On – line Parametric Identification and Discrete Optimal Command of the Flying Objects' Move

Romulus Lungu, Mihai Lungu, Liviu Dinca, Eleonor Stoenescu

Abstract— This paper presents a new on-line parametric identification and discrete optimal command algorithm for mono or multivariable linear systems. The method may be applied with good results to the automatic command of the flying objects' move. The simulation results obtained with this real time algorithm, with parametric identification for the longitudinal and lateral move of an aircraft are also presented. This algorithm may be used, with good results, for identification and optimal command of an air-air rocket's move in vertical plain regarding to target's line [1].

Keywords— Algorithm, Control law, Observer, Optimal.

I. INTRODUCTION

BECAUSE of fast flying parameters' modify for the modern aircrafts and rockets, performing real time identification and optimal or adaptive command algorithms have to be made. The authors of this paper have made such an algorithm. First of all, an off-line parametric identification is made, without command, for obtaining the initial values of these parameters for the on-line identification process.

Using the control system (A) and model's outputs, a discrete optimal command law is projected, using a quality quadratic criterion, which assures the convergence of the difference between control system and model's outputs. The model's parameters, obtained by the on-line identification, are used for calculus of the command law. For the algorithm validation one uses as examples the automatic command of an aircraft longitudinal and lateral move or the rocket's move in vertical plain; time characteristics, representing evolution of state variables of A and their estimate, are plotted. These variables' stabilization and the convergence of the errors $e_i = x_i - \hat{x}_i$ happen in maximum 2 seconds.

The proposed algorithm produces very good results in the case of lateral move's stabilization for transport and fights aircrafts or in the case of parametric identification of an air-air rocket's move in vertical plain regarding to target's line.

Paper published in the Proceedings of the 7th WSEAS International Conference on SYSTEM SCIENCE and SIMULATION in ENGINEERING (ICOSSSE '08), Venice, Italy, November 21-23, 2008, pp. 211 – 216. Manuscript received December 30, 2008.

This work was partially supported by the Romanian Ministry of Education and Research under Grant No. 81-005PNCDI/2007.

The authors are with University of Craiova, Faculty of Electrotechnics Blv. Decebal, No.107, Craiova, Romania. The first three authors are with Avionics Department, whereas the fourth author is with Electric Department. E-mail: Lma1312@yahoo.com, romulus_lungu@yahoo.com

II. CONTINUOUS AND DISCRETE MODELS FOR THE LONGITUDINAL AND LATERAL MOVE OF THE AIRCRAFT

The control system (the move of A) may be described by the input – output equations with general forms [2]

$$\dot{x} = Ax + Bu, \tag{1}$$

$$y = cx, \tag{2}$$

where x is the state vector $(n \times 1), u-$ the command vector $(m \times 1), A-$ the system matrix $(n \times n), B-$ matrix $(m \times m)$, $y-$ output vector $(p \times 1), C-$ measurement system matrix $(p \times n), p \le n$. The estimated model is described by equations

$$\dot{\hat{x}} = \hat{A}\hat{x} + \hat{B}u, \tag{3}$$

$$\hat{y} = \hat{C}\hat{x}, \tag{4}$$

where $\hat{x}$ is the estimation of the state vector, $\hat{y}-$ estimation of the output vector $y, \hat{A}, \hat{B}$ and $\hat{C} = C-$ estimate matrices.

The discrete variants of equations systems (1), (2), and (3), (4) are, respectively [3]

$$x(k+1) = A_d x(k) + B_d u(k), \tag{5}$$

$$y(k) = C_d x(k); \tag{6}$$

$$\hat{x}(k+1) = \hat{A}_d \hat{x}(k) + \hat{B}_d u(k), \tag{7}$$

$$\hat{y}(k) = \hat{C}_d \hat{x}(k); \tag{8}$$

matrices A_d, B_d, C_d and $\hat{A}_d, \hat{B}_d, \hat{C}_d$ are discrete variants of matrices A, B, C and $\hat{A}, \hat{B}, \hat{C}$.

Another description form for the estimated system A (A dynamics estimation) [4] is

$$\hat{y}(k+1) = \hat{x}^T(k+1)\hat{b}(k) + \hat{e}(k+1) \tag{9}$$

or

$$\hat{y}(k+1) = \hat{b}^T(k)\hat{z}(k+1) + \hat{e}(k+1), \tag{10}$$

where $e(k+1) = y(k+1) - \hat{y}(k+1)$,

$$\hat{b}^T(k) = \left[\hat{\alpha}^T(k) \quad \hat{b}_1(k) \quad \hat{\beta}^T(k)\right], \tag{11}$$

with

$$\begin{aligned} \hat{\alpha}^T(k) &= \left[-\hat{a}_1(k) \quad -\hat{a}_2(k) \quad \ldots \quad -\hat{a}_n(k)\right], \\ \hat{\beta}^T(k) &= \left[\hat{b}_1(k) \quad \hat{b}_2(k) \quad \ldots \quad \hat{b}_m(k)\right], \end{aligned} \tag{12}$$

$\hat{\alpha}(np \times p), \hat{b}_1(p \times m), \hat{\beta}[(m-1)p \times m]$;

$$z^T(k+1) = \left[\hat{Y}^T(k) \quad u(k) \quad U^T(k)\right], \tag{13}$$

with

$$\begin{aligned} \hat{Y}^T(k) &= \left[\hat{y}(k) \quad \hat{y}(k-1) \quad \ldots \quad \hat{y}(k-n+1)\right], \\ \hat{U}^T(k) &= \left[u(k-1) \quad u(k-2) \quad \ldots \quad u(k-m+1)\right]; \end{aligned} \tag{14}$$

$\hat{Y}(np \times 1), U[(m-1)m \times 1]$.

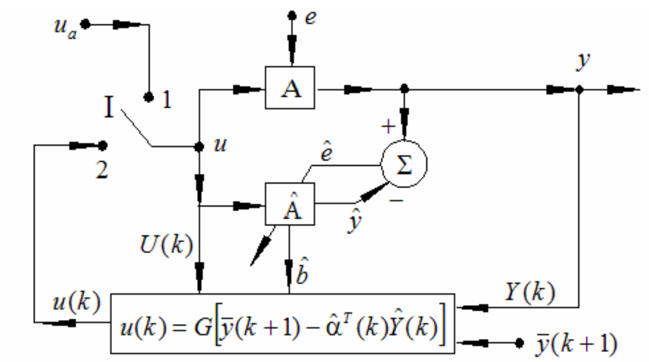

Fig. 1 The structure of the parametric estimation and discrete optimal command of the longitudinal move

If $m = p$, then equation (10) becomes

$$\hat{y}(k+1) = \hat{\alpha}^T(k)\hat{Y}(k) + \hat{b}_1(k)u(k) + \hat{\beta}^T(k)U(k); \tag{15}$$

if $m \neq p$, then $\hat{\beta}^T(k)$ matrix can not be multiplied wit $U(k)$ vector because of their dimensions. That's why, in equation (15) the last term is expressed for each concrete case (function of m and p values). So that, in the case presented below (longitudinal move) $n = 4, m = 1$ and equation (15) becomes

$$\hat{y}(k+1) = \hat{\alpha}^T(k)\hat{Y}(k) + \hat{b}_1(k)u(k), \tag{16}$$

where

$$\begin{aligned} \hat{Y}^T(k) &= \left[\hat{y}(k) \quad \hat{y}(k-1) \quad \hat{y}(k-2) \quad \hat{y}(k-3)\right], \\ \hat{\alpha}^T(k) &= \left[-\hat{a}_1(k) \quad -\hat{a}_2(k) \quad -\hat{a}_3(k) \quad -\hat{a}_4(k)\right], \end{aligned} \tag{17}$$

$\hat{b}_1$ is a $(p \times 1)$ vector, $\hat{y}$ is a $(p \times 1)$ vector and $u(k) - (1 \times 1)$.

