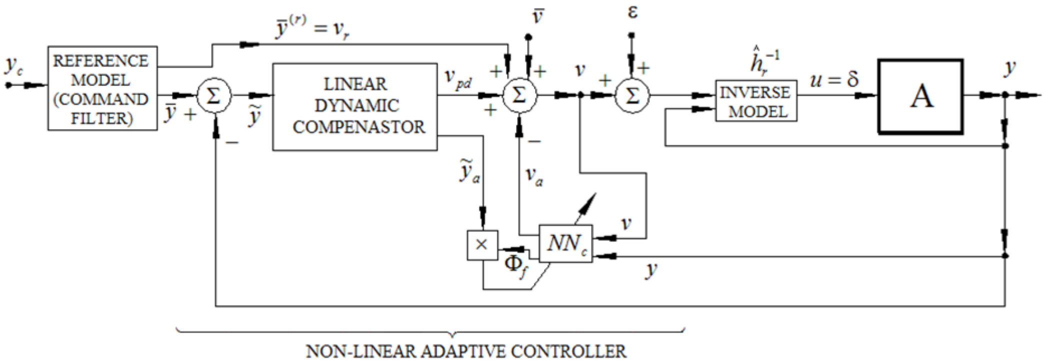

Fig. 2 – Automatic control system with non-linear adaptive controller.

With notations

$$\boldsymbol{E} = \begin{bmatrix} \boldsymbol{e} \\ \varsigma \end{bmatrix}, \overline{A} = \begin{bmatrix} A - d_c bc & -bc_c \\ b_c c & A_c \end{bmatrix}, \overline{b} = \begin{bmatrix} b \\ 0 \end{bmatrix}, \overline{C} = \begin{bmatrix} c & 0 \\ 0 & I \end{bmatrix}, \tag{17}$$

where I is the identity matrix, one obtains

$$\dot{\boldsymbol{E}} = \overline{A}\boldsymbol{E} + \overline{b}\left(v_a - \overline{v} - \varepsilon\right), z = \overline{C}\boldsymbol{E}; \qquad (18$$

A_c, b_c, c_c, d_c from (12) are calculated so that $\overline{A}$ is a Hurwitz matrix.

For the estimation of the vector $\boldsymbol{E}$, the paper's authors propose the introducing in the linear dynamic compensator's structure of a linear state observer of order $(2r-1)$ described by equations (see Fig. 3)

$$\dot{\hat{\boldsymbol{E}}} = \overline{A}\hat{\boldsymbol{E}} + L(z - \hat{z}), \hat{z} = \overline{C}\hat{\boldsymbol{E}}, \tag{19}$$

with the gain matrix L calculated so that matrix $\widetilde{A} = \left(\overline{A} - L\overline{C}\right)$ is stable. Considering $w-$ the sensor's error, $y_m -$ the measured value of y, then $\widetilde{y}_m = \overline{y} - y_m = \widetilde{y} + w$ and the compensator's equations become

$$\dot{\boldsymbol{E}} = \overline{A}\boldsymbol{E} + \overline{b}\left(v_a - \overline{v} - \varepsilon\right) + Gw, z = \overline{C}\boldsymbol{E} + Hw. \tag{20}$$

with $H^{\mathrm{T}} = \begin{bmatrix} 1 & 0 \end{bmatrix}$, $G^{\mathrm{T}} = \begin{bmatrix} -bd_c & b_c \end{bmatrix}$. If the state ς of the compensator is known, one uses a reduced order observer for estimation of the vector $\boldsymbol{e}$ [10]

$$\dot{\hat{\boldsymbol{e}}} = \overline{A}\hat{\boldsymbol{e}} + L_r(z_1 - \hat{z}_1), z_1 = c\hat{\boldsymbol{e}}. \tag{21}$$

