



**HAL**  
open science

# Neural Networks Based Aircraft Fault Tolerant Control

Lunlong Zhong, Felix Mora-Camino

► **To cite this version:**

Lunlong Zhong, Felix Mora-Camino. Neural Networks Based Aircraft Fault Tolerant Control. ATIO 2012, 12th AIAA Aviation Technology, Integration, and Operations Conference and 14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Sep 2012, Indianapolis, United States. pp xxxx, 10.2514/6.2012-5683 . hal-00926483

**HAL Id: hal-00926483**

**<https://hal-enac.archives-ouvertes.fr/hal-00926483>**

Submitted on 14 Jan 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Neural Networks Based Aircraft Fault Tolerant Control

Lunlong Zhong<sup>1</sup>

MAIAA-Automation Research Group, ENAC, Toulouse, 31055, France

Félix Mora-Camino<sup>2</sup>

MAIAA-Automation Research Group, ENAC, Toulouse, 31055, France

The purpose of this communication is to deal with the case in which an aerodynamic actuator failure occurs to an aircraft while it has to perform guidance maneuvers. The problem considered deals with the reallocation of redundant actuators to perform the required maneuvers and maintain the structural integrity of the aircraft. A Nonlinear Inverse Control technique is used to generate online nominal moment along the three axis of the aircraft. Then, taking into account all material and structural constraints as well as the redundant effects from other actuators, a Mathematical Programming problem to be solved online is considered. The proposed solution method is based on dynamic neural networks which appears to provide acceptable response times. Simulation results are displayed and new development perspectives are discussed.

## Nomenclature

$\alpha$	=	angle of attack, <i>rad</i>
$\beta$	=	sideslip angle, <i>rad</i>
$\delta_p, \delta_q, \delta_r$	=	roll, pitch, yaw actuator deflections, <i>rad</i>
$p, q, r$	=	roll, pitch, yaw rates, <i>rad/s</i>
$V$	=	airspeed, <i>m/s</i>

## I. Introduction

In this paper we consider a transportation aircraft in the situation in which a main aerodynamic actuator failure occurs while it has to perform guidance maneuvers. Here using dynamic inversion of flight dynamics, the necessary moments to perform a given guidance maneuver are computed, and then an optimization problem is considered to generate on-line reference values for the fault free actuators. This represents the main difference with other previous approaches to actuator fault management<sup>1-4</sup>. In the case considered, a linear quadratic programming formulation of the optimization problem can be adopted and a neural network approach to get on-line solution is discussed.

## II. Modeling the Effectiveness of Aerodynamic Actuators

The effectiveness of the control surfaces is made apparent by their contribution to the dimensionless coefficients present in the expressions of aerodynamic forces and moments<sup>5</sup>. Then the angular deflections of these control surfaces produce a collective effect over the aircraft which should satisfy structural constraints. The global dimensionless coefficients used to express aerodynamic forces can be given by:

$$C_x = C_{x0} + k C_z^2 \quad (1.1)$$

$$C_y = C_{y\beta} \beta + C_{yp} p l_A / V + C_{yr} r l_A / V + \underline{C}_{y\delta p} \delta_p + \underline{C}_{y\delta r} \delta_r \quad (1.2)$$

$$C_z = C_{z0} + C_{z\alpha} \alpha + C_{z\delta hs} \delta_{hs} + \underline{C}_{z\delta q} \delta_q \quad (1.3)$$

where standard scalar notation for main aerodynamic actuators is replaced by a vector one where each elementary

---

<sup>1</sup> PhD student at MAIAA, Automation Research Group, ENAC, lunlong.zhong@enac.fr

<sup>2</sup> Professor, ENAC, felix.mora@enac.fr

aerodynamic surface is distinguished and assigned to its main effect (roll, pitch or yaw effect) and different coefficient  $C_{ij}$  are also dimensionless. The non dimensional coefficients of the different aerodynamic moments can in general be expressed such as:

$$C_m = C_{m0} + C_{m\alpha} \alpha + C_{mq} q l_A / V + C_{m\delta_{hs}} \delta_{hs} + \underline{C}_{m\delta_q} \delta_q \quad (2.1)$$

$$C_l = C_{l0} + C_{l\beta} \beta + C_{lp} p l_A / V + C_{lr} r l_A / V + \underline{C}_{l\delta_p} \delta_p + \underline{C}_{l\delta_r} \delta_r \quad (2.2)$$

$$C_n = C_{n0} + C_{n\beta} \beta + C_{np} p l_A / V + C_{nr} r l_A / V + \underline{C}_{n\delta_p} \delta_p + \underline{C}_{n\delta_r} \delta_r \quad (2.3)$$

