THE PUBLISHING HOUSE
OF THE ROMANIAN ACADEMY

PROCEEDINGS OF THE ROMANIAN ACADEMY, Series A,
Volume 15, Number 2/2014, pp. 191–199

RECONFIGURABLE CONTROLLER FOR ACTIVE FAULT-TOLERANT CONTROL SYSTEMS WITH APPLICABILITY TO FLIGHT CONTROL

Mihai LUNGU, Romulus LUNGU

University of Craiova, Faculty of Electrical Engineering, Decebal Blv., No.107, Craiova, Romania
Corresponding author: Romulus LUNGU, E-mail: romulus_lungu@yahoo.com

In this paper we design a new reconfigurable controller for active fault-tolerant control systems with direct applicability to aircraft's flight control. The reconfigurable controller is based on a new algorithm for the calculation of the controller gain matrix without using information from a full-order or reduced order observer. The main advantages of the reconfigurable controller, when no failures appear in the system, are related to a reduction in the number of embedded sensors, a decrease in costs, and the cancellation of the adverse effect on sensor errors. When the system is affected by sensor or actuator failures, the new reconfigurable controller may be used as a subsystem of a fault-tolerant control system; the information from the broken sensors or actuators are not used and the system stabilization is carried out by using information only from the functional sensors and actuators.

Key words: fault-tolerant control system, controller reconfiguration, gain matrix, aircraft motion

1. INTRODUCTION

A. Antecedents and motivations

The technological systems became more and more complex and, that is why, the dependence on their control systems has also increased significantly. All the control systems, no matter how ingenious the design is, and how immaculate the manufacture process is carried out, are subject to failures; in these undesirable circumstances, the most efficient way to ensure reliable operation of a system for intended purposes is to rely on fault-tolerant control strategies [1]. A fault-tolerant control system (FTCS) may not offer optimal performances in a strict sense for normal system operation and the philosophy of a FTCS design is different from other design approaches, the behavior of such a system being also different. Although the design of control systems to achieve fault-tolerance for closed-loop control of safety-critical systems has attracted the attention of many scientists, the field of FTCS is still wide open. When a failure occurs is a system (sensor failure or actuator failure) the entire system characteristics change because the actuators may not be able to provide the same level of driving power, while the sensors may not supply dependable measurements [1].

In [2] a bibliographical review on reconfigurable fault-tolerant control systems is presented; the existing approaches to fault detection and diagnosis (FDD) and fault-tolerant control (FTC) are considered and classified according to different criteria such as design methodologies and applications. The main motivation of the significant amount of research on FTCS was the aircraft control system design [3] and aircraft crashes.

There are two types of FTCS: passive (PFTCS) and active (AFTCS). In the passive FTCS, the controllers are fixed and designed to be robust against a class of presumed faults [4]; this kind of system needs neither FDD schemes nor controller reconfiguration, but it has limited fault-tolerant capabilities [2]. In contrast with PFTCS, the active fault-tolerant control systems react to the system component failures actively by reconfiguring control actions such that the stability and acceptable performance of the entire system can be maintained [2]. In such control systems, the controller compensates for the impacts of the faults either by selecting a pre-computed control law [5, 6] or by synthesizing a new one on-line [7], [8]; both approaches are based on real-time FDD schemes which must provide the information regarding the true status of the system. Accordingly, a FTCS must incorporate a controller which achieves stability and satisfactory performances for the system not only under normal operations, but also under fault conditions. When all control components are functioning normally, more emphasis should be placed on the quality of the system behavior; in the presence of a fault the system must survive with an acceptable performance. Typically, an active FTCS can be divided into three subsystems: 1) a fault detection/diagnosis scheme; 2) controller reconfiguration mechanism, and 3) a reconfigurable controller [9].

From the three subsystems point of view, the difference between active FTCS and passive FTCS is related to the inclusion of both FDD and reconfigurable controllers within the overall system structure. The controller must be easily reconfigured, the FDD scheme must have high sensitivity to faults and robustness to model uncertainties or disturbances, while the reconfiguration mechanism must lead, as much as possible, to the recovery of the pre-fault system performances. To ensure the fulfillment of these conditions, the redundancy in hardware, software and communication networks must be taken into consideration [2].

B. Main contribution

In the specialty literature there are some algorithms for the calculation of a system gain matrix (K). One can mention here the eigenstructure assignment, the Ackermann formula [10], the technique for the calculation of the gain matrices of a stabilizing static output feedback based on the so called "Projection Lemma" [11] or the ALGLX algorithm [12]. The last one has an advantage with respect to the others: two matrices (Q, R), which interfere in a Riccati equation, may be unknown matrices; first, the two matrices are determined and, after that, a Riccati matriceal algebraic equation is solved and the gain matrix is calculated.

In this paper we propose a new reconfigurable controller for active fault-tolerant control systems with applicability to flight control. The reconfigurable controller is based on a new algorithm, with two variants (aircraft longitudinal and lateral motions), for the calculation of the controller gain matrix without using information from a full-order observer. Thus, instead of all states' measuring, we use this new algorithm (ALG_00K algorithm) for the synthesis of a new on-line control law which stabilizes the system when failures of sensors or actuators appear. Some of the advantages are related to a reduction in the number of embedded sensors, a decrease in costs, and the cancellation of the adverse effect on sensor errors; the most important utility of our new approach is the possibility of use the new obtained control law in a fault-tolerant control system. The first step to be made will be the change of the state equations' order such that the first state equations are associated with the variables measured by means of the broken sensors, while the last ones correspond to the variables measured by means of the functional sensors. For determining which are the broken sensors or the failure actuators there are a lot of fault detection/diagnosis schemes and we will not focus on these FDI subsystems here, our aim being to obtain an approach which does not take into consideration the information from the broken sensors and failure actuators. The algorithm has been designed for the case $n = 4$ (four states). Without other difficulties, the algorithm may be extended for the cases $n > 4$.