The gain matrix L_r is obtained so that the matrix $\widetilde{A} = \left(\overline{A} - L_r c\right)$ is stable. With vectors $\hat{\boldsymbol{e}}$ and ς, the vector $\widehat{\boldsymbol{E}}^{\mathrm{T}} = [\hat{\boldsymbol{e}}\varsigma]$ is obtained. The signal $\overline{\boldsymbol{E}} = \hat{\boldsymbol{E}}^{\mathrm{T}} P\overline{b}$ is used for the neural network's training; the weights $\hat{W}$ and $\hat{V}$ are obtained with equations [11]

$$\dot{\hat{W}} = -\Gamma_W \left[2\left(\hat{\sigma} - \hat{\sigma}'\hat{V}^T\eta\right) \ \hat{\boldsymbol{E}}^{\mathrm{T}} P\overline{B} + k\left(\hat{W} - \hat{W}_0\right)\right], \ \dot{\hat{V}} = -\Gamma_V \left[2\eta\hat{\boldsymbol{E}}^{\mathrm{T}} P\overline{B}\hat{\sigma}' + k\left(\hat{V} - \hat{V}_0\right)\right], \tag{22}$$

where the role of $\overline{B}$ is played by $\overline{b}$. In (22) σ is, for example, the sigmoid function [12], [13]

$$\sigma(z) = \left(1 + \mathrm{e}^{-az}\right)^{-1}, \tag{23}$$

$\hat{\sigma}' = \left.\dfrac{\mathrm{d}\hat{\sigma}(z)}{\mathrm{dz}}\right|_{z=z_0}$ is the Jacobian of vector $\hat{\sigma}, \hat{W}_0$ and $\hat{V}_0 -$ the initial values of weights $\hat{W}, \hat{V}$, $\Gamma_W, \Gamma_V > 0$, $k > 2\left(k_1^2 + \gamma_1^2\left\|P\overline{B}\right\|^2\right)$, $k_1 = k_2\alpha_1 + \left\|P\overline{B}\right\|\gamma_1$, $k_2 = \left\|P\overline{B}\right\| + \left\|\widetilde{P}\overline{B}\right\|$, P and $\widetilde{P} -$ the solutions of Liapunov equations

$$\overline{A}^T P + P\overline{A} = -Q, \quad \tilde{A}^{\mathrm{T}}\tilde{P} + \tilde{P}\tilde{A} = -\tilde{Q} \ . \qquad 241)$$

P from the signal used for the neural network's training is the solution of first equation (24) with $\overline{A} = \left(A - d_c bc\right)$. The programs for the numerical calculus of function (23) and for the solutions' obtaining of the equations (22) and (24) are presented in [4]. Second output of the compensator $\left(\widetilde{y}_a\right)$ is used for obtaining of an error signal that is useful for training of the neural network. From (4) and (6) one yields

$$y^{(r)} = \overline{y}^{(r)} + v_{pd} - v_a + \overline{v} + \varepsilon, \tag{25}$$

$$\widetilde{y}^{(r)} = -v_{pd} + v_a - \overline{v} - \varepsilon. \tag{26}$$

Error ε may be approximated with the output of a linear neural network NN [5, 14]

$$\varepsilon = W^{\mathrm{T}}\Phi(\eta) + \mu(\eta), \ \|\mu\| < \mu^*, \tag{27}$$

where W is the weights' matrix for the connections between the hidden layer and the output layer (NN has 2 layers and one hidden layer), $\mu(\eta)-$ the reconstruction error of the function and $\eta -$ the input vector of NN

$$\eta = \begin{bmatrix} 1 & \bar{v}_d^{\mathrm{T}}(t) & \bar{y}_d^{\mathrm{T}}(t) \end{bmatrix}^{\mathrm{T}}, \tag{28}$$

$$\begin{aligned} \bar{v}_d^{\mathrm{T}}(t) &= \begin{bmatrix} v(t) & v(t-d) & \cdots & v\left(t-(n_1-r-1)d\right) \end{bmatrix}^{\mathrm{T}}, \\ \bar{y}_d^{\mathrm{T}}(t) &= \begin{bmatrix} y(t) & y(t-d) & \cdots & y\left(t-(n_1-1)d\right) \end{bmatrix}^{\mathrm{T}}, \end{aligned} \tag{291)}$$

with $n_1 \geq n$ and $d > 0$; with $\hat{W}$ is the estimation of W, v_a is projected so that