The expression of the different aerodynamic moments generated by the control surfaces can be approximated by an affine form with respect to the corresponding deflections of the different aerodynamic actuators, so that we get expressions such as:

$$M_{ik} = M_{ik}^0 + \mu_{ik} \delta_k \quad (3)$$

where  $M_{ik}$  is the  $i^{th}$  considered moment (roll, pitch, yaw, bending, flexion),  $\delta_k$  is the deflection of the  $k^{th}$  aerodynamic actuator ( $k \in K = \{\text{aileron, flap, right spoilers, left spoilers, elevator, rudder}\}$ ) and  $\mu_{ik}$  is the current effectiveness of actuator  $k$  to produce moment  $i$ . The current values  $M_{ik}^0(t)$  and  $\mu_{ik}(t)$  depend on the airspeed  $V$  of the aircraft, its flight level  $z$  and on the values of the main motion variables  $\alpha, \beta, p, q$  and  $r$ . Global aerodynamic moments generated by aircraft aerodynamic actuators can be rewritten in an affine form as:

$$L(t) = L^0(t) + \sum_{i \in I^L} X_i^L(t) \delta_i(t) \quad (4.1)$$

$$M(t) = M^0(t) + \sum_{i \in I^M} X_i^M(t) \delta_i(t) \quad (4.2)$$

$$N(t) = N^0(t) + \sum_{i \in I^N} X_i^N(t) \delta_i(t) \quad (4.3)$$

with  $I = I^L \cup I^M \cup I^N$ , where  $I^L$  is the set of actuators able to generate roll moment,  $I^N$  is the set of actuators able to generate yaw moment, while  $I^M$  is the set of actuators generating pitch moment. Fig. 1 displays, in the case of a A340 aircraft, the available wing actuators. The current values of  $L^0(t)$ ,  $X_i^L(t)$ ,  $M^0(t)$ ,  $X_i^M(t)$ ,  $N^0(t)$  and  $X_i^N(t)$  depend on the airspeed  $V$  of the aircraft, its flight level  $z$  and on the values of the main motion variables  $\alpha, \beta, p, q$  and  $r$ .

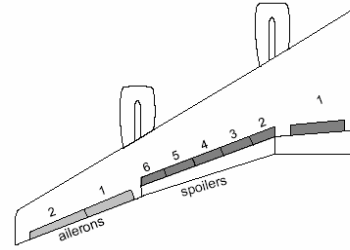


Figure 1. Example of Wing Actuators (A340)

### III. Actuators constraints and limitations

The operation of the different actuators must satisfy to global and local physical constraints which must be taken into account by any flight control system.

#### A. Actuators position and speed limitations

With respect to control surfaces, the following typical constraints should be met:

$$\delta_i^{\min} \leq \delta_i \leq \delta_i^{\max} \quad i \in I \quad (5.1)$$

$$\dot{\delta}_i^{\min} \leq \dot{\delta}_i \leq \dot{\delta}_i^{\max} \quad i \in I \quad (5.2)$$

where  $\delta_i^{\min}$ ,  $\delta_i^{\max}$ ,  $\dot{\delta}_i^{\min}$  and  $\dot{\delta}_i^{\max}$  are extreme position and speed values. These conditions can be considered at sampled instants, it becomes:

$$\max\{\delta_i^{\min}, \delta_i(t - \Delta t) + \dot{\delta}_i^{\min} \Delta t\} \leq \delta_i(t) \quad (6.1)$$

and

$$\delta_i(t) \leq \min\{\delta_i^{\max}, \delta_i(t - \Delta t) + \dot{\delta}_i^{\max} \Delta t\} \quad (6.2)$$