In the case of the lateral move of the aircraft $(n = 4,\ m = 2)$, equation (16) becomes

$$\hat{y}(k+1) = \hat{\alpha}^T(k)\hat{Y}(k) + \hat{b}_1(k)u(k) + \hat{b}_1(k)U(k-1), \tag{18}$$

with Y and $\hat{\alpha}$ having forms (17); $\hat{b}_1$ and $\hat{b}_2$ are vectors $(p \times 2), y-$ vector $(p \times 1)$ and $u-$ vector (2×1).

For command law $(u(k))$ obtaining, one chooses the performance indicator

$$\begin{aligned} J = &\left[\bar{y}(k+1) - \hat{y}(k+1)\right]^T Q\left[\bar{y}(k+1) - \hat{y}(k+1)\right] + \\ &+ u^T(k)Ru(k), \end{aligned} \tag{19}$$

where $\bar{y}(k+1)$ is the imposed output vector while $Q(p \times p)$ and $R(m \times m)$ are symmetric and positive definite matrices, $R-$ nonsingular matrix; $\hat{y}(k+1)$ has forms (15) or (18). The optimal command is obtained from optimum condition $(\partial J / \partial u(k) = 0)$,

$$u(k) = G\left[\bar{y}(k+1) - \hat{\alpha}^T(k)\hat{Y}(k)\right], \tag{20}$$

for $m = 1$ (longitudinal move) and

$$u(k) = G\left[\bar{y}(k+1) - \hat{\alpha}^T(k)\hat{Y}(k) - \hat{b}_2(k)u(k-1)\right] \tag{21}$$

for $m = 2$ (lateral move).

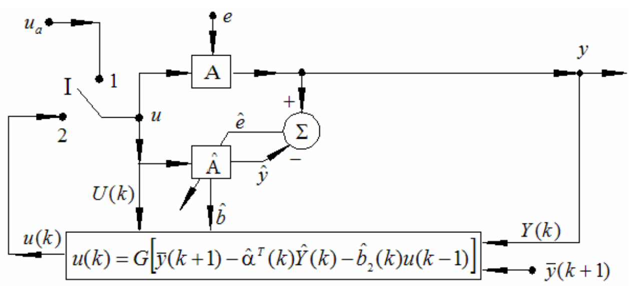

Fig. 2 The structure of the parametric estimation and discrete optimal command of the lateral move

Matrix G from the above equations has the form

$$G = \left[R + \hat{b}_1^T(k)Q\hat{b}_1(k)\right]^{-1}\hat{b}_1^T(k)Q. \tag{22}$$

Q and R matrices may be calculated using ALGLX, algorithm proposed by the authors of this paper or other algorithms [5], [6], [7].

The structure of the parametric estimation and discrete optimal command is presented in fig. 1 (for the longitudinal move of the aircraft) and fig. 2 (for the lateral move of the aircraft).

III. THE ALGORITHM ALGLDR FOR THE IDENTIFICATION OF THE LONGITUDINAL AND LATERAL MOVE'S PARAMETERS

Below one presents the algorithm for the identification of the longitudinal and lateral move's parameters. This algorithm's name is *ALGLDR.*

Algorithm ALGLDR

Step 1: First of all the off – line system A's parameters identification is made, using, for example, the least square method (LSM) [8], resulting the parameters vector $\hat{b}_0 = \hat{b}(0)$; that refers to the coefficients $\hat{b}_j, j = \overline{1,r}, \hat{a}_i = \overline{1,n}$ of the discrete transfer functions of the aircraft's estimated model $\hat{\mathrm{A}}$ (in fig.1 switch I has position 1, $e-$ disturbances and $u = u_a -$ the random input); $\hat{y}(t)$ is then computed and the vectors $\hat{Y}_0 = \hat{Y}(0)$ and $U_0 = U(0)$ are memorized. Also, the covariance matrix P_0 (obtained at the end of identification $P_0 = P(0)$) is also memorized. Then, matrices $A_d, B_d, \hat{A}_d, \hat{B}_d$ are computed and with these state vectors x and $\hat{x}$ are computed using equation (5); these vectors (at the end of identification) are memorized;

Step 2: For simulation of time varying of A's parameters, the parameters of A are modified (for example with 5%) and with the new coefficients A_d and B_d matrices are computed;

Step 3: Switch I has now position 2 (on – line control); using algorithm ALGLX [9], matrices Q' and R matrices are obtained in rapport with $\hat{A}_d$ and $\hat{B}_d$ as follows

Step 3.1: One brings the system described by pair $A(n \times n), B(n \times m)$ to Jordan form $(\overline{A}, \overline{B})$, using transformation $x = T\overline{x}; T$ is a non singular linear transformation [10];

$$\overline{A} = T^{-1}AT, \overline{B} = T^{-1}B = \begin{bmatrix} I_m & \vdots & 0 \end{bmatrix}, \tag{23}$$

where T has the form $T = \begin{bmatrix} B & \vdots & \widetilde{T} \end{bmatrix}$, with $\widetilde{T}$ random matrix $(n \times (n-m))$ so that $\mathrm{rang} T = n$ [6].

Step 3.2: Gain matrix $\overline{K}$ for the optimal control of system $(\overline{A}, \overline{B})$ is obtained so that closed loop system with matrix $\overline{G} = \overline{A} - \overline{B}\overline{K}$ has imposed stable eigenvalues.

Step 3.3: Matrices $\overline{K}$ and $\overline{P}$ are partitioned as follows

$$\overline{K} = \begin{bmatrix} \overline{K}_1 & \vdots & \overline{K}_2 \end{bmatrix}, \overline{P} = \begin{bmatrix} \overline{P}_{11} & \overline{P}_{12} \\ \overline{P}_{21} & \overline{P}_{22} \end{bmatrix}, \overline{P}_{12} = \overline{P}_{21}^T, \overline{P}_{22} = \overline{P}_{22}^T; \tag{24}$$

$\overline{K}_1$ and $\overline{P}_{11}$ are sub matrices $(m \times m)$; sub matrices $\overline{P}_{11}, \overline{P}_{12}, \overline{P}_{22}$ and $\overline{R}$ are calculated in rapport with sub matrices of matrix $\overline{K}$ and with weight matrix $\overline{R} = R$

$$\overline{P}_{11} = R\overline{K}_1, \overline{P}_{12} = \overline{P}_{21}^T = R\overline{K}_2, \overline{P}_{22} = I_{n-m}, \tag{25}$$

where I_{n-m} is the unity matrix $(n-m) \times (n-m)$; for $m = 1$

$$\overline{K} = \begin{bmatrix} k_1 & \vdots & k_{21} & k_{22} & k_{23} \end{bmatrix}, R = [1] \tag{26}$$

and for $m = 2$

$$\overline{P} = \begin{bmatrix} R\overline{K}_1 & R\overline{K}_2 \\ (R\overline{K}_2)^T & I_2 \end{bmatrix}, \tag{27}$$

where $\overline{K}_1$ and $\overline{K}_2$ have forms

$$\overline{K} = \begin{bmatrix} \overline{K}_1 & \vdots & \overline{K}_2 \end{bmatrix}, \overline{K}_1 = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix}, \overline{K}_2 = \begin{bmatrix} k'_{11} & k'_{12} \\ k'_{21} & k'_{22} \end{bmatrix}, \\ R = \begin{bmatrix} 1 & r \\ r & 1 \end{bmatrix}, r = \frac{k_{12} - k_{21}}{k_{11} - k_{22}}. \tag{28}$$