$$v_a = \hat{W}^{\mathrm{T}}\Phi(\eta) \ . \tag{30}$$

3. ADAPTIVE SYSTEM FOR THE COMMAND OF HELICOPTERS' PITCH ANGLE

Lets' consider the case of non-linear dynamic of an experimental helicopter R – 50 with one input and one output (SISO); its dynamic is given by the equation (1) with

$$x^{\mathrm{T}} = \begin{bmatrix} V_x & \omega_y & \theta & \beta & V_z \end{bmatrix}, \quad u = \delta, \quad y = \theta \ , \tag{31}$$

where V_x, V_z are the advance velocity, respectively the vertical velocity, θ and $\omega_y -$ the pitch angle and the pitch angular velocity, $\beta -$ the longitudinal control angle of the main rotor, $\delta -$ the cyclic longitudinal input. Choosing the linearised model of helicopter [15] and, annexing the actuator's equation

$$\delta = \delta_c, \tag{32}$$

the input equation $u = \delta_c$, the output equation $y = \theta$, the system's state equation is obtained

$$\begin{bmatrix} \dot{V}_x \\ \dot{\omega}_y \\ \dot{\theta} \\ \dot{\beta} \\ \dot{V}_z \end{bmatrix} = \begin{bmatrix} X_u & X_q & X_\theta & X_\beta & X_w \\ M_u & M_q & 0 & M_\beta & M_w \\ 0 & 0.999 & 0 & 0 & 0 \\ B_u & -1 & 0 & B_\beta & 0 \\ Z_u & Z_q & Z_\theta & Z_\beta & Z_w \end{bmatrix} \cdot \begin{bmatrix} V_x \\ \omega_y \\ \theta \\ \beta \\ V_z \end{bmatrix} + \begin{bmatrix} X_\delta \\ M_\delta \\ 0 \\ B_\delta \\ Z_\delta \end{bmatrix} \delta . \tag{33}$$

From (33) one yields

$$\dot{y} = \dot{\theta} = \omega_y, \ddot{y} = \ddot{\theta} = M_u V_x + M_q \dot{\theta} + M_\beta \beta + M_w V_z + M_\delta \delta . \tag{34}$$

The second equation (34) must be completed with a non-linear term

$$h_r(x,u) = h_r(x,\delta_c), y^{(r)} = h_r(r,u), h_r \overset{\Delta}{=} \frac{\mathrm{d}^r h}{\mathrm{d}t^r} . \tag{35}$$

The relative degree of the system being $r = 2$ in (8), the output of the reference model has the form

$$\bar{y} = \frac{\omega_{r0}^2}{\mathrm{s}^2 + 2\xi_0\omega_{r0}\mathrm{s} + \omega_{r0}^2} y_c, \xi_0 = 0.7, \omega_{r0} = 10 \,\mathrm{rad/s} . \tag{36}$$

From the analysis of equations (34), one notices that $H_d(\mathrm{s})$ from Fig. 1 has the terms s^2 and $\lambda_1 s$ in the denominator and the term b_0 in the numerator. Choosing $\lambda_1 = b_0$, one gets the transfer function of the system with output y

$$H_d(s) = \frac{b_0}{s\ (s + b_0)} \quad . \tag{37}$$

Equation (10) becomes

$$\ddot{y} = -\lambda_1 \dot{y} + b_0 v + \varepsilon . \tag{38}$$

By elimination of $\ddot{y} = \ddot{\theta}$ between the equation

$$\ddot{y} = \ddot{\theta} = M_u V_x + M_q \dot{\theta} + M_\beta \beta + M_w V_z + M_\delta \delta \tag{39}$$

and the equation (38), one yields

$$M_u V_x + M_q \dot{\theta} + M_\beta \beta + M_w V_z + M_\delta \delta = -\lambda_1 \dot{\theta} + b_0 v + \varepsilon ; \tag{40}$$