2. DESIGN OF THE ALGORITHM FOR THE AIRCRAFT'S LONGITUDINAL MOTION

The linearised longitudinal motion of the aircrafts is generally described by the state equations:

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

where the vector of inputs u has only one component (the elevator deflection δ_e). The matrices A and B are $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, where n represents the number of the states and m - the number of inputs; the pair (A, B) is assumed to be controllable. In the case of the longitudinal motion, one may consider $n = 4$, $m = 1$. Using the algorithms from the specialty literature, the gain matrix $K = [k_1 \; k_2 \; k_3 \; k_4]$ is calculated in order to obtain an optimal control law $u = u_1 = -Kx$, which stabilizes the system (1). The closed loop system is described by the equation $\dot{x} = (A - BK)x$. By using the previous two equations we write:

$$\dot{u}_1 = -K(A - BK)x. \tag{2}$$

We assume that two from the four sensors of the system are broken; in this case, the algorithm designs a gain matrix F, having the form $F = [0 \; 0 \; \tilde{k}_3 \; \tilde{k}_4]$, such that the resulting control law $u_2 = -Fx$ has the same stabilization effect on the aircraft motion; obviously, the number of broken sensors equals the number of zeros in the matrix F. The control law u_1 corresponds to the aircraft without broken sensors, while u_2 corresponds to the aircraft with two broken sensors. Thus, from the four states, only the last two must be measured (the first two have not to be measured or estimated by means of an observer). For the new system it results:

$$\dot{u}_2 = -F\left(A - BF\right)x. \tag{3}$$

Because the two control laws must have the same stabilization effect on the aircraft motion, we impose $\dot{u}_1 = \dot{u}_2$ and, for the same initial conditions, it yields:

$$K(A - BK) = F(A - BF). \tag{4}$$

The design of our new reconfigurable controller is equivalent to the determination of the matrix F. For doing this, first we partitionate the matrices A, B, K, and F as follows:

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, K = \begin{bmatrix} K_1 & K_2 \end{bmatrix},\ F = \begin{bmatrix} 0_{m\times(n-m)} & \widetilde{K} \end{bmatrix}, \tag{5}$$

where $A_{11} \in \mathbb{R}^{(n-m)\times(n-m)}$, $A_{12}, B_1 \in \mathbb{R}^{(n-m)\times m}$, $A_{21}, K_1 \in \mathbb{R}^{m\times(n-m)}$, $A_{22}, B_2, K_2, \widetilde{K} \in \mathbb{R}^{m\times m}$. Substituting (5) into (4) it results the following equation system:

$$\begin{cases} \widetilde{K}A_{21} = K_1\left(A_{11} - B_1K_1\right) + K_2\left(A_{21} - B_2K_1\right), \\ \widetilde{K}\left(A_{22} - B_2\widetilde{K}\right) = K_1\left(A_{12} - B_1K_2\right) + K_2\left(A_{22} - B_2K_2\right); \end{cases} \tag{6}$$

the system has two equations and only one unknown term (matrix $\widetilde{K}$). $\widetilde{K}$ is determined from the first equation of the system (6), but the obtained solution may not verify the second equation of the system. That is why we solve the first equation (6) and the obtaining matrix $\widetilde{K}$ will be considered as the starting solution for the solving of the second equation (6). Denoting the elements of matrix A-BK with $\alpha_{ij} = a_{ij} - b_i k_j$, $i, j = \overline{1, n}$, where a_{ij}, b_i, k_j are the elements of matrices A, B, and K, the first equation (6) gets the form:

$$\begin{cases} a_{31}\widetilde{k}_3 + a_{41}\widetilde{k}_4 = \beta_1, \\ a_{32}\widetilde{k}_3 + a_{42}\widetilde{k}_4 = \beta_2, \end{cases} \tag{7}$$

with

$$\beta_1 = \alpha_{11}k_1 + \alpha_{21}k_2 + \alpha_{31}k_3 + \alpha_{41}k_4\,, \beta_2 = \alpha_{12}k_1 + \alpha_{22}k_2 + \alpha_{32}k_3 + \alpha_{42}k_4\,. \tag{8}$$

The system (7) has two equations and two unknown variables $\widetilde{k}_3$ and $\widetilde{k}_4$. The elements of the matrix K (i.e. k_1, k_2, k_3, k_4) are considered to be known because the first step of the ALG_00K algorithm is the calculation of the gain matrix K by means of the ALGLX algorithm [12] or other algorithms. The system (7) may be written under the form of a matriceal equation $A_{21}\widetilde{K} = \beta$ or:

$$\begin{bmatrix} a_{31} & a_{41} \\ a_{32} & a_{42} \end{bmatrix} \begin{bmatrix} \widetilde{k}_3 & \widetilde{k}_4 \end{bmatrix} = \begin{bmatrix} \beta_1 & \beta_2 \end{bmatrix} \tag{9}$$

Remark 1. The only assumption of this algorithm is that A_{21} is not singular. If the assumption holds, the solving of the system (7) is made by using the least square method [13], or, more direct, as $\widetilde{K} = A_{21}^{-1}\beta$; otherwise, we use the pseudo-inverse of the matrix $A_{21}\left(A_{21}^{+}\right)$; thus, one obtains $\widetilde{K} = A_{21}^{+}\beta$.