## B. Global constraints

Global constraints are in general related with structural considerations. It can be shown that total wing bending and torsion moments during maneuver can be written in an affine form as<sup>6</sup>:

$$M_b(t) = A_b(t) + \sum_{i \in I^{wing}} Y_{bi}(t) \delta_i(t) \quad (7.1)$$

and

$$M_f(t) = A_f(t) + \sum_{i \in I^{wing}} Y_{fi}(t) \delta_i(t) \quad (7.2)$$

with  $I^{wing} \subset I$  is the set of wing actuators contributing to the bending and torsion moments, where  $A_b$ ,  $Y_{bi}$ ,  $A_f$  and  $Y_{fi}$  depend also on the airspeed  $V$  of the aircraft, its flight level  $z$  and on the values of motion variables  $\alpha$ ,  $\beta$ ,  $p$ ,  $q$  and  $r$ .

Then the global wing bending and flexion constraints can be written as:

$$A_b(t) + \sum_{i \in I^{wing}} Y_{bi}(t) \delta_i(t) \leq M_{bend}^{\max} \quad (8.1)$$

and

$$A_f(t) + \sum_{i \in I^{wing}} Y_{fi}(t) \delta_i(t) \leq M_{tor}^{\max} \quad (8.2)$$

where  $M_{bend}^{\max}$  and  $M_{tor}^{\max}$  are maximum acceptable bending and torsion moments at the wing root. Here it is supposed that the satisfaction of these global constraints implies the satisfaction of local bending and torsion moments constraints.

## IV. Actuators Allocation for Maneuvering Aircraft

Here we study the case of a pure stabilized roll maneuver where the following conditions should be met by the body angular rates of the aircraft:

$$\tau_p \dot{p} + p = p_c \quad (9.1)$$

$$q = 0 \quad (9.2)$$

$$\tau_r \dot{r} + r = (g/V) \sin \phi \quad (9.3)$$

where roll and yaw motions follow first order dynamics while pitch dynamics remains frozen. Here  $p_c$  is the desired roll rate and  $\tau_p$  and  $\tau_r$  are time constants. The dynamic constraint relative to the yaw rate is characteristic of an equilibrated turn, its completion should allow to avoid noticeable lateral load factors during this roll maneuver.

Applying the non linear inverse control approach<sup>7</sup>, we get the necessary on-line values for each aerodynamic moment:

$$\tilde{M}(t) = (A - C) r(t) p(t) + E(p(t)^2 - r(t)^2) \quad (10.1)$$

and

$$\begin{bmatrix} \tilde{L}(t) \\ \tilde{N}(t) \end{bmatrix} = \begin{bmatrix} A & -E \\ -E & C \end{bmatrix} \begin{bmatrix} \frac{1}{\tau_p} (p_c - p(t)) \\ \frac{1}{\tau_r} ((g/V(t)) \sin \phi(t) - r(t)) \end{bmatrix} \quad (10.2)$$

Here we consider the situation where the failure affects some of the commonly used actuators but some actuator

redundancy remains to try to perform maneuvers.

Depending on the remaining degree of redundancy between elementary actuators, it is possible that a solution matches exactly the following constraints:

$$\sum_{i \in I^L} X_i^L(t) \tilde{\delta}_i(t) = \tilde{L}(t) - L^0(t) \quad (11.1)$$

$$\sum_{i \in I^M} X_i^M(t) \tilde{\delta}_i(t) = \tilde{M}(t) - M^0(t) \quad (11.2)$$

$$\sum_{i \in I^N} X_i^N(t) \tilde{\delta}_i(t) = \tilde{N}(t) - N^0(t) \quad (11.3)$$