From this, one identifies $v = \hat{h}_r(x, \delta)$ and ε ; $x^{\mathrm{T}} = \begin{bmatrix} \theta & \dot{\theta} \end{bmatrix}$; one gets

$$b_0 v = \left(\hat{M}_q + \lambda_1\right)\dot{\theta} + M_\delta \delta = \hat{h}_r(x, \delta), \tag{41}$$

For the calculus of $\delta = \delta_c$ and $\varepsilon,$ equation (33) may be written

$$\begin{bmatrix} \dot{V}_x \\ \dot{\beta} \\ \dot{V}_z \end{bmatrix} = \begin{bmatrix} X_u & X_\beta & X_w \\ B_u & B_\beta & 0 \\ Z_u & Z_\beta & Z_w \end{bmatrix} \begin{bmatrix} V_x \\ \beta \\ V_z \end{bmatrix} + \begin{bmatrix} X_\theta & X_q & X_\delta \\ 0 & -1 & B_\delta \\ Z_\theta & Z_q & Z_\delta \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \delta \end{bmatrix}. \tag{42}$$

One obtains

$$\delta_c = \frac{1}{\hat{M}_\delta}\left[b_0 v - \left(\hat{M}_q + \lambda_1\right)\dot{\theta}\right] = \hat{h}_r^{-1}(x, v); \tag{43}$$

$$\varepsilon = M_u V_x + M_\beta \beta + M_w V_z . \tag{44}$$

The component v_r has the form

$$v_r = \frac{1}{b_0}\left(\dddot{\bar{y}} + b_0 \ddot{\bar{y}}\right) \tag{45}$$

and the control laws (6) and (26) become

$$v = k_p \tilde{y} + k_d \dot{\tilde{y}} + \frac{1}{b_0}\ddot{\bar{y}} + \dot{\bar{y}} - v_a + \bar{v},\ \ddot{\tilde{y}} = -b_0 k_p \tilde{y} - b_0 k_d \dot{\tilde{y}} + \left(b_0 v_a - b_0 \bar{v} - \varepsilon\right); \tag{46}$$

the equation of the closed loop system (the equation of the error's dynamic) is

$$\begin{bmatrix} \dot{\tilde{y}} \\ \ddot{\tilde{y}} \end{bmatrix} = b_0 \begin{bmatrix} 0 & 1 \\ -k_p & -k_d \end{bmatrix} \begin{bmatrix} \tilde{y} \\ \dot{\tilde{y}} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \left(b_0 v_a - b_0 \bar{v} - \varepsilon\right). \tag{47}$$

The characteristic equation of this system is

$$\mathrm{s}^2 + b_0 k_d s + b_0 k_p = 0. \tag{48}$$

Using the notations $b_0 k_d = 2\xi\omega_0$, $b_0 k_p = \omega_0^2$, setting $\xi = 0.7$ and $\omega_0 = 10\,\mathrm{rad/s}$, one obtains k_p , k_d ; $b_0 = 1$.

One considers $\boldsymbol{E} = \boldsymbol{e} = \begin{bmatrix} \tilde{y} & \dot{\tilde{y}} \end{bmatrix}^{\mathrm{T}}$, $\overline{C} = \overline{c} = \begin{bmatrix} 1 & 0 \end{bmatrix}$, $z = \tilde{y} = e$ and $\hat{z} = \hat{e}$; $\hat{\boldsymbol{E}}$ – the observer state (19). The gain matrix L is calculated so that matrix $\widetilde{A} = \left(A - LC\right)$ is stable; $\overline{A}$ is the matrix of system from equation
(47). The component v_a has the form

$$v_a = \hat{W}^{\mathrm{T}} \sigma\left(\hat{V}^{\mathrm{T}} \eta\right) , \tag{49}$$

with σ of form (23), with $a = \begin{bmatrix} 1 & 0.9 & 0.8 & 0.7 & 0.6 & 0.5 & 0.4 \end{bmatrix}$, η of form (28), $n_1 = n = 5, d = 0.05$; $\hat{W}$ and $\hat{V}$ are the solutions of equations (22);

$$\eta^{\mathrm{T}} = \begin{bmatrix} 1 & v(t) & v(t-d) & v(t-2d) & v(t-3d) & y(t) & y(t-d) \end{bmatrix} . \tag{50}$$

$\Gamma_w = 23, \Gamma_v = 12.5, k = 0.115$, P and $\widetilde{P}$ are calculated with (24).

