Non-linear Adaptive Controllers with Linear Dynamic Compensator and Neural Network

LUNGU MIHAI
Avionics Department
University of Craiova, Faculty of Electrotechnics
Blv. Decebal, No.107, Craiova, Dolj
lma1312@yahoo.com
ROMANIA

LUNGU ROMULUS
Avionics Department
University of Craiova, Faculty of Electrotechnics
Blv. Decebal, No.107, Craiova, Dolj
romulus_lungu@yahoo.com
ROMANIA

Abstract: - The paper presents complex adaptive non-linear systems with one input and one output which are based on dynamic inversion. 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 (command filter). 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. For estimation the dynamic compensator’s state, the non-linear adaptive controller may have a linear reduced order observer. 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.

Key-Words: compensator, neural network, dynamic inversion.

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 describe of the flying object’s dynamic. Another reason is the non-linear character of the actuators (because of the saturation and/or their displacement velocity). The observers must be easily adaptable and their project algorithms must allow the estimation of the state 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 train of the neural networks is based on the signals from state observers; these observers get information about the control system’s error [2], [3], [4].

2 Dynamic SISO systems

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

\[ \dot{x} = f(x, u), \quad y = h(x), \]  

with \( x(n \times 1), n \) unknown \( f \) and \( h \) unknown nonlinear functions, \( u \) and \( y \) measurable.

One projects an adaptive control law \( v \) after (in rapport with) the output; the neural network (NN) models a function that depends on the values of input and output of \( A \) at different time moments so that \( y(t) \) follows the finite \( \bar{y}(t) \). The feedback linearization may be made through transformation [5]

\[ v = \hat{h}(y, u), \]
with \( v \) is the pseudo-command signal and \( \hat{h}_r(y,u) \) the best approximation of \( h_r(x,u) = h_r(x(y),u) \).

Equation (2) is equivalent with the following one
\[
    u = \hat{h}_r(y,v),
\]
where
\[
    \hat{h}_r = h_r, \quad \text{one yields } \ y^{(r)} = v; \quad \text{otherwise } (\hat{h}_r \neq h_r)
\]
\[
    y^{(r)} = v + \varepsilon,
\]
is stable. Considering the sensor \( \tilde{y} \),
\[
    y^{(r)} = \tilde{y},
\]
is the approximation of function \( h_r \) (inversion error). Assessing \( y \) to follow \( \tilde{y} \), then \( v \) has the form [5], [6], [7]
\[
    v = y^{(r)} + v_{pd} - v_u + \bar{v},
\]
where \( v_{pd} \) is the output of the dynamic linear compensator for stabilization, used for linearised dynamic (4), with \( \varepsilon = 0, v_u \) the adaptive command that must compensate \( \varepsilon \) and \( \bar{v} \) has the form [8]
\[
    \bar{v} = k_v \left[ \| \hat{Z} \| + \| \hat{Z} \| E \right] + k_v \bar{E},
\]
with \( k_v, k_r > 0 \) gain constants, \( \| \hat{Z} \| \) the Frobenius norm of matrix \( \hat{Z} \), \( \hat{Z} \) - the ideal matrix of the neural network and \( E = \hat{E}P\bar{B} \), with \( \hat{E}, \bar{P} \) and \( \bar{B} \) - matrices. The derivative \( \bar{y}^{(r)} \) is introduced for the conditioning of the dynamic error \( \bar{y} = \bar{y} - y \). 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 \( v_u \) of form (11).

Considering
\[
    y^{(r)} = \begin{bmatrix} y^{(r)} & \cdots & y^{(r)} \end{bmatrix}, \quad Z^{(r)} = \begin{bmatrix} Z^{(r)} & \cdots \end{bmatrix}, \quad \lambda^{(r)} = \begin{bmatrix} \lambda_0 & \lambda_1 & \cdots \end{bmatrix}, \quad b^{(r)} = \begin{bmatrix} b_0 & b_1 & \cdots \end{bmatrix},
\]
with \( b_i, i = 0, p, \lambda_j, j = 0, r - 1 \) the coefficients of the numerator and denominator of the transfer function for the system with input \( u_y \) and output \( y \), the linear system with input \( v \) and output \( y \) is described by equation
\[
    y^{(r)} = -\lambda^{(r)} Y + b^{(r)} Z + \varepsilon.
\]
If \( p = 0 \), then \( Z = v, b = b_0 \) and the previous equation becomes
\[
    y^{(r)} = -\lambda^{(r)} Y + b_0 v + \varepsilon.
\]
In the particular case \( y^{(r)} = \bar{y}^{(r)} \), one obtains
\[
    v_u = \frac{1}{b_0} (\bar{y}^{(r)} + \lambda^{(r)} \bar{y}).
\]