In this case the maneuver will be performed still in a standard way. Otherwise, an approximate maneuver should be defined. Here we propose to solve on-line the following linear quadratic problem given by:

$$\begin{aligned} \min_{\underline{\delta}} \quad & w_L \left( \sum_{i \in I^L} X_i^L(t) \tilde{\delta}_i(t) - \tilde{L}(t) + L^0(t) \right)^2 + w_M \left( \sum_{i \in I^M} X_i^M(t) \tilde{\delta}_i(t) - \tilde{M}(t) + M^0(t) \right)^2 \\ & + w_N \left( \sum_{i \in I^N} X_i^N(t) \tilde{\delta}_i(t) - \tilde{N}(t) + N^0(t) \right)^2 \end{aligned} \quad (12)$$

with the following constraints:

$$A_b(t) + \sum_{i \in I^{wing}} Y_{bi}(t) \tilde{\delta}_i(t) \leq M_{bend}^{\max} \quad (13.1)$$

$$A_f(t) + \sum_{i \in I^{wing}} Y_{fi}(t) \tilde{\delta}_i(t) \leq M_{tor}^{\max} \quad (13.2)$$

$$\delta_i^{\min} \leq \tilde{\delta}_i \leq \delta_i^{\max} \quad i \in I_{\bar{F}} \quad (14.1)$$

$$\tilde{\delta}_i^{\min} \leq \tilde{\delta}_i \leq \tilde{\delta}_i^{\max} \quad i \in I_{FL} \quad (14.2)$$

$$\max \left\{ \delta_i^{\min}, \tilde{\delta}_i(t - \Delta t) + \dot{\delta}_i^{\min} \Delta t \right\} \leq \tilde{\delta}_i(t) \quad i \in I_{\bar{F}} \quad (14.3)$$

$$\tilde{\delta}_i(t) \leq \min \left\{ \delta_i^{\max}, \tilde{\delta}_i(t - \Delta t) + \dot{\delta}_i^{\max} \Delta t \right\} \quad i \in I_{\bar{F}} \quad (14.4)$$

$$\max \left\{ \tilde{\delta}_i^{\min}, \tilde{\delta}_i(t - \Delta t) + \tilde{\delta}_i^{\min} \Delta t \right\} \leq \tilde{\delta}_i(t) \quad i \in I_{FS} \quad (14.5)$$

$$\tilde{\delta}_i(t) \leq \min \left\{ \tilde{\delta}_i^{\max}, \tilde{\delta}_i(t - \Delta t) + \tilde{\delta}_i^{\max} \Delta t \right\} \quad i \in I_{FS} \quad (14.6)$$

with

$$\tilde{\delta}_{i_j} = 0 \quad \text{if } i_j \in I_{FF}, j \in \{p, q, r, ths\} \quad (15.1)$$

and

$$\tilde{\delta}_{i_j} = \bar{\delta}_{i_j} \quad \text{if } i_j \in I_{FP}, j \in \{p, q, r, ths\} \quad (15.2)$$

where  $I_{\bar{F}}$  is the set of fully operational actuators,  $I_{FL}$ ,  $I_{FS}$  are respectively the set of actuators whose angular positions, angular speed are subject to addition limitations,  $I_{FP}$  is the set of actuators which are stuck at a known angular position,  $I_{FF}$  is the set of actuators which are not subject to a torque from their servo-control and with a zero deflection.  $w_L$ ,  $w_M$  and  $w_N$  are positive weighting parameters which, in the case of a roll maneuver, are such as:

$$w_L \gg w_M \quad \text{and} \quad w_L \gg w_N \quad (16)$$

The above mathematical programming problem can be solved using standard programming techniques and making use as a start of the previous value of the deflections of the actuators. Then in a few iterations the solution of this small size linear quadratic problem should be obtained. Another approach much more direct and systematic considers the use of neural network to solve on-line the above class of problems.

## V. Neural Networks for Online Solution

The basic idea for solving an optimization problem using a tailored neural network is to make sure that the neural network will converge asymptotically and that the equilibrium point of the neural network will correspond to the optimal solution of the optimization problem. In 1986, Tank and Hopfield introduced a linear programming neural network solver realized with an analog circuit which appeared to be well suited for applications that require on-line resolution<sup>8</sup>. After that, many neural network models for solving linear and quadratic programming problems have been proposed in the literature. For a review see [9,10].

According to the relationship between the states of the neural network and the values of primal and dual decision variables, it is possible to divide the existing recurrent neural network for solving linear and quadratic programming problems into three classes: primal neural network, primal-dual neural network, and dual neural network. In the present case, the programming problem is subject to equality and inequality constraints as well as bounding limits. Adopting a primal-dual neural network it is necessary to add many slack variables and the size of network becomes larger. This primal-dual neural network should be globally convergent. Its convergence speed can be adjusted by choosing an adequate value for its learning parameter<sup>11</sup>.