Step 3.4: Matrices $\overline{Q}$ and Q are calculated

$$\overline{Q} = -\left[\overline{PA} + \overline{A}^T\overline{P} - \overline{PBK}\right], \tag{29}$$

$$Q = (T^{-1})^T \overline{Q} T^{-1}; \tag{30}$$

then, knowing matrices A, B, Q and R, one solves EMAR and obtains P

$$PA + A^TP - PBR^{-1}B^TP + Q = 0; \tag{31}$$

one calculates gain matrix with equation

$$K = R^{-1}B^TP. \tag{32}$$

Step 3.5: One calculates the eigenvalues of matrix $G = (A - BK)$; if these are placed in left complex semi plane (matrix G is stable), then gain matrix is the one already obtained; otherwise one returns to *step 3.1* and chooses another matrix $\widetilde{T}$, of course another matrix T and the calculus conform to algorithm's steps is again achieved.

Step 4: G matrix from (22) is obtained with $\hat{b}_1$ extracted from $\hat{b}(k)$;

Step 5: Command $u(k)$ is computed with equation (20) or (21) using $\hat{Y}^T(k)$ and $\hat{\alpha}^T(k)$ with elements (17), $\hat{b}_2(k)$

extracted from $\hat{b}(k)$ and component $u(k-1)$ of $U(k)$;

Step 6: The vectors $x(k+1), \hat{x}(k+1)$ are obtained using (5), respectively (7); $y(k+1)$ and $\hat{y}(k+1)$ are calculated as bellow

$$y(k+1) = C_d(k+1), \hat{y}(k+1) = \hat{C}_d(k+1). \qquad (33)$$

Vectors $\hat{Y}(k+1)$ and $U(k+1)$ are memorized and the error

$$\hat{e}(k+1) = y(k+1) - \hat{y}(k+1) \qquad (34)$$

is computed;

Step 7: The actualization of covariance matrix is made with formula [4]

$$P(k+1) = P(k)\left[I_{m+n} - \frac{\hat{z}(k+1)\hat{z}^T(k+1)}{\lambda + \hat{z}^T(k+1)P(k)\hat{z}(k+1)} P(k)\right] \qquad (35)$$

and, with this,

$$\hat{b}(k+1) = \hat{b}(k) + P(k+1)\hat{z}(k+1)\hat{e}(k+1), \qquad (36)$$

where $\hat{z}(k+1)$ has the form (13);

Step 8: $k \to k+1$ and one returns to *step 4*; if $k < k_{imposed}$, the program stops; state variables $x_i(t)$ and $\hat{x}_i(t)$ are plotted.

IV. IDENTIFICATION AND OPTIMAL COMMAND OF THE AIRCRAFT LONGITUDINAL MOVE

For the identification and discrete optimal command algorithm's validation, presented above, a simulation program was made in the MATLAB medium (the program is presented in Appendix).

The longitudinal move of the aircraft is described by equation [11]

$$\begin{bmatrix} \Delta\dot{V}_x \\ \Delta\dot{\alpha} \\ \Delta\dot{\theta} \\ \Delta\dot{\omega}_y \end{bmatrix} = \begin{bmatrix} -0.007 & 0.012 & -9.81 & 0 \\ -0.128 & -0.54 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0.065 & 0.96 & 0 & -0.99 \end{bmatrix} \begin{bmatrix} \Delta V_x \\ \Delta\alpha \\ \Delta\theta \\ \Delta\omega_y \end{bmatrix} + \begin{bmatrix} 0 \\ -0.04 \\ 0 \\ -12.5 \end{bmatrix} \delta_p .$$

where V_x is the longitudinal component of flight's speed, $\alpha-$ the attack angle, $\theta-$ the pitch angle, ω_y- the pitch angular velocity and δ_p- the elevator deflection

Off–line identification of the longitudinal move is made and the vector associated to estimated model $\hat{\mathrm{A}}$ is obtained

$$\hat{b}_0^T = \hat{b}^T(0) = [4.013 \; -6.042 \; 4.047 \; -1.017 \; 0.058 \; -0.096 \; 0.106 \; -0.070]$$

of the system A, whose vector is b_0,

$$b_0^T = [3.98 \; -5.95 \; 3.95 \; -0.98 \; 2\cdot10^{-5} \; 6.1\cdot10^{-5} \; -6\cdot10^{-5} \; -2\cdot10^{-5}]$$

and matrices

$$\hat{A}_d = \begin{bmatrix} 4.012 & 1 & 0 & 0 \\ -6.042 & 0 & 1 & 0 \\ 4.046 & 0 & 0 & 1 \\ -1.017 & 0 & 0 & 0 \end{bmatrix}, \hat{B}_d = \begin{bmatrix} 0.058 \\ -0.096 \\ 0.106 \\ -0.070 \end{bmatrix}, \hat{C}_d^T = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \hat{D}_d = 0,$$

$$A_d = \begin{bmatrix} 3.984 & 1 & 0 & 0 \\ -5.954 & 0 & 1 & 0 \\ 3.954 & 0 & 0 & 1 \\ -9.984 & 0 & 0 & 0 \end{bmatrix}, B_d = 10^{-4}\begin{bmatrix} 0.203 \\ 0.610 \\ -0.605 \\ -0.201 \end{bmatrix},$$

$$C_d^T = \hat{C}_d, D_d = \hat{D}_d.$$

After switching of I, it begins the on – line identification regime; considering that A's parameters change with 5%, it results the vector

$$b^T = [3.78 \; -5.65 \; 3.75 \; -0.93 \; 1.9\cdot10^{-5} \; 5.8\cdot10^{-5} \; -5.7\cdot10^{-5} \; -1.9\cdot10^{-5}].$$

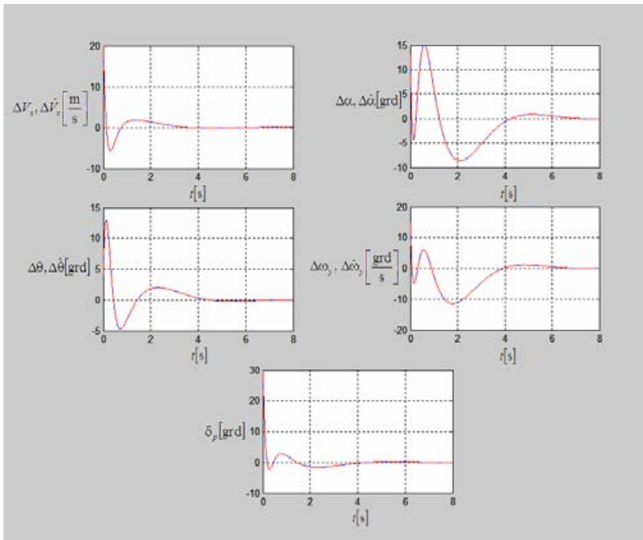

Fig. 3 Dynamics of state variables and of their estimations for the longitudinal move

Using algorithm ALGLX, one calculates matrices Q and R [9]: $Q = 1.548, R = 1$ and, with these, one obtains G and u for each calculus step, vectors $x, \hat{x}, y, \hat{y}, \hat{e}, P, \hat{b}$ and one plots time characteristics $x_1 = \Delta V_x, \hat{x}_1 = \Delta\hat{V}_x$, $x_2 = \Delta\alpha$, $\hat{x}_2 = \Delta\hat{\alpha}$, $\hat{x}_3 = \Delta\hat{\theta}$, $x_4 = \omega_y, \hat{x}_4 = \hat{\omega}_y$ and $u = \delta_p$ (presented in fig. 3; the curves are $x_i -$ blue, $\hat{x}_i -$ red).