The solution of the first equation (6) represents the starting solution $\left(\widetilde{K}_0\right)$ in the determination of the matrix $\widetilde{K}$ from the second equation (6). The second equation (6) is nonlinear and it requires a starting (initial) solution. It leads to the following nonlinear equations system:

$$\begin{cases} -b_3\widetilde{k}_3^2 + a_{33}\widetilde{k}_3 - b_4\widetilde{k}_3\widetilde{k}_4 + a_{43}\widetilde{k}_4 = \beta_3\,, \\ -b_4\widetilde{k}_4^2 + a_{44}\widetilde{k}_4 - b_3\widetilde{k}_3\widetilde{k}_4 + a_{34}\widetilde{k}_3 = \beta_4\,, \end{cases} \tag{10}$$

where $\beta_3 = \alpha_{13}k_1 + \alpha_{23}k_2 + \alpha_{33}k_3 + \alpha_{43}k_4$, $\beta_4 = \alpha_{14}k_1 + \alpha_{24}k_2 + \alpha_{34}k_3 + \alpha_{44}k_4$. The nonlinear system (10) is solved by using the method of the secant [14] starting from an initial solution $\tilde{K}_0 = \left[\tilde{k}_{3_0}\ \tilde{k}_{4_0}\right]$. In Matlab environment, for the solving of the nonlinear systems with the method of the secant the command *fsolve* is used. This way, we obtain $\tilde{k}_3$, $\tilde{k}_4$, and the matrix $F = \left[0\ 0\ \tilde{k}_3\ \tilde{k}_4\right]$; the control law u_2, depending on the matrix F, must stabilize all the n states although the system has feedbacks only after the last two states. The other states (x_1, x_2) will not be measured (the sensors for x_1 and x_2 have been considered broken) or estimated (the FTCS has not an observer). After the determination of the matrix F we study the stability of the closed loop system. If $\sigma(A - BF) \notin C_-$, the above calculations are repeated until the $\sigma(A - BF) \in C_-$. We determine again the gain matrix K because the ALGLX algorithm [12] calculates this matrix by means of coordinates change and a randomly chosen matrix T. Because the matrix K is determined in several cycles and because the matrix T is random, the gain matrix K is different each time when we run the ALGLX algorithm. In the case of an actuator failure, our approach can be applied after an appropriate modification of the matrix B; thus, the column which corresponds to the failure actuator is replaced by a column of zeros and the matrix F is determined in the same way.

The iterative ALG_00K algorithm, for the stabilization of the aircraft's longitudinal motion is summarized as follows: **Step 1:** We change the order of the state equations which describe the aircraft's motion such that the first equations are associated with the variables measured by means of the broken sensors, while the last ones correspond to the variables measured by means of the functional sensors. We check if the matrix A_{21} is nonsingular. If the condition is fulfilled A_{21}^{-1} is calculated; otherwise the pseudo-inverse of the matrix A_{21} $\left(A_{21}^{+}\right)$ is obtained; **Step 2:** We calculate the matrix K with the ALGLX algorithm [12]; **Step 3:** We partitionate the matrices A, B, K, and F according to equation (5); **Step 4:** We calculate $\alpha_{ij}, \beta_i, i = \overline{1, n}$ and, after that, we solve the equation system (7) by means of the least square method and we obtain the starting (initial) solution $\tilde{K}_0 = \left[\tilde{k}_{3_0}\ \tilde{k}_{4_0}\right]$. Instead of the least square method, the inverse or pseudo-inverse of the matrix A_{21} can be used to solve (7); **Step 5:** Nonlinear system (10) is solved with the method of the secant [14] and we obtain $\tilde{k}_3$ and $\tilde{k}_4$; **Step 6:** We get the matrix $F = \left[0\ 0\ \tilde{k}_3\ \tilde{k}_4\right]$ and the system closed loop eigenvalues. If $\sigma(A - BF) \in C_-$ the gain matrix F has been correctly determined; otherwise, we return to step 2 and the steps 2-6 are again run until the system is stable.

3. DESIGN OF THE ALGORITHM FOR THE AIRCRAFT'S LATERAL MOTION

The linearised lateral motion of the aircrafts is described by equations (1), but, this time, the vector of inputs (u) has two components (the rudder deflection δ_r and the ailerons deflection δ_a). The matrices A and B are $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$; in this case $n = 4$, $m = 2$ (the system has 4 states and 2 inputs). The pair (A, B) is assumed to be controllable and two from the four sensors of the systems are broken; in this case, the algorithm designs a gain matrix F, having the form $F = \left[0_{m\times(n-m)}\ \tilde{K}\right]$ such that the resulting optimal control law $u_2=Fx$ has the same effect (stabilization effect) on the aircraft motion; obviously, the number of broken sensors equals the number of null columns in the matrix F. The design of the gain matrix F is made by solving the matriceal equation (4), too. As in the previous case, the matrices A, B, K, and F are partitioned by using (5), with $A_{11} \in \mathbb{R}^{(n-m)\times(n-m)}$, $A_{12}, B_1 \in \mathbb{R}^{(n-m)\times m}$, $A_{21}, K_1 \in \mathbb{R}^{m\times(n-m)}$, $A_{22}, B_2, K_2, \tilde{K} \in \mathbb{R}^{m\times m}$. We obtain again the equations system (6), but, this time, the dimensions of the matrices are different; the system is now a system with 8 equations and 4 unknown variables. We split it into two new systems: the former is a linear system with 4 equations and 4 unknown variables (the system associated to the first equation (6)), while the latter is a nonlinear system with 4 equations and 4 unknown variables. Using the following notations:

$$\begin{gathered}
A_{22} = \begin{bmatrix} a_{33} & a_{34} \\ a_{43} & a_{44} \end{bmatrix}, B_2 = \begin{bmatrix} b_{31} & b_{32} \\ b_{41} & b_{42} \end{bmatrix}, \tilde{K} = \begin{bmatrix} \tilde{k}_1 & \tilde{k}_2 \\ \tilde{k}_3 & \tilde{k}_4 \end{bmatrix}, \Gamma = \begin{bmatrix} \Gamma_1 & \Gamma_2 \\ \Gamma_3 & \Gamma_4 \end{bmatrix} = K_1\left(A_{11} - B_1K_1\right) + K_2\left(A_{21} - B_2K_1\right), \\
\Phi = \begin{bmatrix} \Phi_1 & \Phi_2 \\ \Phi_3 & \Phi_4 \end{bmatrix} = K_1\left(A_{12} - B_1K_2\right) + K_2\left(A_{22} - B_2K_2\right).
\end{gathered} \tag{11}$$