Here a general linear-quadratic programming problem is considered as follows:

$$\min f(\underline{\delta}) = \frac{1}{2} \underline{\delta}^T Q \underline{\delta} + \underline{c}^T \underline{\delta} \quad (17)$$

$$\text{s.t. } h(\underline{\delta}) = J \underline{\delta} - \underline{d} = \underline{0} \quad (18.1)$$

$$g(\underline{\delta}) = A \underline{\delta} - \underline{b} \leq \underline{0} \quad (18.2)$$

$$\underline{\xi}^- \leq \underline{\delta} \leq \underline{\xi}^+ \quad (18.3)$$

where  $\underline{\delta}$  is the actuator deflections vector and where matrix  $Q$  is assumed symmetric positive semi-definite such that quadratic programming and linear programming can both to be handled by Eq. (17) and Eq. (18).

Once the set of constraints Eq. (18.1), Eq. (18.2), Eq. (18.3) is feasible, at least one optimal solution  $\underline{\delta}^*$  will meet the Karush-Kuhn-Tucker optimality conditions (KKT)<sup>12</sup>. Then Eq. (17) and Eq. (18) can be turned equivalent to the following set of linear variational inequalities:

$$(\underline{y} - \underline{y}^*)^T (H \underline{y}^* + \underline{p}) \geq \mathbf{0} \quad \forall \underline{y} \in \Omega \quad (19)$$

with the primal-dual variables  $\underline{y} = [\underline{\delta}^T \quad \underline{u}^T \quad \underline{v}^T]^T$ . Then the problem is to find a solution vector  $\underline{y}^*$ . Its feasible region  $\Omega$  and its lower/upper limits are given by:

$$\Omega := \left\{ \underline{y} \mid \underline{\xi}^- \leq \underline{y} \leq \underline{\xi}^+ \right\}, \quad \underline{\xi}^- = [\underline{\xi}^- \quad -\underline{\omega}^+ \quad \underline{0}]^T, \quad \underline{\xi}^+ = [\underline{\xi}^+ \quad \underline{\omega}^+ \quad \underline{\omega}^+]^T \quad (20)$$

Here  $\underline{\omega}^+$  has an appropriate dimension and each of its entries is sufficiently large to replace  $+\infty$  numerically. The coefficients are defined as:

$$\underline{p} = [\underline{c}^T \quad -\underline{d}^T \quad \underline{b}^T]^T, \quad H = \begin{bmatrix} Q & -J^T & A^T \\ J & \underline{0} & \underline{0} \\ -A & \underline{0} & \underline{0} \end{bmatrix} \quad (21)$$

Then the neural network model which solves Eq. (17) and Eq. (18) is given by:

$$\frac{dy}{dt} = \lambda(E + H^T) \left\{ P_{\Omega} \left( \underline{y} - (H\underline{y} + \underline{\rho}) \right) - \underline{y} \right\} \quad (22)$$

where  $\lambda$  is a positive learning parameter which can be used to adjust the convergence speed of the network,  $E$  is an identity matrix,  $P_{\Omega}[\cdot]$  is a piecewise-linear function defined as

$$P_{\Omega}[y_i] = \begin{cases} \zeta_i^-, & \text{if } x_i \leq \zeta_i^- \\ \zeta_i^+, & \text{if } x_i \geq \zeta_i^+ \\ y_i, & \text{otherwise} \end{cases} \quad (23)$$

## VI. Numerical Example

This example is taken from [13], which is concerned with the control of the unstable lateral/directional dynamics of the X-33 vehicle at critical conditions during the entry flight. By assuming that a linear relationship exists between the pseudo-control vector  $\underline{y}$  and the actual actuators deflections, [13] formulates the problem as:

$$\underline{y} = B\underline{\delta} \quad (24)$$

where  $\underline{y} = [p \ q \ r]^T$ ,  $\underline{\delta} = [\delta_{revi}, \delta_{levo}, \delta_{rbf}, \delta_{lbf}, \delta_{rvr}, \delta_{lvr}, \delta_{revo}, \delta_{levo}]^T$  with  $\delta_{revi}, \delta_{levo}$  = right and left inboard elevons;  $\delta_{rbf}, \delta_{lbf}$  = right and left body flaps;  $\delta_{rvr}, \delta_{lvr}$  = right and left rudders; and  $\delta_{revo}, \delta_{levo}$  = right and left outboard elevons and  $B$  is a  $3 \times 8$  real valued matrix.