V. IDENTIFICATION AND OPTIMAL COMMAND OF THE AIRCRAFT LATERAL MOVE

One considers now the case of lateral move of a Boeing 744 [12], which flies with, $M = 0.8$ and $H = 40 \cdot 10^3$ ft; the lateral move's state equation is

$$\begin{bmatrix} \Delta\dot{\beta} \\ \Delta\dot{\omega}_z \\ \Delta\dot{\omega}_x \\ \Delta\dot{\varphi} \end{bmatrix} = \begin{bmatrix} -0.0558 & -0.9968 & 0.0802 & 0.0415 \\ 0.598 & -0.115 & -0.0318 & 0 \\ 0.305 & 0.388 & -0.465 & 0 \\ 0 & 0.0805 & 1 & 0 \end{bmatrix} \begin{bmatrix} \Delta\beta \\ \Delta\omega_z \\ \Delta\omega_x \\ \Delta\varphi \end{bmatrix} + \begin{bmatrix} 0.0073 & 0 \\ -0.475 & 0.123 \\ 0.153 & 1.063 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_d \\ \delta_e \end{bmatrix}.$$

where β is the sideslip angle, $\omega_x -$ the roll angular velocity, $\omega_z -$ the yaw angular velocity, $\varphi -$ the roll angle, $\delta_d -$ the direction deflection whereas $\delta_e -$ the aileron deflection.

For the lateral move of an aircraft

$$b_0^T = b^T(0) = \begin{bmatrix} 3.99 & -5.98 & 3.98 & -0.99 & 0 & 0 & 0 & 0 \\ 3.99 & -5.98 & 3.98 & -0.99 & 0.01 & -0.03 & 0.03 & -0.01 \end{bmatrix},$$

$$A_d^{11} = A_d^{21} = \begin{bmatrix} 3.99 & 1 & 0 & 0 \\ -5.98 & 0 & 1 & 0 \\ 3.98 & 0 & 0 & 1 \\ -9.99 & 0 & 0 & 0 \end{bmatrix}, B_d^{11} = \begin{bmatrix} 0.001 \\ -0.004 \\ 0.004 \\ -0.001 \end{bmatrix}, B_d^{21} = \begin{bmatrix} 0.010 \\ -0.031 \\ 0.031 \\ -0.010 \end{bmatrix},$$

$$C_d^{11^T} = C_d^{21^T} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, D_d^{11} = D_d^{21} = 0,$$

$$\hat{b}_0^T = \hat{b}^T(0) = \begin{bmatrix} 3.97 & -5.90 & 3.90 & -0.96 & 0.04 & 0.10 & 0.23 & -0.09 \\ 3.97 & -5.90 & 3.90 & -0.96 & 0.01 & 0.13 & 0.26 & -0.10 \end{bmatrix},$$

$$\hat{A}_d^{11} = \hat{A}_d^{21} = \begin{bmatrix} 3.96 & 1 & 0 & 0 \\ -5.90 & 0 & 1 & 0 \\ 3.90 & 0 & 0 & 1 \\ -0.96 & 0 & 0 & 0 \end{bmatrix}, \hat{B}_d^{11} = \begin{bmatrix} 0.004 \\ -0.109 \\ 0.236 \\ -0.092 \end{bmatrix}, \hat{B}_d^{21} = \begin{bmatrix} 0.013 \\ -0.136 \\ 0.263 \\ -0.101 \end{bmatrix},$$

$$\hat{C}_d^{11^T} = \hat{C}_d^{21^T} = C_d^{11^T}, \hat{D}_d^{11} = \hat{D}_d^{21} = 0.$$

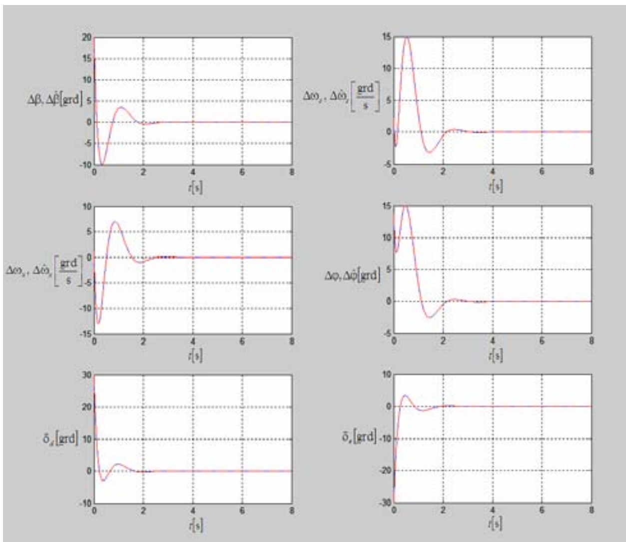

Fig. 4 Dynamics of state variables and of their estimations for the lateral move

Considering that A's parameters change with 5%, it results the vector

$$b^T = \begin{bmatrix} 3.79 & -5.68 & 3.78 & -0.94 & 0 & 0 & 0 & 0 \\ 3.79 & -5.68 & 3.78 & -0.94 & 0.01 & -0.03 & 0.03 & -0.01 \end{bmatrix}.$$

Matrices Q and R are: $Q = 0.0382$, $R = \begin{bmatrix} 1 & 0.985 \\ 0.985 & 1 \end{bmatrix}$.

State variables and their estimates are $x_1 = \Delta\beta, \hat{x}_1 = \Delta\hat{\beta}$, $x_2 = \Delta\omega_z, \hat{x}_2 = \Delta\hat{\omega}_z, x_3 = \Delta\omega_x, \hat{x}_3 = \Delta\hat{\omega}_x, x_4 = \Delta\varphi, \hat{x}_4 = \Delta\hat{\varphi}$ and $u^T = [\delta_d \quad \delta_e]$. Time variations of these variables are presented in fig. 4 (the curves are $x_i -$ blue, $\hat{x}_i -$ red).