The linear system becomes $A_{21}\widetilde{K}^T = \Gamma^T$ and, from this equation, $\widetilde{K}_0$ (the intermediary value of $\widetilde{K}$) is obtained. If A_{21} is not singular, it results $\widetilde{K}^T = A_{21}^{-1}\Gamma^T$; otherwise, instead of A_{21}^{-1}, the pseudo-inverse A_{21}^{+} will be used. Thus, the solving of the linear system can be achieved directly (by using A_{21}^{-1} or A_{21}^{+}) or by means of the least square method [13]. Matriceal equation $A_{21}\widetilde{K}^T = \Gamma^T$ is transformed into an algebraic equations system:

$$\begin{cases} a_{31}\widetilde{k}_1 + a_{41}\widetilde{k}_2 = \Gamma_1\,, a_{32}\widetilde{k}_1 + a_{42}\widetilde{k}_2 = \Gamma_2\,, \\ a_{31}\widetilde{k}_3 + a_{41}\widetilde{k}_4 = \Gamma_3\,, a_{32}\widetilde{k}_3 + a_{42}\widetilde{k}_4 = \Gamma_4\,. \end{cases} \tag{12}$$

The nonlinear system, associated to the second matriceal equation (6) is:

$$\begin{cases} \widetilde{k}_1\left(a_{33} - b_{31}\widetilde{k}_1 - b_{32}\widetilde{k}_3\right) + \widetilde{k}_2\left(a_{43} - b_{41}\widetilde{k}_1 - b_{42}\widetilde{k}_3\right) = \Phi_1, \widetilde{k}_1\left(a_{34} - b_{31}\widetilde{k}_2 - b_{32}\widetilde{k}_4\right) + \widetilde{k}_2\left(a_{44} - b_{41}\widetilde{k}_2 - b_{42}\widetilde{k}_4\right) = \Phi_2, \\ \widetilde{k}_3\left(a_{33} - b_{31}\widetilde{k}_1 - b_{32}\widetilde{k}_3\right) + \widetilde{k}_4\left(a_{43} - b_{41}\widetilde{k}_1 - b_{42}\widetilde{k}_3\right) = \Phi_3, \widetilde{k}_3\left(a_{34} - b_{31}\widetilde{k}_2 - b_{32}\widetilde{k}_4\right) + \widetilde{k}_4\left(a_{44} - b_{41}\widetilde{k}_2 - b_{42}\widetilde{k}_4\right) = \Phi_4, \end{cases} \tag{13}$$

and it is solved by means of the Newton successive approximations method [15]. Denoting:

$$\begin{aligned} f_1\left(\widetilde{k}_1, \widetilde{k}_2, \widetilde{k}_3, \widetilde{k}_4\right) &= \widetilde{k}_1\left(a_{33} - b_{31}\widetilde{k}_1 - b_{32}\widetilde{k}_3\right) + \widetilde{k}_2\left(a_{43} - b_{41}\widetilde{k}_1 - b_{42}\widetilde{k}_3\right) - \Phi_1\,, \\ f_2\left(\widetilde{k}_1, \widetilde{k}_2, \widetilde{k}_3, \widetilde{k}_4\right) &= \widetilde{k}_1\left(a_{34} - b_{31}\widetilde{k}_2 - b_{32}\widetilde{k}_4\right) + \widetilde{k}_2\left(a_{44} - b_{41}\widetilde{k}_2 - b_{42}\widetilde{k}_4\right) - \Phi_2\,, \\ f_3\left(\widetilde{k}_1, \widetilde{k}_2, \widetilde{k}_3, \widetilde{k}_4\right) &= \widetilde{k}_3\left(a_{33} - b_{31}\widetilde{k}_1 - b_{32}\widetilde{k}_3\right) + \widetilde{k}_4\left(a_{43} - b_{41}\widetilde{k}_1 - b_{42}\widetilde{k}_3\right) - \Phi_3\,, \\ f_4\left(\widetilde{k}_1, \widetilde{k}_2, \widetilde{k}_3, \widetilde{k}_4\right) &= \widetilde{k}_3\left(a_{34} - b_{31}\widetilde{k}_2 - b_{32}\widetilde{k}_4\right) + \widetilde{k}_4\left(a_{44} - b_{41}\widetilde{k}_2 - b_{42}\widetilde{k}_4\right) - \Phi_4, \end{aligned} \tag{14}$$

the system (13) gets the form:

$$\begin{cases} f_1\left(\widetilde{k}_1, \widetilde{k}_2, \widetilde{k}_3, \widetilde{k}_4\right) = 0\,, f_2\left(\widetilde{k}_1, \widetilde{k}_2, \widetilde{k}_3, \widetilde{k}_4\right) = 0\,, \\ f_3\left(\widetilde{k}_1, \widetilde{k}_2, \widetilde{k}_3, \widetilde{k}_4\right) = 0\,, f_4\left(\widetilde{k}_1, \widetilde{k}_2, \widetilde{k}_3, \widetilde{k}_4\right) = 0\,. \end{cases} \tag{15}$$