The compensator may be described by state equations
\[
    \xi = A_\xi + b_\xi v, \quad v_{pd} = c_\xi + d_\xi v,
\]
where \( \xi \) has at least dimension \( (r - 1) \),
\[
    e = c_\xi e^T = \begin{bmatrix} e & \dot{e} & \cdots & e^{(r-1)} \end{bmatrix},
\]
\[
    c = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix},
\]
The state equation of the linear subsystem with input \( (v + e) \) and output \( y \) is
\[
    \dot{x} = Ax + b(v + e), v = v_{pd} - v_u + \bar{v},
\]
where
\[
    A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}.
\]
The stable state \( \bar{x}(\bar{y} = v = 0) \) verifies equation \( A\bar{x} = 0 \) and, taking into account (14), leads to the equation of the error vector \( e \equiv \bar{x} = \bar{x} - x \),
\[
    \dot{e} = Ae - bv_{pd} + b(v_u - \bar{v} - \varepsilon).
\]

With notations
\[
    E = \begin{bmatrix} e & \xi \end{bmatrix}, \quad \bar{E} = \begin{bmatrix} A - dLc & -bcL \\ b_c - \lambda \end{bmatrix}, \quad \bar{b} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}, \quad \bar{I} = \begin{bmatrix} 1 & 0 \end{bmatrix},
\]
where \( I \) is the identity matrix with the \( \xi \)’s dimension, one obtains
\[
    \dot{E} = \bar{E}E + \bar{b}(v_u - \bar{v} - \varepsilon), \quad z = \bar{CE} ;
\]
\[
    A_\xi, b_a, c_a, d_a \text{ from } (12) \text{ are calculated such that } \bar{E} \text{ be a Hurwitz matrix}.
\]

For the estimation of the error vector \( e \) one may use a linear state observer of order \( (2r - 1) \) described by equations
\[
    \hat{e} = \bar{a} \hat{e} + L(z - \hat{z}), \hat{z} = \bar{CE},
\]
with the gain matrix \( L \) calculated so that matrix \( \bar{A} = (\bar{A} - L\bar{C}) \) is stable. Considering \( w \) - the sensor measure error, \( y_u \) - the measured value of \( y \), then \( \hat{y}_u = \tilde{y} - y_u = \tilde{y} + w \) and the compensator’s equations become
\[
    \hat{E} = \bar{AE} + \bar{b}(v_u - \bar{v} - \varepsilon) + Gw, z = \bar{CE} + Hw.
\]
If state \( \xi \) of the compensator is known, one uses a reduced order observer for estimation of vector \( e \)
\[
    \hat{e} = \bar{a} \hat{e} + \hat{L} \hat{z} = \bar{CE}.
\]

The gain matrix \( L \) is obtained so that matrix \( \bar{A} = (\bar{A} - L, c) \) is stable. With vectors \( \hat{e} \) and \( \xi \) vector \( \hat{E} = \begin{bmatrix} \hat{e} & \xi \end{bmatrix} \) is obtained.

The signal \( \hat{E} = \hat{E}^T P\bar{B} \) is used for neural network’s adapting; the weights \( \hat{w} \) and \( \hat{v} \) are obtained with equation

\[
    \hat{y} = A_\hat{y} + b_\hat{y} v, \quad \hat{x} = A_\hat{x} + b_\hat{x} v,
\]
where \( A_\hat{y}, b_\hat{y} \) are matrices of the command filter and \( b_\hat{y} \) is the pseudo-command signal.
\[
\dot{W} = -\Gamma_w \left[ 2(\sigma - \sigma' \eta^T)E^T P \dot{B} + k(\ddot{W} - \dot{W}_0) \right],
\]
\[
\dot{\sigma} = \frac{d\sigma(\tau)}{d\tau} \text{ is the Jacobian of vector } \sigma, \dot{W}_0 \text{ and } \dot{V}_0
\]
where the role of \( \dot{B} \) is played by \( \dot{\sigma} \). In (22) \( \sigma \) is the sigmoid function
\[
\sigma(\tau) = \frac{1}{1 + e^{-\omega \tau}},
\]
with the initial values of weights \( \dot{W}, \dot{V}, \Gamma_w, \Gamma_v > 0 \),
\[
k > 2(k^2 + \gamma_1)P^T P \geq 0, \quad k = k_1 a_1 + [P^T P]_{11}^T, \quad k_2 = [P^T P]_{33}.
\]
\( P \) and \( \dot{P} \) – the solutions of Liapunov equations
\[
A^T P + PA = -Q, A^T \dot{P} + P \dot{A} = -\dot{Q}.
\]

Second output of the compensator \( (\ddot{y}) \) is used for obtaining of an error signal that is useful for adapting of the neural network’s weights.