VI. IDENTIFICATION AND OPTIMAL COMMAND OF THE ROCKET'S MOVE IN VERTICAL PLAIN

One considers the case of a rocket move in vertical plain with lateral deviation y in rapport with equal line signal line; it is directed by three points method (the co-linearity CP–R–T; CP – control point, R – rocket, T - target). Lateral deviation y in rapport with equal signal line is described by differential equation [13], [14]

$$\dot{y} = V\vartheta + Vf_T, \tag{37}$$

where V is the flight velocity, $\vartheta -$ slope of the trajectory $f_T -$ disturbance. ϑ is defined by [14]

$$T_V\dot{\vartheta} = \alpha - T_V \frac{g}{V}\cos\vartheta. \tag{38}$$

Expressing the above equation in variables α and θ, it becomes

$$T_V\dot{\theta} = T_V\dot{\alpha} + \alpha - T_V\frac{g}{V}\cos\vartheta \tag{39}$$

or has the liniarised form

$$\begin{aligned} T_V\dot{\theta} &= T_V\Delta\dot{\alpha} + \Delta\alpha - \tilde{\alpha}, \\ \tilde{\alpha} &= -\frac{g}{V}T_V\cos\vartheta_0, \dot{\theta} = \Delta\dot{\theta}; \end{aligned} \tag{40}$$

Flying object's move in vertical plain is defined by equation (38) and the following one

$$\dot{V} + \frac{c_x}{c_y^\alpha T_V}V = \left(1 + \frac{c_x}{c_y^\alpha}\right)\frac{F_T}{m} - g\sin\vartheta \tag{41}$$

Rocket's move around mass centre in vertical plain is described by equation [14]

$$J_z\ddot{\theta} + m_{\dot{\theta}}\dot{\theta} + m_\alpha \alpha = m_\delta \delta, \tag{42}$$

where J_z is the inertia moment in rapport with lateral (horizontal) axis, $m_{\dot{\theta}}$ – dynamic damp moment coefficient, m_α – static stabilization moment coefficient, m_δ – command coefficient moment. With $\theta = \vartheta + \alpha$, (38) and (41) the above equation leads to linear form

$$\Delta\ddot{\alpha} + 2\xi\omega_0\Delta\dot{\alpha} + \omega_0^2\Delta\alpha = k_\delta\delta + m_{\dot{\theta}}\frac{g}{J_z V}\cos\theta_0, \Delta\delta = \delta, \tag{43}$$

where ξ is the damp coefficient and ω_0 – proper frequency of the rotation move in vertical plain;

$$\begin{aligned} &2\xi\omega_0 = \frac{m_{\dot{\theta}}}{J_z} + \frac{1}{T_V}, k_\delta = \frac{m_\delta}{J_z}, \\ &\omega_0^2 = \frac{m_\alpha}{J_z} + \frac{m_{\dot{\theta}}}{J_z T_V} + g\frac{\sin\theta_0}{VT_V} - \frac{\dot{T}_V}{T_V^2} \cong \frac{m_\alpha}{J_z} + \frac{m_{\dot{\theta}}}{J_z T_V}. \end{aligned} \tag{44}$$

Equation (42) is equivalent with the following one

$$\ddot{\theta} = -\frac{m_{\dot{\theta}}}{J_z}\dot{\theta} - \frac{m_\alpha}{J_z}\alpha + k_\delta\delta, \tag{45}$$

which, taking into account

$$\frac{m_{\dot{\theta}}}{J_z} = 2\xi\omega_0 - \frac{1}{T_V} = a_1, \frac{m_\alpha}{J_z} \cong \omega_0^2 - \frac{1}{T_V}\frac{m_{\dot{\theta}}}{J_z} = \omega_0^2 - \frac{a_1}{T_V} = a_0, \tag{46}$$

leads to

$$\ddot{\theta} = a_1\dot{\theta} - a_0\Delta\alpha + k_\delta\delta. \tag{47}$$

By derivation of equation (37), one obtains [15]

$$\ddot{y} = \frac{V}{T_V}\Delta\alpha + W_T, W_T = \dot{V}f_T + V\dot{f}_T - g\cos\vartheta_0; \tag{48}$$

W_T is an equivalent disturbance having the significance of normal acceleration to equal signal line [16]. For calculus of above equations' coefficients one uses calculus equations from [17]

$$T_V = \frac{1}{d_1}, \omega_0 = \sqrt{d_1 d_4 - d_3}, \xi = \frac{d_1 + d_4}{2\sqrt{d_1 d_4 - d_3}}, k_\delta = d_2, \tag{49}$$

where d_1, d_2, d_3, d_4 are read using diagrams or graphic characteristics for different rocket types at different flight seconds. For instance, for an Oerlikon rocket, in the 10th flight second

$$\begin{aligned} &d_1 = 1.5\,\mathrm{s}^{-1}, d_2 = 40\,\mathrm{s}^{-2}, d_3 = -10\,\mathrm{s}^{-2}, d_4 = 1.2\,\mathrm{s}^{-1}, \\ &V = 400\,\mathrm{m/s}, T_V = 0.66\,\mathrm{s}, \omega_0 = 4.669\,\mathrm{s}^{-1}, \xi = 0.062, \\ &k_\delta = d_2 = 40\,\mathrm{s}^{-2}, a_1 = 0.92\,\mathrm{s}^{-1}, a_0 = 23.18\,\mathrm{s}^{-2}. \end{aligned} \tag{50}$$

For the 40th flight second

$$\begin{aligned} &d_1 = 2.5\,\mathrm{s}^{-1}, d_2 = 100\,\mathrm{s}^{-2}, d_3 = -80\,\mathrm{s}^{-2}, d_4 = 1\,\mathrm{s}^{-1}, \\ &V = 400\,\mathrm{m/s}, T_V = 0.4\,\mathrm{s}, \omega_0 = 9.083\,\mathrm{s}^{-1}, \xi = 0.0212, \\ &k_\delta = d_2 = 100\,\mathrm{s}^{-2}, a_1 = 2.115\,\mathrm{s}^{-1}, a_0 = 87.7881\,\mathrm{s}^{-2}. \end{aligned} \tag{51}$$

For the 50th flight second

$$\begin{aligned} &d_1 = 1\,\mathrm{s}^{-1}, d_2 = 25\,\mathrm{s}^{-2}, d_3 = -60\,\mathrm{s}^{-2}, d_4 = 0.4\,\mathrm{s}^{-1}, \\ &V = 400\,\mathrm{m/s}, T_V = 1\,\mathrm{s}, \omega_0 = 7.77\,\mathrm{s}^{-1}, \xi = 0.011, \\ &k_\delta = d_2 = 25\,\mathrm{s}^{-2}, a_1 = 0.8197\,\mathrm{s}^{-1}, a_0 = 61.2082\,\mathrm{s}^{-2}. \end{aligned} \tag{52}$$

Choosing the state vector $x^T = [y \quad \dot{y} \quad \Delta\alpha \quad \dot{\theta}]$, system formed by equations (40), (47) and (48) becomes

$$\begin{aligned} &\dot{x}_1 = x_2, \\ &\dot{x}_2 = \frac{V}{T_V}x_3 + W_T, \\ &\dot{x}_3 = -\frac{1}{T_V}x_3 + x_4 + \frac{1}{T_V}\tilde{\alpha}, \\ &\dot{x}_4 = -a_0 x_3 - a_1 x_4 + k_\delta\delta. \end{aligned} \tag{53}$$

If the input is $u = \delta$ and the disturbances vector is $u_p^T = [\tilde{\alpha} \quad W_T]$, the above equations system have the form

$$\dot{x} = Ax + Bu + Eu_p, \tag{54}$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{V}{T_V} & 0 \\ 0 & 0 & -\frac{1}{T_V} & 1 \\ 0 & 0 & -a_0 & -a_1 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ k_\delta \end{bmatrix}, E = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ \frac{1}{T_V} & 0 \\ 0 & 0 \end{bmatrix}. \tag{55}$$