One obtains $\overline{v}$ using (7), where $k_z = 0.8$, $k_v = 0.7, \overline{Z} = 50$. The values of the coefficients from (33) are

$$\begin{aligned}
&X_u = -0.0553,\ X_q = 1.413,\ X_\theta = -32.1731,\ X_\beta = -19.9033,\ X_w = 0.0039,\\
&X_\delta = 11.2579,\ M_u = 0.2373,\\
&M_q = -6.9424,\ M_\beta = 68.2896,\ M_w = 0.002,\ M_\delta = -38.6267,\ Z_u = -0.0027,\\
&Z_q = -0.0236,\ Z_\theta = -0.2358,\\
&Z_\beta = -0.1233,\ Z_w = -0.5727,\ B_u = 0.0101,\ B_\beta = -2.1633,\ B_\delta = -4.2184,\ Z_\delta = 0.0698,\\
&\hat{M}_\delta = 0.5 M_\delta,\ \hat{M}_q = 2 M_q.
\end{aligned} \tag{51}$$

The block diagram of the system for automatic command of the pitch angle is presented in Fig. 3. This structure of the system from Fig. 3 and its project represents one of the authors' contributions in this paper.

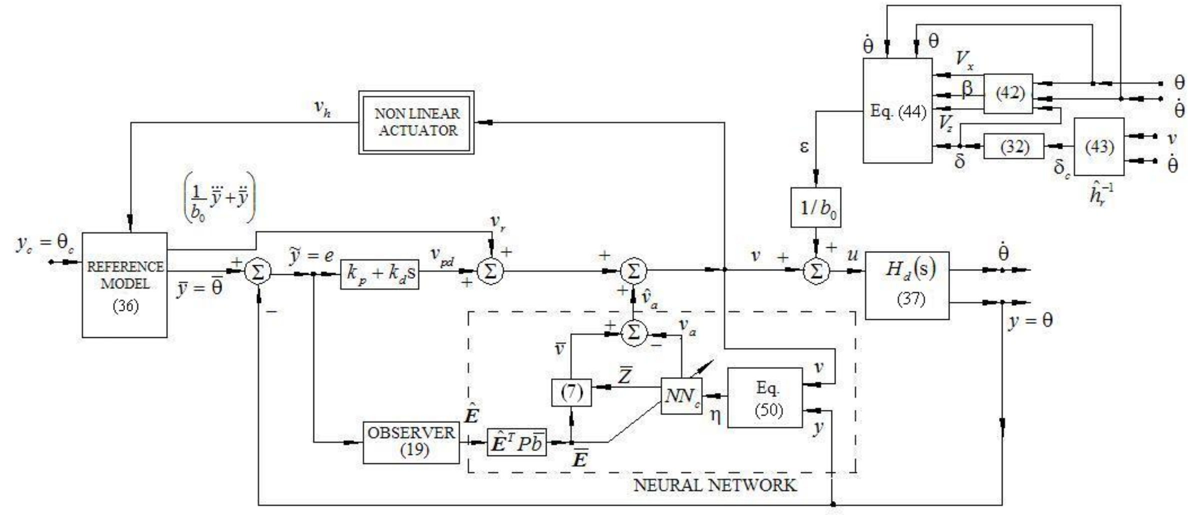

Fig. 3 – Block diagram of the system for automatic command of the pitch angle.