Now to satisfy Eq. (24) as well as control surfaces limits such as  $\delta_i^{\min} \leq \delta_i \leq \delta_i^{\max}$   $i \in \{revi, levo, rbf, lbf, rvr, lvr, revo, levo\}$ , we introduce a quadratic programming with the form Eq. (17) and Eq. (18) by taking Eq. (24) as an equality constraint, and we get the effective quadratic problem:

$$f(\underline{\delta}) = (\underline{y} - B\underline{\delta})^T \Lambda (\underline{y} - B\underline{\delta}) \quad (25)$$

where  $\Lambda$  is a diagonal matrix with diagonal elements  $w_p, w_q, w_r$ .

Here we present a simulation scenario of the operation of this neural network where the controls are such as  $p = 10 \text{deg/s}$ ,  $q = r = 0$ , all actuators are fault free except right inboard elevon stuck at zero at 2.1s. The time step adopted for time discretization is  $\Delta t = 0.05 \text{s}$  (for sampling the output of neural network to control signal). Neural network parameters have been chosen such as  $10^{10}$  replace  $+\infty$  numerically in Eq. (19),  $\lambda = 10^6$ ,  $w_p = 10$ ,  $w_q = 1$ ,  $w_r = 1$ .

From Fig. 2, we can see that the neural network converges after about 1ms. The time evolution of different actuators position and controlled output are displayed by Fig. 3 and Fig. 4 respectively. There a star symbol is used to denote the failure instant.

It appears that the neural network solution can handle the failure situation satisfactory even if many factors such as the dynamics of the actuators, actuators deflection rates constraints and dynamic inversion controller time lag where not considered.

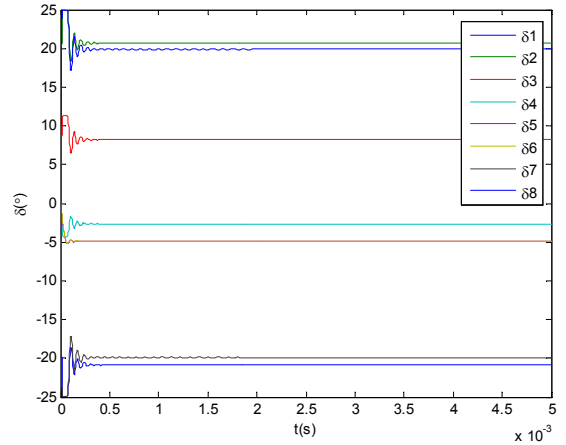
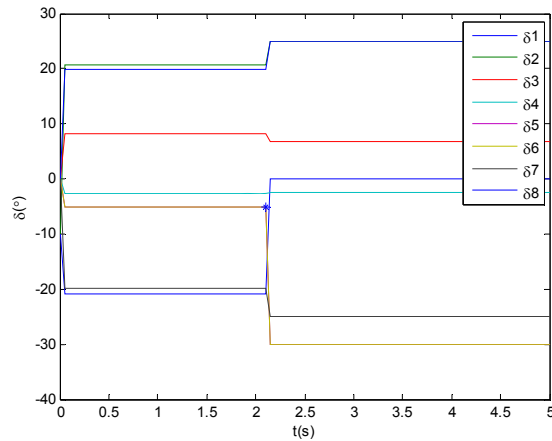
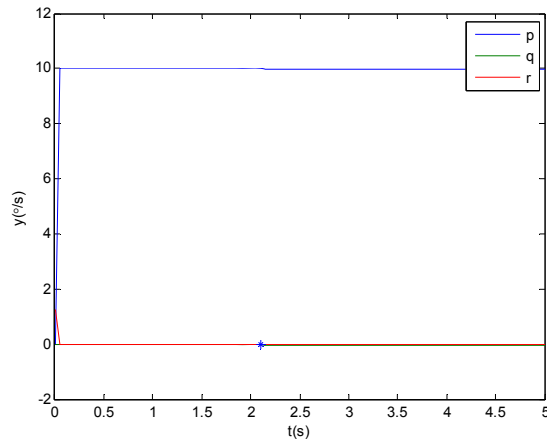