From (4) and (6) one yields
\[
y'(\tau) = \dot{y}(\tau) + \nu_p - \nu_0 + v + \epsilon,
\]
equivalent with the dynamic error’s equation
\[
\ddot{y}(\tau) = -\dot{v}_p + \dot{v}_0 + v - \nu.
\]

### 3 Adaptive system for the helicopter pitch angle command

Let’s consider the case of nonlinear dynamic of an experimental helicopter \( R-50 \) with one input and one output; its dynamic is (1) with
\[
x'_r = [V_r, \omega, \theta, \beta, V_r, \delta_x, \theta],\ u = \delta, \ y = \theta,
\]
where \( V_r, V_x \) are the advance velocity, respectively the vertical velocity, \( \theta \) and \( \omega \) – 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 \[9\] and, annexing the actuator’s equation
\[
\tau_0 + \delta = \delta_x,
\]
the new state vector \( x'_r = [\dot{V}_x, \omega, \theta, \beta, V_x, \delta_x] \)

input \( u = \delta_x \), output \( y = \theta \) and state equation are obtained
\[
\begin{bmatrix}
\dot{V}_x \\
\dot{\omega} \\
\dot{\theta} \\
\dot{\beta} \\
\dot{V_x} \\
\dot{\delta_x}
\end{bmatrix} =
\begin{bmatrix}
X_1 \\
M_1 \\
M_2 \\
M_3 \\
V_x \\
\delta_x
\end{bmatrix} +
\begin{bmatrix}
\nu_1 \\
\nu_2 \\
\nu_3 \\
\nu_4 \\
\nu_5 \\
\nu_6
\end{bmatrix},
\]
with \( \nu = W^T \Phi(\eta) + \mu(\eta). \mu \leq \mu' \),

where \( W \) is the weights’ matrix for the connections between layer 2 and layer 3 (NN has 2 layers), \( \mu(\eta) \) – the reconstruction error of the function and \( \eta \) – the input vector of NN
\[
\eta = [v'(t), v'(t)],
\]
where
\[
\begin{bmatrix}
v'(t) \\
v'(t - d)
\end{bmatrix}\]
The transfer function for the system with input \( u \) and output \( y \) has the form

\[
H_y(s) = \frac{b_0}{s^3 + \lambda_2 s^2 + \lambda_1 s + \lambda_0} = \frac{b_0}{s^3 + (\lambda_2 + \lambda_1 + \lambda_0) s^2 + (\lambda_1 + \lambda_0) s + \lambda_0};
\]  
\( 38 \)

One has chosen \( \lambda_1 = b_0 \). According to (8) it results \( Z = v, \lambda_0 = \lambda_1 = 0 \) and equation (9) becomes

\[
\ddot{y} = -b_0 \dot{y} + b_0 y + \varepsilon.
\]  
\( 39 \)

By elimination of \( \ddot{y} = \ddot{\theta} \) between the second equation (34) and (35) one yields

\[
\ddot{y} = \ddot{\theta} = M_s \dot{V}_s + M_s \dot{\beta} + M_s \dot{V}_s - \frac{M_s}{\tau} \ddot{\delta} + \frac{M_s}{\tau} \ddot{\delta} + \frac{M_s}{\tau} \ddot{\delta} +
\]  
\( 40 \)

Replacing \( \dot{y} \) given by the second equation (34) and \( \ddot{y} \) given by (40) in (39) one may identify \( \nu \) and \( \varepsilon \) as follows: 1) \( \nu \) must have the form \( \nu = h_r(x,u) = h_r(x,\delta, \dot{\delta}) \), where \( x = [\theta, \dot{\theta}, \ddot{\theta}] \); \( h_r \) depends on some of state vector \( \theta \)'s components; 2) \( \varepsilon \) depends on the other terms that contain \( V_s, \dot{V}_s, \beta, \dot{\beta}, V_s, \dot{V}_s, \dot{\beta} \). One obtains

\[
\delta = \nu_r \frac{\tau}{M_s} \begin{bmatrix} b_0 \nu + M_s \dot{\beta} + \dot{\delta} \end{bmatrix} = \dot{\theta} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},
\]  
\( 41 \)

\[
\varepsilon = (M_s + 2b_0)(M_s \dot{V}_s + M_s \dot{\beta} + M_s V_s + M_s \dot{\delta}) + \frac{M_s}{\tau} \ddot{\delta}.
\]  