Using the Taylor series of the functions $f_i,\ i = \overline{1,4}$, around the point $P_0\left(\widetilde{k}_{1_0}, \widetilde{k}_{2_0}, \widetilde{k}_{3_0}, \widetilde{k}_{4_0}\right)$ and considering the current point $P\left(\widetilde{k}_1, \widetilde{k}_2, \widetilde{k}_3, \widetilde{k}_4\right)$, it results:

$$\begin{cases} f_1(P_0) + \dfrac{\partial f_1(P_0)}{\partial \widetilde{k}_1}\left(\widetilde{k}_1 - \widetilde{k}_{1_0}\right) + \dfrac{\partial f_1(P_0)}{\partial \widetilde{k}_2}\left(\widetilde{k}_2 - \widetilde{k}_{2_0}\right) + \dfrac{\partial f_1(P_0)}{\partial \widetilde{k}_3}\left(\widetilde{k}_3 - \widetilde{k}_{3_0}\right) + \dfrac{\partial f_1(P_0)}{\partial \widetilde{k}_4}\left(\widetilde{k}_4 - \widetilde{k}_{4_0}\right) + O\left(\left\|P - P_0\right\|\right) = 0, \\ f_2(P_0) + \dfrac{\partial f_2(P_0)}{\partial \widetilde{k}_1}\left(\widetilde{k}_1 - \widetilde{k}_{1_0}\right) + \dfrac{\partial f_2(P_0)}{\partial \widetilde{k}_2}\left(\widetilde{k}_2 - \widetilde{k}_{2_0}\right) + \dfrac{\partial f_2(P_0)}{\partial \widetilde{k}_3}\left(\widetilde{k}_3 - \widetilde{k}_{3_0}\right) + \dfrac{\partial f_2(P_0)}{\partial \widetilde{k}_4}\left(\widetilde{k}_4 - \widetilde{k}_{4_0}\right) + O\left(\left\|P - P_0\right\|\right) = 0, \\ f_3(P_0) + \dfrac{\partial f_3(P_0)}{\partial \widetilde{k}_1}\left(\widetilde{k}_1 - \widetilde{k}_{1_0}\right) + \dfrac{\partial f_3(P_0)}{\partial \widetilde{k}_2}\left(\widetilde{k}_2 - \widetilde{k}_{2_0}\right) + \dfrac{\partial f_3(P_0)}{\partial \widetilde{k}_3}\left(\widetilde{k}_3 - \widetilde{k}_{3_0}\right) + \dfrac{\partial f_3(P_0)}{\partial \widetilde{k}_4}\left(\widetilde{k}_4 - \widetilde{k}_{4_0}\right) + O\left(\left\|P - P_0\right\|\right) = 0, \\ f_4(P_0) + \dfrac{\partial f_4(P_0)}{\partial \widetilde{k}_1}\left(\widetilde{k}_1 - \widetilde{k}_{1_0}\right) + \dfrac{\partial f_4(P_0)}{\partial \widetilde{k}_2}\left(\widetilde{k}_2 - \widetilde{k}_{2_0}\right) + \dfrac{\partial f_4(P_0)}{\partial \widetilde{k}_3}\left(\widetilde{k}_3 - \widetilde{k}_{3_0}\right) + \dfrac{\partial f_4(P_0)}{\partial \widetilde{k}_4}\left(\widetilde{k}_4 - \widetilde{k}_{4_0}\right) + O\left(\left\|P - P_0\right\|\right) = 0; \end{cases} \tag{16}$$

the very small terms are denoted above with $O\left(\left\|P - P_0\right\|\right)$. Neglecting these very small terms, it yields:

$$\begin{cases} D_{11}\widetilde{k}_1 + D_{12}\widetilde{k}_2 + D_{13}\widetilde{k}_3 + D_{14}\widetilde{k}_4 = E_1\,, D_{21}\widetilde{k}_1 + D_{22}\widetilde{k}_2 + D_{23}\widetilde{k}_3 + D_{24}\widetilde{k}_4 = E_2\,, \\ D_{31}\widetilde{k}_1 + D_{32}\widetilde{k}_2 + D_{33}\widetilde{k}_3 + D_{34}\widetilde{k}_4 = E_3\,, D_{41}\widetilde{k}_1 + D_{42}\widetilde{k}_2 + D_{43}\widetilde{k}_3 + D_{44}\widetilde{k}_4 = E_4\,, \end{cases} \tag{17}$$

with

$$\begin{gathered} D_{11} = \frac{\partial f_1(P_0)}{\partial \widetilde{k}_1}, D_{12} = \frac{\partial f_1(P_0)}{\partial \widetilde{k}_2}, D_{13} = \frac{\partial f_1(P_0)}{\partial \widetilde{k}_3}, D_{14} = \frac{\partial f_1(P_0)}{\partial \widetilde{k}_4}, D_{21} = \frac{\partial f_2(P_0)}{\partial \widetilde{k}_1}, D_{22} = \frac{\partial f_2(P_0)}{\partial \widetilde{k}_2}, D_{23} = \frac{\partial f_2(P_0)}{\partial \widetilde{k}_3}, D_{24} = \frac{\partial f_2(P_0)}{\partial \widetilde{k}_4}, \\ D_{31} = \frac{\partial f_3(P_0)}{\partial \widetilde{k}_1}, D_{32} = \frac{\partial f_3(P_0)}{\partial \widetilde{k}_2}, D_{33} = \frac{\partial f_3(P_0)}{\partial \widetilde{k}_3}, D_{34} = \frac{\partial f_3(P_0)}{\partial \widetilde{k}_4}, D_{41} = \frac{\partial f_4(P_0)}{\partial \widetilde{k}_1}, D_{42} = \frac{\partial f_4(P_0)}{\partial \widetilde{k}_2}, D_{43} = \frac{\partial f_4(P_0)}{\partial \widetilde{k}_3}, D_{44} = \frac{\partial f_4(P_0)}{\partial \widetilde{k}_4}, \\ E_1 = -f_1(P_0) + D_{11}\widetilde{k}_{1_0} + D_{12}\widetilde{k}_{2_0} + D_{13}\widetilde{k}_{3_0} + D_{14}\widetilde{k}_{4_0}, E_2 = -f_2(P_0) + D_{21}\widetilde{k}_{1_0} + D_{22}\widetilde{k}_{2_0} + D_{23}\widetilde{k}_{3_0} + D_{24}\widetilde{k}_{4_0}, \\ E_3 = -f_3(P_0) + D_{31}\widetilde{k}_{1_0} + D_{32}\widetilde{k}_{2_0} + D_{33}\widetilde{k}_{3_0} + D_{34}\widetilde{k}_{4_0}, E_4 = -f_4(P_0) + D_{41}\widetilde{k}_{1_0} + D_{42}\widetilde{k}_{2_0} + D_{43}\widetilde{k}_{3_0} + D_{44}\widetilde{k}_{4_0}. \end{gathered} \tag{18}$$