With flying parameters' values from [17], for the 10th second of flight, following step by step the algorithm, one obtained successively the results

$$\begin{aligned} &\hat{b}_0^T = [3.885 \ \ -5.647 \ \ 3.637 \ \ -0.875 \ \ -0.055 \ \ 0.164 \ \ 0.027 \ \ -0.125], \\ &b_0^T = [4.005 \ -6.013 \ \ 4.011 \ -1.003 \ -0.00001 \ -10^{-5} \ \ -10^{-5} \ \ -10^{-5}], \\ &b^T = [3.805 \ -5.713 \ 3.811 \ -0.959 \ -9.506 \cdot 10^{-6} \ \ 0 \ 0 \ -9.524 \cdot 10^{-6}], \end{aligned}$$

$$\hat{A}_d = \begin{bmatrix} 3.901 & 1 & 0 & 0 \\ -5.716 & 0 & 1 & 0 \\ 3.728 & 0 & 0 & 1 \\ -0.913 & 0 & 0 & 0 \end{bmatrix}, \hat{B}_d = \begin{bmatrix} -0.079 \\ -0.070 \\ -0.010 \\ 0.158 \end{bmatrix},$$

$$A_d = \begin{bmatrix} 3.772 & 1 & 0 & 0 \\ -5.620 & 0 & 1 & 0 \\ 3.722 & 0 & 0 & 1 \\ -0.924 & 0 & 0 & 0 \end{bmatrix}, B_d = \begin{bmatrix} 0.009 \\ 0.103 \\ 0.102 \\ 0.009 \end{bmatrix},$$

$$\hat{C}_d = C_d = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}, \hat{D}_d = D_d = 0, Q = 7.565, R = 1.$$

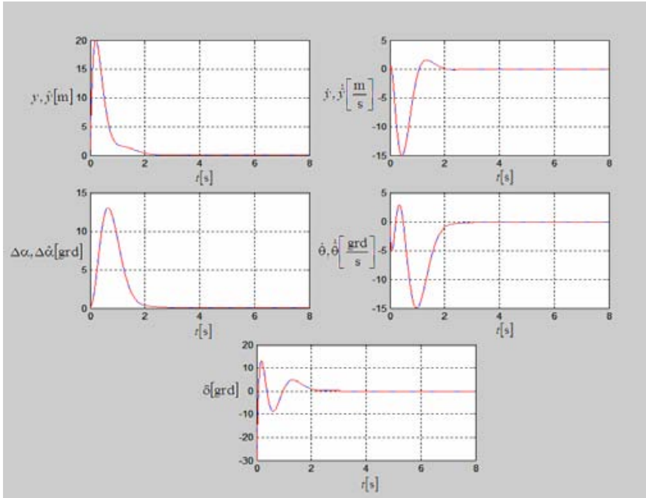

Fig. 5 – Time varying of $x_i, \hat{x}_i$ and δ for the 10th second of flight

In fig. 5 state variables $x_i(t), \hat{x}_i(t)$ and $\delta(t)$ (for the 10th second of flight) are presented ($x_i(t)$ with blue and $\hat{x}_i(t)$ with red).

Similarly, for the 40th and 50th flight second, it results

$$\hat{b}_0^T = [3.845 \ -5.503 \ 3.469 \ -0.811 \ -0.006 \ 0.061 \ 0.136 \ -0.056],$$

$$b_0^T = [4.02 \ -6.06 \ 4.05 \ -1.01 \ -41 \cdot 10^{-6} \ -46 \cdot 10^{-6} \ -46 \cdot 10^{-6} \ -42 \cdot 10^{-6}],$$

$$b^T = [3.822 \ -5.759 \ 3.851 \ -0.964 \ -3.971 \cdot 10^{-5} \ 0 \ 0 \ -4 \cdot 10^{-5}]$$

$$\hat{A}_d = \begin{bmatrix} 3.818 & 1 & 0 & 0 \\ -5.451 & 0 & 1 & 0 \\ 3.445 & 0 & 0 & 1 \\ -0.812 & 0 & 0 & 0 \end{bmatrix}, \hat{B}_d = \begin{bmatrix} -0.001 \\ 0.066 \\ -0.050 \\ 0.076 \end{bmatrix},$$

$$A_d = \begin{bmatrix} 3.759 & 1 & 0 & 0 \\ -5.586 & 0 & 1 & 0 \\ 3.694 & 0 & 0 & 1 \\ -0.917 & 0 & 0 & 0 \end{bmatrix}, B_d = \begin{bmatrix} 0.039 \\ 0.429 \\ 0.426 \\ 0.038 \end{bmatrix},$$

$$\hat{C}_d = C_d = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}, \hat{D}_d = D_d = 0, Q = 0.1067, R = 1,$$

respectively

$$\hat{b}_0^T = [4.016 \ -6.043 \ 4.038 \ -1.011 \ 0.079 \ -0.069 \ 0.049 \ -0.071],$$

$$b_0^T = [4.01 \ -6.03 \ 4.02 \ -1.006 \ -41 \cdot 10^{-7} \ -45 \cdot 10^{-7} \ -46 \cdot 10^{-7} \ -41 \cdot 10^{-7}],$$

$$b^T = [3.81 \ -5.72 \ 3.82 \ -0.95 \ -3.96 \cdot 10^{-6} \ -4.36 \cdot 10^{-6} \ -4.37 \cdot 10^{-5} \ -3.97 \cdot 10^{-6}],$$

$$\hat{A}_d = \begin{bmatrix} 3.949 & 1 & 0 & 0 \\ -5.860 & 0 & 1 & 0 \\ 3.870 & 0 & 0 & 1 \\ -0.960 & 0 & 0 & 0 \end{bmatrix}, \hat{B}_d = \begin{bmatrix} -0.048 \\ -0.038 \\ 0.009 \\ -0.063 \end{bmatrix},$$

$$A_d = \begin{bmatrix} 3.781 & 1 & 0 & 0 \\ -5.649 & 0 & 1 & 0 \\ 3.754 & 0 & 0 & 1 \\ -0.936 & 0 & 0 & 0 \end{bmatrix}, B_d = \begin{bmatrix} 0.039 \\ 0.432 \\ 0.431 \\ 0.039 \end{bmatrix},$$

$$\hat{C}_d = C_d = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}, \hat{D}_d = D_d = 0, Q = 0.1067, R = 1.$$

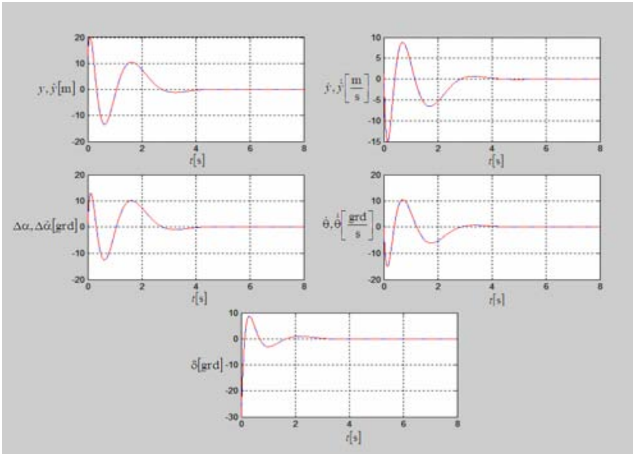

Fig. 6 – Time varying of $x_i, \hat{x}_i$ and δ for the 40th second of flight

In fig. 6 and fig. 7 state variables $x_i(t), \hat{x}_i(t)$ and $\delta(t)$ (for the 40th and 50th second of flight) are presented ($x_i(t)$ with blue and $\hat{x}_i(t)$ with red).