Actuators' characteristics (time delays, nonlinearities with saturation zone) lead to neural network's training difficulties. This is why a block "PCH" is introduced; it limits the adaptive pseudo-control v_a and v by the mean of one component which represents an estimation of the actuator's dynamic (PCH – Pseudo control Hedging). PCH "moves back the reference model" introducing a correction of the reference model's response; it depends on actuator's position [3, 15]. Because the dependence between δ and δ_c is expressed by a non-linear function h_a, one yields $\hat{h}_r(x,\delta_c) \neq \hat{h}_r(x,\hat{\delta})$; it results a difference between two functions $v_h = \hat{h}_r(x,\delta_c) - \hat{h}_r(x,\hat{\delta})$. Taking into account that $\hat{h}_r(x,\delta_c) = \hat{h}_r\left(x, \hat{h}_r^{-1}(x,v)\right) = v$, function v_h becomes

$$v_h = v - \hat{h}_r(x,\hat{\delta}). \tag{52}$$

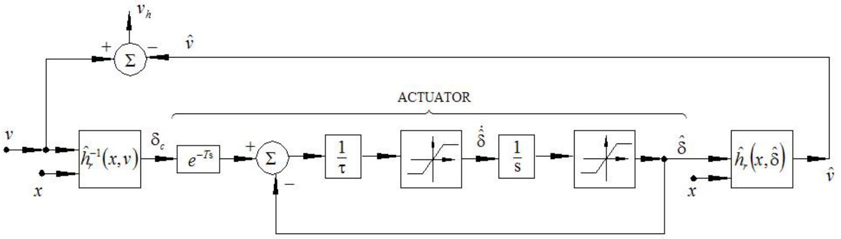

Fig. 4 – The model of the non-linear actuator.

This signal is introduced in the reference model as an additional input [3]; one compares it with $\bar{y}^{(r)}$ inside the reference model and, after integration, it leads to the modify of the signals $\bar{y}$ and $\tilde{y}$. The block diagram of the subsystem formed by (52) and actuator is presented in Fig. 4.

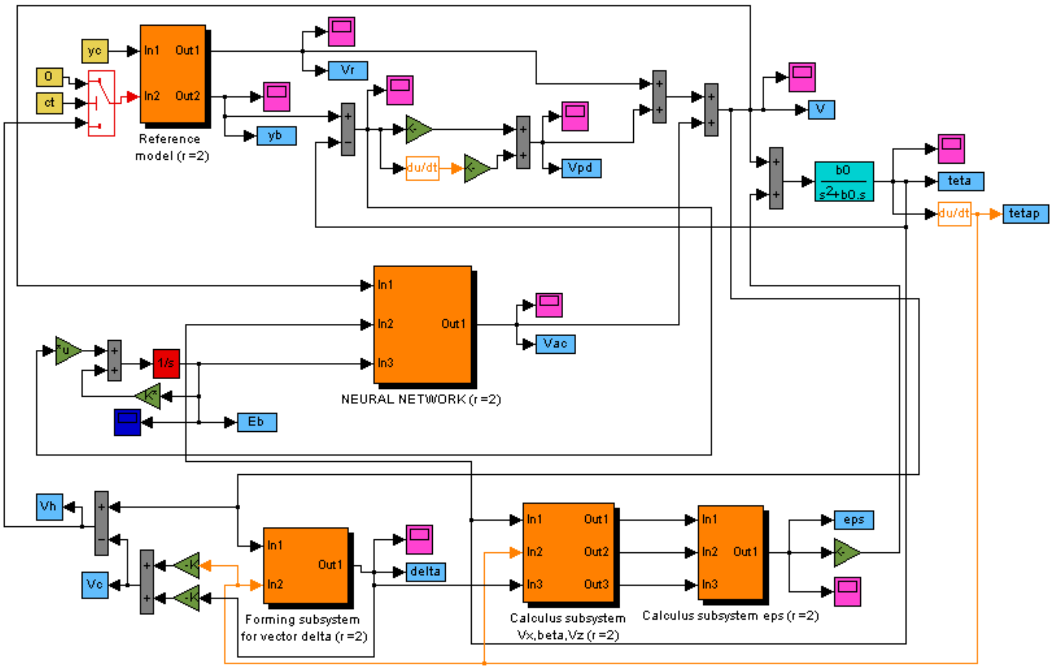

Fig. 5 – The Matlab/Simulink model for the structure from Fig. 3.