The system (13) became a linear system of form (17) and it is solved iteratively [15]. We start from an initial solution $P_0(\tilde{k}_{1_0}, \tilde{k}_{2_0}, \tilde{k}_{3_0}, \tilde{k}_{4_0})$ - solution of the linear system (12) and we choose a convergence error (for example $\varepsilon = 10^{-4}$). The coefficients D_{ij} and the terms $E_i\left(i, j = \overline{1, n}\right)$ are calculated and the linear system (17) is solved with the least square method; it results the solution at iteration 1 - $P_1(\tilde{k}_{1_1}, \tilde{k}_{2_1}, \tilde{k}_{3_1}, \tilde{k}_{4_1})$. If this solution satisfies the conditions $\left|\tilde{k}_{1_1} - \tilde{k}_{1_0}\right| < \varepsilon, \left|\tilde{k}_{2_1} - \tilde{k}_{2_0}\right| < \varepsilon, \left|\tilde{k}_{3_1} - \tilde{k}_{3_0}\right| < \varepsilon, \left|\tilde{k}_{4_1} - \tilde{k}_{4_0}\right| < \varepsilon,$ the solution P_1 is good; otherwise, the iterative process continues: $P_0(\tilde{k}_{1_0}, \tilde{k}_{2_0}, \tilde{k}_{3_0}, \tilde{k}_{4_0}) \leftarrow P_1(\tilde{k}_{1_1}, \tilde{k}_{2_1}, \tilde{k}_{3_1}, \tilde{k}_{4_1})$, the coefficients D_{ij} and the terms $E_i\left(i, j = \overline{1, n}\right)$ are again calculated, and the linear system (17) is again solved until, at iteration "i", the conditions $\left|\tilde{k}_{1_i} - \tilde{k}_{1_0}\right| < \varepsilon, \left|\tilde{k}_{2_i} - \tilde{k}_{2_0}\right| < \varepsilon, \left|\tilde{k}_{3_i} - \tilde{k}_{3_0}\right| < \varepsilon, \left|\tilde{k}_{4_i} - \tilde{k}_{4_0}\right| < \varepsilon$ are simultaneously satisfied. The coefficients D_{ij} are calculated in Matlab environment as follows:

$$
\begin{aligned}
D_{i1} &= \frac{f_i(\tilde{k}_{1_0} + h, \tilde{k}_{2_0}, \tilde{k}_{3_0}, \tilde{k}_{4_0}) - f_i(\tilde{k}_{1_0}, \tilde{k}_{2_0}, \tilde{k}_{3_0}, \tilde{k}_{4_0})}{h}, D_{i2} = \frac{f_i(\tilde{k}_{1_0}, \tilde{k}_{2_0} + h, \tilde{k}_{3_0}, \tilde{k}_{4_0}) - f_i(\tilde{k}_{1_0}, \tilde{k}_{2_0}, \tilde{k}_{3_0}, \tilde{k}_{4_0})}{h}, \\
D_{i3} &= \frac{f_i(\tilde{k}_{1_0}, \tilde{k}_{2_0}, \tilde{k}_{3_0} + h, \tilde{k}_{4_0}) - f_i(\tilde{k}_{1_0}, \tilde{k}_{2_0}, \tilde{k}_{3_0}, \tilde{k}_{4_0})}{h}, D_{i4} = \frac{f_i(\tilde{k}_{1_0}, \tilde{k}_{2_0}, \tilde{k}_{3_0}, \tilde{k}_{4_0} + h) - f_i(\tilde{k}_{1_0}, \tilde{k}_{2_0}, \tilde{k}_{3_0}, \tilde{k}_{4_0})}{h},
\end{aligned} \tag{19}
$$

where h is chosen, for example, 10^{-2}. The general formula for the calculation of the coefficients E_i is: $E_i = -f_i(P_0) + D_{i1}\tilde{k}_{1_0} + D_{i2}\tilde{k}_{2_0} + D_{i3}\tilde{k}_{3_0} + D_{i4}\tilde{k}_{4_0}$, with $i = \overline{1, n}$. After solving the system (17) and after the determination of the solutions $\tilde{k}_1, \tilde{k}_2, \tilde{k}_3, \tilde{k}_4$, the matrix F is built as follows: $F = \left[0_{m \times (n-m)} \;\; \tilde{K}\right] = \begin{bmatrix} 0 & 0 & \tilde{k}_1 & \tilde{k}_2 \\ 0 & 0 & \tilde{k}_3 & \tilde{k}_4 \end{bmatrix}$, and the stability of the closed loop system is analyzed. If $\sigma(A - BF) \notin C_-$, we determine again the gain matrix K (by using the ALGLX algorithm or any other algorithm) and all the above calculations are repeated until $\sigma(A - BF) \in C_-$. The above designed reconfigurable controller (lateral motion) is useful not only in the case of sensor failures. In the case of an actuator failure, our approach can be applied after an appropriate modification of the matrix B; thus, the column which corresponds to the failure actuator is replaced by a column of zeros and the matrix F is determined in the same way.