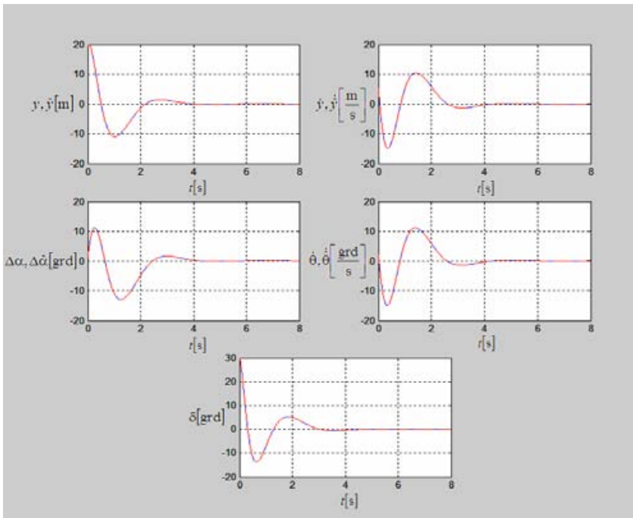

Fig. 7 – Time varying of $x_i, \hat{x}_i$ and δ for the 50th second of flight

VII. CONCLUSIONS

The paper presents an on - line parametric identification and discrete optimal command algorithm for linear systems. For validation, it is used to the command of an aircraft's longitudinal and lateral move and for the case of a rocket move in vertical plain with lateral deviation y in rapport with equal line signal line.

Simulation programs, based on presented algorithm, were made in Matlab. The obtained graphics are time variation of the state variables associated to the control system and to the estimated model of the aircraft $x_i(t), \hat{x}_i(t)$ and the evolution of command $\delta(t)$.

APPENDIX

```
clear all; close all;
% Data presentation (step 0)
A=[-0.0558 -0.9968 0.0802 0.04415;
    0.598 -0.115 -0.0318 0;
    0.305 0.388 -0.465 0;
    0 0.0805 1 0];
B=[0.0073 0;-0.475 0.123; 0.153 1.063;0 0];
C=[0 0 1 0];
Ts=0.01;
x=[0.08;0.02;0.03;0.3]*180/pi;
xc=zeros(4,1);
n=size(A,1);
m=size(B,2);
s=size(C,1);
D=zeros(s,m);
% Off-line identification
%Step 1:Off-line identification(Switch on position 1)
[num1,den1]=ss2tf(A,B,C,D,1);
[num2,den2]=ss2tf(A,B,C,D,2);
sysc1=tf(num1,den1);sysc2=tf(num2,den2);
sysd1=c2d(sysc1,Ts);sysd2=c2d(sysc2,Ts);
[numd1,dend1]=tfdata(sysd1,'v');
[numd2,dend2]=tfdata(sysd2,'v');
AA1=[dend1(1) dend1(2) dend1(3) dend1(4) dend1(5)];
BB1=[numd1(1) numd1(2) numd1(3) numd1(4) numd1(5)];
AA2=[dend2(1) dend2(2) dend2(3) dend2(4) dend2(5)];
BB2=[numd2(1) numd2(2) numd2(3) numd2(4) numd2(5)];
tho1=poly2th(AA1,BB1);
tho2=poly2th(AA2,BB2);
u=idinput(100,'rbs');e=randn(100,1);
% run algldr_ee;
y1=idsim([u,e],tho1);z1=[y1,u];
y2=idsim([u,e],tho2);z2=[y2,u];
th1=arx(z1,[4 4 1]);
yc1=idsim([u,e],th1);
th2=arx(z2,[4 4 1]);
yc2=idsim([u,e],th2);
subplot(211);t=1:length(y1);
plot(t,y1,'b',t,yc1,'r');grid;
subplot(212);t=1:length(y2);
plot(t,y2,'b',t,yc2,'r');grid;
[Adc1,Bdc1,Cdc1,Ddc1]=th2ss(th1);
[Adc2,Bdc2,Cdc2,Ddc2]=th2ss(th2);
[numdc1,dendc1]=th2tf(th1);
[numdc2,dendc2]=th2tf(th2);
[Ad1,Bd1,Cd1,Dd1]=th2ss(tho1);
[Ad2,Bd2,Cd2,Dd2]=th2ss(tho2);
Y=[yc1(length(yc1));yc1(length(yc1)-1);
yc1(length(yc1)-2);yc1(length(yc1)-3)];
alf1=-(dendc1(2:length(dendc2)))';
alf2=-(dendc2(2:length(dendc2)))';alf=alf1;
beta1=(numdc1(2:length(numdc1)))';
beta2=(numdc2(2:length(numdc2)))';
beta=[beta1 beta2];
b1=[beta1(1) beta2(1)];
b2=[beta1(2) beta2(2)];
bc=[alf' b1 b2]';
% Calculus of x(k),y(k),xc(k),yc(k)
ub=[u(length(u)-1);u(length(u)-1)];
u=[u(length(u));u(length(u))];
Ad=Ad1;Bd=[Bd1 Bd2];
Cd=Cd1;Dd=[Dd1 Dd2];
Adc=Adc1;Bdc=[Bdc1 Bdc2];
Cdc=Cdc1;Ddc=[Ddc1 Ddc2];
x=Ad*x+Bd*u;xc=Adc*xc+Bdc*u;
x1(1)=x(1);x2(1)=x(2);
x3(1)=x(3);x4(1)=x(4);
xc1(1)=xc(1);xc2(1)=xc(2);
xc3(1)=xc(3);xc4(1)=xc(4);
% On-line identification (Switch on position 2)
% Step 2:Time variaion of the system parameters
numd1=numd1*(95/100);
numd2=numd2*(95/100);
for i=2:length(dend1)
dend1(i)=(dend1(i))*(95/100);
dend2(i)=(dend2(i))*(95/100);
end
% Calculus of matrices Ad,Bd
AA1=[dend1(1) dend1(2) dend1(3) dend1(4) dend1(5)];
AA2=[dend2(1) dend2(2) dend2(3) dend2(4) dend2(5)];
BB1=[numd1(1) numd1(2) numd1(3) numd1(4) numd1(5)];
BB2=[numd2(1) numd2(2) numd2(3) numd2(4) numd2(5)];
tho1=poly2th(AA1,BB1);
[Ad1,Bd1,Cd1,Dd1]=th2ss(tho1);
tho2=poly2th(AA2,BB2);
[Ad2,Bd2,Cd2,Dd2]=th2ss(tho2);
Ad=Ad1;Bd=[Bd1 Bd2];
Cd=Cd1;Dd=[Dd1 Dd2];
% Step 3:Usage of algorithm ALGLX
close all;
Q=[100 0 0 0;0 1 0 0;0 0 100 0;0 0 0 10];
R=[1 -0.5;-0.5 1];
[K,P,E] = LQR(Adc,Bdc,Q,R);
% Matrix T presentation
N3=randn(4,2);
```

```
for i=1:4
   for j=1:2
      T(i,j)=Bdc(i,j);
      T(i,j+2)=N3(i,j);
   end
end
% Obtaining of matrices Ab,Bb,Kb,Rb
Ab=(inv(T))*Adc*T;
Bb=(inv(T)*Bdc);
Kb=place(Ab,Bb,E);
for i=1:2
   for j=1:2
      K10(i,j)=Kb(i,j);
      K20(i,j)=Kb(i,j+2);
   end
end
k11=K10(1,1);k12=K10(1,2);
k21=K10(2,1);k22=K10(2,2);
Rb=[1 (k12-k21)/(k11-k22);(k12-k21)/(k11-k22) 1];
ee=eig(Rb);N1=Rb*K20;N2=transpose(N1);
Pb=[Rb*K10 Rb*K20;N2 eye(2)]; R=Rb;
Qb=-(Pb*Ab+(transpose(Ab))*Pb-Pb*Bb*Kb);
% Variant 2
PPP=transpose(inv(T))*Pb*inv(T);
KKK=inv(R)*transpose(Bdc)*PPP;
EEE=eig(Adc-Bdc*KKK);m=rank(T);
while real(ee(1))<0 | real(ee(2))<0 | real(EEE(1))>0 |
real(EEE(2))>0 | real(EEE(3))>0 | real(EEE(4))>0 | m<4
% Matrix T presentation
N3=randn(4,2);
for i=1:4
   for j=1:2
      T(i,j)=Bdc(i,j);
      T(i,j+2)=N3(i,j);
   end
end
% Obtaining of matrices Ab,Bb,Kb,Rb
Ab=(inv(T))*Adc*T;
Bb=(inv(T)*Bdc);
Kb=place(Ab,Bb,E);
for i=1:2
   for j=1:2
      K10(i,j)=Kb(i,j);
      K20(i,j)=Kb(i,j+2);
   end
end
k11=K10(1,1);k12=K10(1,2);
k21=K10(2,1);k22=K10(2,2);
Rb=[1 (k12-k21)/(k11-k22);(k12-k21)/(k11-k22) 1];
ee=eig(Rb);
N1=Rb*K20;N2=transpose(N1);
Pb=[Rb*K10 Rb*K20;N2 eye(2)]; R=Rb;
Qb=-(Pb*Ab+(transpose(Ab))*Pb-Pb*Bb*Kb);
% Variant 2
PPP=transpose(inv(T))*Pb*inv(T);
KKK=inv(R)*transpose(Bdc)*PPP;
EEE=eig(Adc-Bdc*KKK);m=rank(T);
end
% Variant 1
Q=transpose(inv(T))*Qb*inv(T);
R=Rb;[K,P,E] = LQR(Adc,Bdc,Q,R);
% Determination of matrices Q si R
Q=(pinv(C'))*Q*(pinv(C));Q=Q(1,1);
% Optimal command determination
for k=1:100
% Step 4:Calculus of matrix G
Par=R+(b1')*Q*b1;
G=(inv(Par))*(b1')*Q;
yb=0;u=G*(yb-(alf')*Y-b2*ub); uu(:,k)=u;
% Step 6: x(k+1),y(k+1),xc(k+1),yc(k+1) calculus
x=Ad*x+Bd*u;y=Cd*x;
xc=Adc*xc+Bdc*u;yc=Cdc*xc;
x1(k+1)=x(1);x2(k+1)=x(2);
x3(k+1)=x(3);x4(k+1)=x(4);
xc1(k+1)=xc(1);xc2(k+1)=xc(2);
xc3(k+1)=xc(3);xc4(k+1)=xc(4);
z=[Y' u' ub']';
Y=[yc;Y(1:length(Y)-1)];e=y-yc;
% Step7
lambda=0.95;P=100*eye(n+m); bc=bc+P*z*e;alf=bc(1:4);
end
subplot(321);
plot(t,x1,'b',t,xc1,'r');grid;
subplot(322);
plot(t,x2,'b',t,xc2,'r');grid;
subplot(323);
plot(t,x3,'b',t,xc3,'r');grid;
subplot(324);
plot(t,x4,'b',t,xc4,'r');grid;
subplot(325);
plot(t1,uu(1,:),'r');grid;
subplot(326);
plot(t1,uu(2,:),'r');grid;
```