In the case of non-linear actuator for the case of the longitudinal movement of the helicopter (equation (33)), the system from Fig. 3 includes the model of non-linear actuator (Fig. 4), in which $x = \dot{\theta}$; the block of calculus for (32) is replaced with the subsystem from Fig. 4. One chooses $T = 0.03\,\mathrm{s}$ and the control limits in position and speed of the actuators $5\,\mathrm{deg}$, respectively 50 deg/s [15]. In Fig. 5, the Matlab/Simulink model for the structure from Fig. 3 is presented; one has chosen $\theta_c = 5\mathrm{deg}$. Each subsystem of the system from Fig. 5 represents a complex Matlab/Simulink model.

In Fig. 6 the functions $\bar{\theta}(t), \theta(t), \varepsilon(t), \hat{v}_a(t), \hat{\delta}(t), \delta(t)$ and $v(t)$ ($\bar{\theta}, \varepsilon, \hat{\delta}$ – with continuous line and $\theta, \hat{v}_a, \delta$ with dashed line) are presented. If the actuator is a linear one $\theta \to \bar{\theta}, \hat{v}_a \to \varepsilon$ (the adaptive component of the command compensates the approximation's error h_r), $\delta \to \hat{\delta}$ and $\nu \to 0$. If the actuator is non-linear, one obtains the time characteristics from Fig. 7; additionally, the characteristics $\nu_h(t)$ and $\dot{\theta}(\theta)$ appear. When $\nu_h = 0$ the actuator is in the saturation state and it works in the linear zone when $\nu_h \neq 0$. The characteristic $\dot{\theta}(\theta)$ (phase portrait of the system) shows that the non-linear system tends to a stable limit cycle. The project of the structure from Fig. 5 and of its subsystems and the obtained graphical time characteristics represent contributions of this paper's authors.

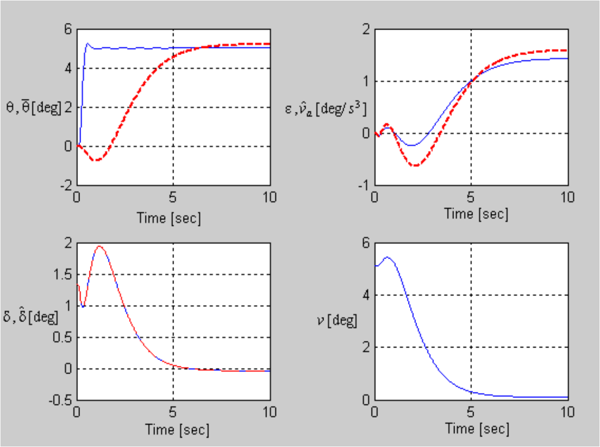

Fig. 6 – Time characteristics in the case of linear actuator's use.

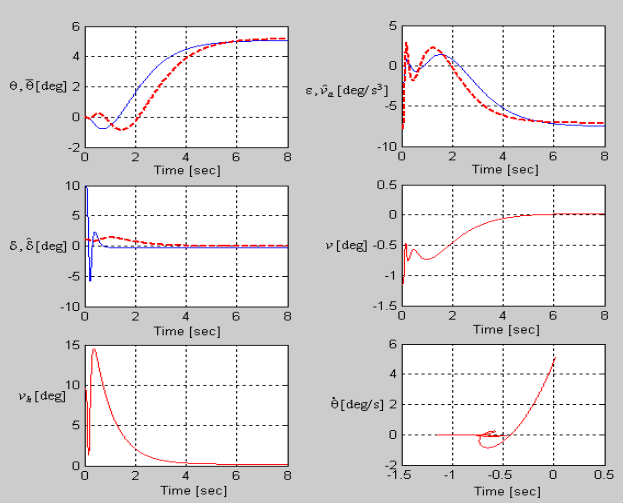

Fig. 7 – Time characteristics in the case of non-linear actuator's use.

4. CONCLUSIONS

The aim of the adaptive command is to compensate the dynamic inversion error. Thus, the command law has two components: the command given by the linear dynamic compensator and the adaptive command given by the neural network. As control system one chooses the non-linear model of helicopter's dynamics in longitudinal plain. The reference model is linear. One obtains the structure of the adaptive control system of the pitch angle and Matlab/Simulink models of the adaptive command system's subsystems. Using these, some characteristics families are obtained; these describe the adaptive command system's dynamics with linear or non-linear actuator.