The iterative ALG_00K algorithm, for the stabilization of the aircraft's lateral motion, in the case of sensor or actuator failures, is summarized as follows: **Step 1:** We change the order of the state equations which describe the aircraft's lateral motion such that the first equations are associated with the variables measured by means of the broken sensors, while the last state equations correspond to the variables measured by means of the functional sensors; the change of the states' order is equivalent to a change of the lines' order in the state equation matrices. We check if the matrix A_{21} is nonsingular. If the condition is fulfilled one calculates A_{21}^{-1}; otherwise the pseudo-inverse of the matrix A_{21} $\left(A_{21}^{+}\right)$ is obtained; **Step 2:** We calculate the matrix K with the ALGLX algorithm [12]; **Step 3:** We partitionate the matrices A, B, K, and F according to equation (5); **Step 4:** The matrices Γ and Φ are calculated by using equation (11); **Step 5:** The equations system (12) is solved with the least square method and it results the intermediary solution $\tilde{K}_0$. Instead of the least square method, the inverse or pseudo-inverse of the matrix A_{21} can be used; **Step 6:** We iteratively solve the nonlinear system (13) by using the Newton successive approximations method [15] and we obtain $\tilde{k}_1, \tilde{k}_2, \tilde{k}_3, \tilde{k}_4$; this is equivalent to the solving of the linear system (17); **Step 7:** The matrix F $\left(F = \left[0_{m \times (n-m)} \;\; \tilde{K}\right]\right)$ is obtained and the closed loop eigenvalues are calculated. If $\sigma(A - BF) \in C_-$ the gain matrix F has been correctly determined; otherwise, the steps 2-7 are again run until the system is stable.

4. SIMULATION RESULTS

The validation of the ALG_00K algorithm is carried out for 2 concrete examples (longitudinal and lateral motions of the aircrafts).

A. Example 1 (longitudinal motion)

Let us consider the longitudinal motion of an aircraft [16]:

$$\begin{bmatrix}\Delta\dot{u}\\ \Delta\dot{\alpha}\\ \Delta\dot{q}\\ \Delta\dot{\theta}\end{bmatrix}=\begin{bmatrix}-0.0188 & -6.0736 & -13.3897 & -9.7183\\ -0.002 & -0.87 & 0.991 & -0.0131\\ 0 & -1.328 & 0 & 0\\ 0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}\Delta u\\ \Delta\alpha\\ \Delta q\\ \Delta\theta\end{bmatrix}+\begin{bmatrix}0\\ -0.4321\\ -3.0499\\ 0\end{bmatrix}\delta_e\,; \tag{20}$$

here u is the component of the aircraft velocity along the aircraft longitudinal axis (OX), α – the aircraft attack angle, θ - the aircraft pitch angle, q – the aircraft pitch angular rate, δ_e is the elevator deflection, while Δ is associated with the perturbation of the variables from their nominal values. The software implementation of the algorithm has been carried out in Matlab environment. Resuming the ALG_00K algorithm, we obtained F=[0 0 –7.8056 –4.8996]. In Fig.1 we present the time histories of the system states and input $u = \delta_e$ with two line types: continuous line – classical controller (based on the ALGLX algorithm) and dashed line – new reconfigurable controller (based on the ALG_00K algorithm) for active fault-tolerant control systems.

As one can see in this example, the matrix F assures the stability of the system and all the states and input variables tend to their desired values. Making a parallel between the classical controller and our new reconfigurable controller we remark an insignificant increase of the overshoot for the case of ALG_00K approach. This fact is not important in the undesirable case of sensor or actuator failures; the aircraft is stabilized by using the information from the other sensors and the overshoot increase remains secondary. In this example, the classical controller is based on the ALGLX algorithm but this approach can be replaced by other already existing algorithms. From the transient regime point of view, there are not significant differences between the two controllers; our reconfigurable controller's main advantages are related to a reduction in the number of embedded sensors, a decrease in costs, and the cancellation of the adverse effect on sensor errors.

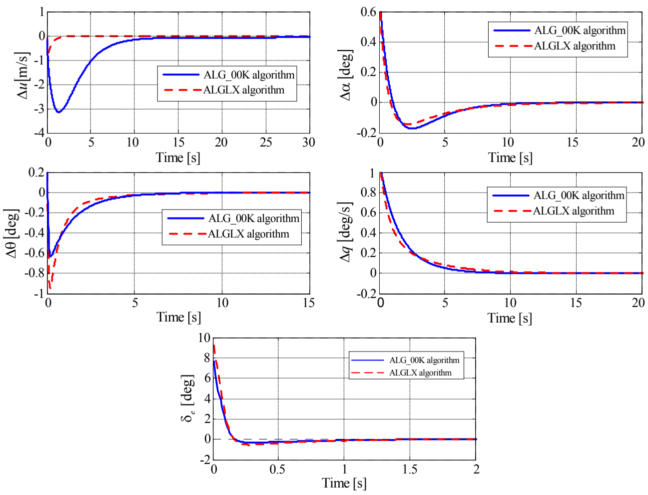

Fig. 1 – Time histories of the states and input variable for classical controller and FTCS reconfigurable controller (longitudinal motion).

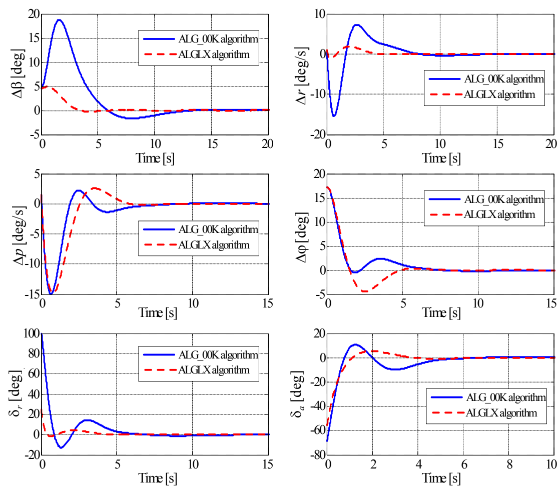

Fig. 2 – Time histories of the states and input variable for classical controller and FTCS reconfigurable controller (lateral motion).