REFERENCES

[1] R. Lungu, M. Lungu *On – line parametric identification and discrete optimal command of flyght objects.* The 6th International Conference on Electromechanic and Energetic Systems, SIELMEN, 2 – 5 September 2007, Chisinau, Moldova Republic, 3 pp.

[2] N. Prljaca, Z. Gajic "Optimal Control & Filtering of Weakly Coupled Linear Discrete Stochastic Systems by The Eigenvector Approach", *WSEAS Transactions on systems and control, Volume 2, September 2007*, pp. 435-441.

[3] R. Sadeghian, P. Karimaghaee, A. Khayatian „Controller Reduction of Discrete Linear Closed Loop Systems in a Certain Frequency Domain". *International Journal of Circuits, Systems and Signal Processing, Issue 2, Vol. 1,* 2007.

[4] P. Eykhoff *Identificarea sistemelor*. Technical Publisher, Bucharest, 1977.

[5] L. Ljung, J. Sjoberg, H. Hjalmarsson *Identification, Adaptation, Learning – The Science of Learning Models from Data*, Edited by S. Bittoni, Springer Verloy Berlin Heideberg, 1996.

[6] J. W. Choi „A Simultaneous Assignment Methodology of Right/Left Eigenstructures". *IEEE Transactions on Aerospace and Electronic Systems,* 1998, pp. 625 – 634.

[7] J. W. Choi " LQR Design with Eigenstructure Assignment Capability". *IEEE Transactions on Aerospace and Electronic Systems, vol. 35, Nr. 2*, *April, 1999,* pp. 700 – 707.

[8] M. Arezki, A. Benallal, P. Meyueis, A. Guessonum, D. Berkani "Analysis of fast recursive least squares algorithms for adaptive filtering". *International Journal of Circuits, Systems and Signal Processing, Issue 1, Vol. 1, 2007.*

[9] M. Lungu, R. Lungu, N. Jula, C. Cepisca, M. Calbureanu, "Aspects regarding a new method for the optimal control law's synthesis of aircrafts' move". *WSEAS Transactions on Circuits and Systems, Issue 6, Volume 7, June 2008*, pp. 502 – 511.

[10] T. Tain-Sou "Automatic Gain Control for Unity Feedback Control Systems with Large Parameters Variations". *WSEAS Transactions on Systems and Control, Volume 2,* December 2007, pp. 537–545.

[11] Mc.L. Donald *Automatic Flight Control Systems.* New York, London, Toronto, Syd-ney, Tokyo, Singapore.

[12] K. Koray, S. Durmaz, E. Jafarov *Longitudinal Dynamics Analysis of Boeing 747-400*. Proceedings of the 9th WSEAS International Conference on Automatic Control, Modeling & Simulation, Istanbul, Turkey, May 27-29, 2007, pp. 82–87.

[13] R. Lungu, M. Lungu *Optimal Control of the Rocket's Lateral Deviation in Rapport with Equal Signal Line*. A 6–a Conferinta Internaţionala de Sisteme Electromecanice si Energetice, SIELMEN, 2–5 September 2007, Chisinau, Moldova Republic.

[14] R. Lungu *Sisteme de dirijare aerospaţială.* Sitech Publisher, Craiova, 2002.

[15] M. Lungu *Calculul vitezelor de apropiere şi deviaţie de la linia ţintei.* The International Symposium on Naval and Marine Education. NAV-MAR-EDU 2001, Constanţa, May 24–26, 2001, pp. 113–119.

[16] R. Lungu, L. Dincă, M. Lungu. *Dynamical Errors Calculus of Some Rockets Guidance Systems Using the Direct Method in Three Points.* The 3rd International Conference on Advanced Engineer Design, Prague, June, 2003, CD–6 pp.

[17] I. Aron, R. Lungu *Automate de stabilizare şi dirijare.* Military Publisher. Bucharest, 1991.