The authors' contributions in this paper are: 1) the structural block diagram of the adaptive command system from fig. 1, with the linear part described by equations (8) ÷ (10); 2) the block diagram from fig. 2, equivalent with the one from [5], where the linear dynamic model has the structure from fig. 1; the structural block diagram from fig. 3 and its project; 4) the Matlab/Simulink models of the subsystems of the structure from fig. 6; the graphical characteristics from fig. 6 (for the linear actuator case) and from fig. 7 (non-linear actuator case with the model from fig. 5) which describe time evolution of the helicopter's pitch angle, time evolution of the command law's components and offer information regarding the quality and the stability of the non-linear model's dynamic processes for the adaptive command system of the helicopters' pitch angle.

REFERENCES

1. CALISE, A.J., *Flight Evaluation of an Adaptive Velocity Command System for Unmanned Helicopters*, AIAA Guidance, Navigation and Control Conference and Exhibit, Vol. 2, 11–14 August, 2003, Austin, Texas.
2. HOSEINI, S.M., FARROKHI, M., KASHKONEI, A.J., *Robust Adaptive Control Systems Using Neural Networks*, The International Journal of Control, Vol. 3, 2006.
3. JOHNSON, E.N., CALISE, A.J., *Adaptive Guidance and Control for Autonomous Launch Vehicles*, IEEE Aerospace Conference, Biy Ykg, MT, April, 2001.
4. LUNGU, M., *Sisteme de conducere a zborului.* Editura Sitech, Craiova, 2008.

5. CALISE, A.J., HOVAKYMYAN, N., IDAN, M., *Adaptive Output Control of Nonlinear Systems Using Neural Networks*, Automatica, **37**, *8*, pp. 1201–1211 2001.
6. GREGORY, L.P., *Adaptive Inverse Control of Plants with Disturbances*. Stanford University, 2000.
7. HOVAKIMYAN, N., NARDI, F., KIM, N., CALISE, A.J., *Adaptive Output Feedback Control of Uncertain Systems Using Single Hidden Layer Neural Networks*, IEEE Transactions on Neural Networks, **13**, *6*, 2002.
8. SHARMA, M., CALISE, A.J., *Adaptive Trajectory Control for Autonomous Helicopters*, AIAA Guidance, Navigation and Control Conference, 14 – 17 August, 2001, Montreal, Canada.
9. TRIVAILO, P.M., CARN, C.L., *The inverse determination of aerodynamic loading from structural response data using neural networks*, Inverse Problems in Science and Engineering, **14**, *4*, pp. 379–395, 2006.
10. ELTANTAWIE, M.A. *Aplication of neuro-fuzzy reduced order observer in magnetic bearing systems*, Proceedings of the International MultiConference on Engineers and Computer Scientists, Vol. II, Hong Kong, 2010.
11. BALESTRASSI, P.P., POPOVA, E., PAIVA, A.P., MARANGON LIMA, J.W. *Design of experiments on neural network's training for nonlinear time series forecasting.* Elsevier Journal – Neurocomputing, **72**, pp. 1160–1178, 2009.
12. CHEN, X., GAOFENG, W., WEI, Z., SHENG, C., SHILEI, S. *Efficient sigmoid function for neural networks based FPGA design*. Springer Publisher, 2006.
13. FERRARI S., *Smooth function approximation using neural networks*, IEEE Transactions on Neural Networks, **16**, *1*, 2005.
14. SHAO, L., WANG, J., SHAO, S. *Study on the fitting ways of artificial neural networks*, Journal of Coal Science and Engineering, **14**, *2*, 2008.
15. HOVAKIMYAN, N., KIM, N., CALISE, A.J., PARASAD, J.V.R., *Adaptive Output Feedback for High – Bandwidth Control of an Unmanned Helicopter*, AIAA Guidance, Navigation and Control Conference, 6 – 9 August, 2008, Montreal, Canada.

Received May 5, 2010