B. Example 2 (lateral motion)

Now, we study the case of the lateral motion of an aircraft flying with Mach number M = 0.8 at the altitude H = 4000 ft [17]; the state equation of the lateral motion is:

$$\begin{bmatrix}\Delta\dot{\beta}\\ \Delta\dot{r}\\ \Delta\dot{p}\\ \Delta\dot{\varphi}\end{bmatrix}=\begin{bmatrix}-0.0558 & -0.9968 & 0.0802 & 0.0415\\ 0.598 & -0.115 & -0.0318 & 0\\ 0.305 & 0.388 & -0.465 & 0\\ 0 & 0.0805 & 1 & 0\end{bmatrix}\begin{bmatrix}\Delta\beta\\ \Delta r\\ \Delta p\\ \Delta\varphi\end{bmatrix}+\begin{bmatrix}0.0073 & 0\\ -0.475 & 0.123\\ 0.153 & 1.063\\ 0 & 0\end{bmatrix}\begin{bmatrix}\delta_r\\ \delta_a\end{bmatrix}, \tag{21}$$

where β is the aircraft sideslip angle, r - the aircraft yaw angular rate, p - the aircraft roll angular rate, φ - the aircraft roll angle, δ_r - the rudder deflection, and δ_a - the deflection of the ailerons. Resuming, step by step, the ALG_00K algorithm, we get $\widetilde{k}_1 = -1.99, \widetilde{k}_2 = -3.18, \widetilde{k}_3 = 1.68, \widetilde{k}_4 = 2.86$. To determine the unknown variables $\widetilde{k}_1, \widetilde{k}_2, \widetilde{k}_3, \widetilde{k}_4$, the Newton successive approximations method [15] can be used. In Fig.2 we presented the time histories of the system 4 states and inputs $u_1 = \delta_r$, $u_2 = \delta_a$ with the same two line types as in Fig.1.

The algorithm resumes steps 2-7 until the closed loop system is stable. From the aircraft lateral motion example one remarks again the convergence of the ALG_00K algorithm – the matrix F assures the stability of the system and all the states and inputs tend to their desired values. Same remarks, as in the longitudinal motion case, result from the overshoot and transient regime point of view. When the system is affected by sensors or actuators failure, our new reconfigurable controller may be successfully used as a subsystem of a fault-tolerant control system; the information from the broken sensors or actuators are not used and the stabilization of the system is carried out by using information only from the functional sensors and actuators.

5. CONCLUSIONS

In this paper we propose a new reconfigurable controller for active fault-tolerant control systems with applicability to aircraft's flight control. The reconfigurable controller is based on a new algorithm for the calculation of the controller gain matrix without using information from a full-order or reduced order observer. The main advantages of our reconfigurable controller, when no failures appear in the system, are related to a reduction in the number of embedded sensors, a decrease in costs, and the cancellation of the adverse effect on sensor errors. When the system is affected by sensors or actuators failure, our new reconfigurable controller may be successfully used as a subsystem of a fault-tolerant control system; the information from the broken sensors or actuators are not used and the stabilization of the system is carried out using information only from the functional sensors and actuators. The new controller "makes" the FTCS to react to the system component failures actively by reconfiguring control actions such that the stability and acceptable performance of the entire system can be maintained.

REFERENCES

1. NOURA, H., THEILLIOL, D., PONSARD, J., CHAMSEDDINE, A., *Fault-tolerant control systems. Design and practical applications*, Springer, 2009.
2. ZHANG, Y., JIANG, J., *Bibliographical review on reconfigurable fault-tolerant control systems*, Annual Reviews in Control, **32**, 2008.
3. STEINBERG, M., *Historical overview of research in reconfigurable flight control*, Proceedings of IMechE, Part G: Journal of Aerospace Engineering, **219**, 2005, pp. 263–275.
4. LIAO, F., WANG, J.L., YANG, G.H., *Reliable robust flight tracking control: An LMI approach*, IEEE Transactions on Control Systems Technology, **10**, *1*, 2002, pp. 76–89.
5. LUNGU, R., LUNGU, M., ROTARU, C. *Non-linear Adaptive System for the Control of the Helicopters Pitch's Angle*. Proceedings of the Romanian Academy, Series A: Mathematics, Physics, Technical Sciences, Information Science, **12**, *2*, 2011, pp. 133–142.
6. ZHANG, Y.M., JIANG, J., *Integrated active fault-tolerant control using IMM approach*, IEEE Transactions on Aerospace and Electronic Systems, **37**, *4*, 2001, pp. 1221–1235.
7. PATTON, R.J., *Fault-tolerant control: The 1997 situation*, Proceedings of the 3rd IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes, 1997, pp. 1033–1055.
8. ZHANG, Y.M., JIANG, J., *An active fault-tolerant control system against partial actuator failures*, IEE Proceedings – Control Theory and Applications, **149**, *1*, 2002, pp. 95–104.
9. JIANG, J., *Fault-tolerant control system – An introductory overview*, Acta Automatica Sinica, **31**, *1*, 2005, pp. 161–174.
10. HOW, J.P., *Aircraft stability and control*, Lecture notes, 2004.
11. SKELTON, R.E., IWASAKI, T., GRIGORIADIS, K., *A unified algebraic approach to linear control design*, Taylor and Francis, Boca Raton, FL, 1998.
12. LUNGU, M., *Sisteme de conducere a zborului*, Craiova, Sitech Publisher, 2008.
13. GHINEA, M., FIRETEANU, V., *Matlab – calcul numeric – grafică – aplicații*, Teora Publisher, 2001.

14. BERBENTE, C., ZANCU, S., TĂTARANU, C., MITRON, S., *Metode numerice de calcul pentru aviație*, Bucharest Polytechnic Institute, 1998.
15. YU, K.M., LISIENKO, V.G., SAPLIN, A.V., *Modified Newton-Raphson method for solving a system of nonlinear equations in problems of complicated heat exchange*, Journal of Engineering, Physics and Thermo-Physics, **61**, *3*, 1991, pp. 1168–1174.
16. DIEDRICH, A.J., *The Multidisciplinary Design and Optimization of an Unconventional, Extremely Quiet Transport Aircraft*, Master Thesis, 2005.
17. WEI, X., VERHAEGEN, M., *Robust fault detection observer design for linear uncertain systems*, International Journal of Control, **84**, *1*, 2011, pp. 197–215.

Received May 27, 2013