Figure 2. Convergent response of neural network



**Figure 3. Time evolution of actuators deflections**



**Figure 4. Time evolution of controlled output**

## VII. Conclusion

In this paper, a new approach to manage the control surfaces of an aircraft under an actuator failure scenario has been developed. The main objective is to maintain the possibility to perform standard maneuvers with the remaining fault free actuators while limiting the structural strain (maximum wing bending and torsion moments) of the aircraft.

Once the necessary aerodynamics forces and moments to perform the manoeuvre have been computed by inversion of the flight dynamics, the contributions of each remaining actuator to the aerodynamic forces and moments can be determined on-line by a neural network solving a linear quadratic optimization problem. A numerical example considering the use of a neural network to solve on line the actuator reallocation problem has been displayed to illustrate the feasibility of the proposed approach for on-line management of the partially failed control channels.

## References

- <sup>1</sup>España, M. and Gilyard, G., "Aircraft Adaptive Performance Optimization Using Redundant Control Effectors," *Automatic Control in Aerospace*, Palo Alto, CA USA, 1994, pp. 12-16.
- <sup>2</sup>Peterson J.A.M. and Bodson, M., "Interior-point Algorithms for Control Allocation," *Journal of guidance, control, and dynamics*, Vol. 28, No. 3, 2005, pp. 471-480.
- <sup>3</sup>Gaulocher, S. L., Cumer, C. and Alazard, D., "Aircraft load alleviation during maneuvers using optimal control surface combinations," *Journal of Guidance Control and Dynamics*, Vol. 30, No. 2, 2007, pp. 591-600.



- <sup>4</sup>Burken, J. J., Lu, P., Wu, Z. and Bahm, C., “Two Reconfiguration Flight-Control Design Methods: Robust Servomechanism and Control Allocation,” *Journal of Guidance, Control and Dynamics*, Vol. 24, No. 3, 2001, pp. 482-494.
- <sup>5</sup>Etkin, B. and Reid, L. D., *Dynamics of flight: stability and control*, 3 edition, Wiley, 1995.
- <sup>6</sup>Roux, E. , “Modèle de Masse Voilure: Avions de transport civil,” Ph.D. Dissertation, SupAéro-ONERA, Toulouse, France, 2006.
- <sup>7</sup>Andrei,G., “Contribution à la commande tolérante aux pannes pour la conduite du vol,” Ph.D. Dissertation, INSA, Toulouse, France, 2010.
- <sup>8</sup>Tank, D. W. and Hopfield, J. J., “Simple ‘Neural’ Optimization Networks: An A/D Converter, Signal Decision Circuit, and a Linear Programming Circuit,” *IEEE Transactions on Circuits and Systems*, Vol. 33, No. 5, 1986, pp. 533–541.
- <sup>9</sup>Xia, Y. and Wang, J., “Recurrent Neural Networks for Optimization: the State of the Art,” *Recurrent neural networks: design and applications*, edited by Larry Medsker and Lakhmi C. Jain, CRC Press, 2001.
- <sup>10</sup>Kamel, M. S. and Xia, Y., “Cooperative recurrent modular neural networks for constrained optimization: a survey of models and applications,” *Cognitive Neurodynamics*, Vol. 3, No. 1, 2009, pp. 47-81.
- <sup>11</sup>ZhangY., “On the LVI-based Primal-Dual Neural Network for Solving Online Linear and Quadratic Programming Problems,” *American Control Conference*, Portland, OR, USA, 2005, pp. 1351-1356.
- <sup>12</sup>Luenberger, D. G. and Ye, Y., *Linear and Nonlinear Programming*. Springer, 2008.
- <sup>13</sup>Demenkov, M., “*Geometric Algorithms for Input Constrained Systems with Application to Flight Control*,” Ph.D. Dissertation, De Montfort University, UK, 2007.