Because \( \lambda^T = [0 \hspace{1cm} 0 \hspace{1cm} b_0] \) and \( \bar{Y} = [\ddot{y} \hspace{1cm} \dot{y} \hspace{1cm} y] \), with (11) one obtains

\[
\nu_r = \frac{1}{b_0} (\ddot{y} + b_0 \dot{y}).
\]  
\( 42 \)

By choosing for the compensator a proportional - derivative control law \( v_{pr} = k_\rho \ddot{y} + k_\delta \dot{y} \), the law (6), in which the role of \( \ddot{y} \) is played by \( \dot{v} \), becomes

\[
v = k_\rho \ddot{y} + k_\delta \dot{y} + \frac{1}{b_0} (\ddot{y} + b_0 \dot{y} - v) + v_r.
\]  
\( 43 \)

Taking into account (4) and (43) it results

\[
\ddot{y} = -b_0 k_\rho \ddot{y} - b_0 k_\delta \dot{y} - b_0 \dot{y} + (b_0 \nu_v - b_0 v - \varepsilon). \quad 44
\]

The system that describes the dynamic of the error \( e^T = [\ddot{y} \hspace{1cm} \dot{y} \hspace{1cm} y] \) is

\[
\begin{bmatrix} \ddot{y} \\ \dot{y} \\ y \end{bmatrix} =
\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} \ddot{y} \\ \dot{y} \\ y \end{bmatrix} + 0 b_0 \nu_v - b_0 v - \varepsilon.
\]
\( 45 \)

For the calculus of coefficients \( b_0, k_\rho, k_\delta \), one sets values for the roots of the characteristic equation \( s^3 + b_0 s^2 + b_0 k_\rho s + b_0 k_\delta = 0; \) (46) that means to set the coefficients of this equation to be Višnegradska type; \( b_0 = 1 \).

From the expression of the control law and (12) it results that the linear dynamic compensator has order 1 with only a state variable \( S = \ddot{y} = \varepsilon \). As a consequence, conform to equation (17), one

![Fig.3. Block diagram of the system for automatic command of the pitch angle](image-url)
derivates \( \dot{\mathbf{V}}_s, \dot{\mathbf{V}}_w, \dot{\mathbf{B}} \) must be known. The first, the forth and the fifth equations (33) lead to
\[
\begin{align*}
\dot{\mathbf{V}}_s &= X_s \mathbf{V}_s + X_s \mathbf{B}_s + X_s \mathbf{B}_s \mathbf{v}_s + X_s \mathbf{z}_s \delta, \\
\dot{\mathbf{B}} &= \mathbf{B}_s \mathbf{V}_s - \mathbf{B}_s \mathbf{B}_s + \mathbf{B}_s \delta, \\
\dot{\mathbf{V}}_w &= Z_w \mathbf{V}_w + Z_w \mathbf{z}_w + Z_w \mathbf{B}_s + Z_w \mathbf{V}_s + Z_w \mathbf{z}_s \delta.
\end{align*}
\]
(49)

The values of the coefficients from (33) are
\[
\begin{align*}
X_s &= -0.0553, X_s &= 1.413, X_s &= -32.1731, X_s &= -19.9033, \\
X_s &= 0.0039, X_s &= 0.2373, X_s &= -6.9424, X_s &= 68.2896, \\
X_s &= 0.002, B_s &= 0.0101, B_s &= -2.1633, Z_s &= -0.0027, \\
Z_s &= -0.0236, Z_s &= -0.2358, Z_s &= -0.1233, Z_s &= -0.5727, \\
Z_s &= 11.2579, Z_s &= -38.6267, B_s &= -4.2184, Z_s &= 0.0698, \\
\hat{M}_s &= 0.5M_s, \hat{M}_s &= 2M_s, \tau &= 0.05.
\end{align*}
\]

For the calculation of \( \delta \) equation (32) is used, with \( \delta \) of form (41). The previous equations system (49) is equivalent with the following one
\[
\begin{bmatrix}
\dot{\mathbf{V}}_s \\
\dot{\mathbf{B}} \\
\dot{\mathbf{V}}_w
\end{bmatrix} =
\begin{bmatrix}
X_s & X_s & X_s \\
B_s & B_s & 0 \\
Z_s & Z_s & Z_s
\end{bmatrix}
\begin{bmatrix}
\mathbf{V}_s \\
\mathbf{B} \\
\mathbf{V}_w
\end{bmatrix} +
\begin{bmatrix}
X_s & X_s & X_s \\
0 & 0 & -1 \\
0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
\mathbf{v}_s \\
\mathbf{B} \\
\mathbf{v}_w
\end{bmatrix} +
\begin{bmatrix}
\mathbf{z}_s \\
\mathbf{z}_w \\
\mathbf{z}_s
\end{bmatrix} \delta
\]
(50)

With this one (41) becomes
\[
\varepsilon = (\mathbf{M}_s + 2\mathbf{b}_s) \begin{bmatrix} \mathbf{M}_s & \mathbf{M}_s & \mathbf{M}_s \\ \mathbf{M}_s & \mathbf{M}_s & \mathbf{M}_s \end{bmatrix} \begin{bmatrix} \mathbf{V}_s \\ \mathbf{B} \\ \mathbf{V}_w \end{bmatrix} + \begin{bmatrix} \mathbf{M}_s & \mathbf{M}_s & \mathbf{M}_s \end{bmatrix} \begin{bmatrix} \mathbf{v}_s \\ \mathbf{B} \\ \mathbf{v}_w \end{bmatrix} + \begin{bmatrix} \mathbf{z}_s \\ \mathbf{z}_w \\ \mathbf{z}_s \end{bmatrix} \delta.
\]
(51)

The block diagram of the reference model is the one presented in fig.4 and the block diagram of the system for automat command of the pitch angle is presented in fig.3.

Fig.4. The block diagram of the reference model

Actuators’ characteristics (time delays, nonlinearities with saturation zone) lead to adapting difficulties of the neural network. This is why a block “PCH” is introduced; it limits the adaptive pseudo-control \( \mathbf{v}_s \) and \( \mathbf{v} \) 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 model response; this correction depends on actuator’s position [3], [9].

Because the dependence between \( \delta \) and \( \delta_s \) is expressed by a non-linear function \( h_s \), one yields
\[
\hat{h}_s (x, \delta_s) \neq h(x, \delta).
\]
(52)

It results a difference between the two functions
\[
\mathbf{v}_s = h(x, \delta_s) - h(x, \delta).
\]
(53)

Taking into account that
\[
\hat{h} (x, \delta) = h(x, \delta, \mathbf{v}) = \mathbf{v},
\]
function (53) becomes
\[
\mathbf{v}_s = \mathbf{v} - \hat{h} (x, \delta).
\]
(55)

This signal is introduced in the reference model as an additional input [3]; one compares it with \( \mathbf{y}^*(\cdot) \) inside of the reference model and, after integration, it will lead to the modify of the signals \( \mathbf{y} \) and \( \mathbf{y}^* \). The block diagram of the subsystem formed by (55) and actuator is presented in fig.5.

Fig.5. The model of the non-linear actuator

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.5), in which \( x = \theta \); the block of calculus for (32) is replaced with the subsystem from fig.5. One choose \( T = 0.03s \) and the control limits in position and speed of the actuators 5grd, respectively 50grd/s [9].

In fig.6 Matlab/Simulink model for the structure from fig.3 is presented; one has chosen \( \theta_c = 5\text{grd} \).

Fig.6. Matlab/Simulink model for the structure from fig.3

In fig.7 the functions \( \theta(t), \theta(t), \theta(t), \theta(t), \delta(t) \) and \( \nu(t) \) (\( \theta, \varepsilon, \delta \) – with blue color and \( \theta, \varepsilon, \delta \) with red color) are presented. If the actuator is a linear one \( \theta \to \delta, \varepsilon \to \varepsilon \) (the adaptive component of the command compensates \( h_s \) approximation’s error), \( \delta \to \delta \) and \( \nu \to 0 \).

If the actuator is non-linear one obtains the characteristics from fig.8; additionally, characteris-
tics $v_e(t)$ and $\dot{\theta}(t)$ appear. When $v_e = 0$ the actuator is in the saturation state and it works in the linear zone when $v_e \neq 0$. The characteristic $\dot{\theta}(t)$ (phase portrait of the system) shows that the non-linear system tends to a stable limit cycle.

Fig. 7. Time characteristics in the case of linear actuator’s use ($r = 3$)

Fig. 8. Time characteristics in the case of non-linear actuator’s use ($r = 3$)

4 Conclusion

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.

References:


