FAULT - TOLERANT PROCEDURES FOR AIR DATA ELABORATION

Alberto Calia, Eugenio Denti, Roberto Galatolo, Francesco Schettini
University of Pisa – Department of Aerospace Engineering, Pisa

Keywords: *air data, neural networks, failure management*

Abstract

In a modern Full-Authority Fly-By-Wire Flight Control System (FBW/FCS), the air data (static pressure, Mach number, and angles of attack and sideslip) are critical items for the safety of flight operations. As a matter of fact, air data are used for gain scheduling in the control laws and for the envelope protection. Such data have to be obtained by specific computation algorithms on the basis of local airflow measurements performed by redundant air data probes. The algorithms also have to manage the redundancy in order to detect possible failures and to provide consolidated outputs.

This paper describes two different approaches to the development of air data computation algorithms. The first one, widely illustrated in [1], uses polynomial calibration functions tuned on wind tunnel test data relevant to the new jet trainer Aermacchi M-346. The second approach is based on neural networks trained in two ways: using the same wind tunnel data and using preliminary flight test data.

The paper also illustrates the monitoring and voting algorithms developed in order to identify possible probe failures and to provide a voted value for each air data parameter.

Finally, the results of the different approaches are presented by comparisons with the wind tunnel data and preliminary flight test data.

1 Introduction

The air data system consists of all the elements which allow the static pressure (P_{sa}), the Mach number (M_∞) and the angles of attack (α) and sideslip (β) to be evaluated on the basis of local airflow measurements provided by external air data probes. Such evaluation is performed by dedicated algorithms, implemented either in the Flight Control Computers (FCCs) or in specific processing units. In modern FBW/FCS the need to satisfy the safety requirements [2], [3] imposes an adequate redundancy of the components, as well as the definition of robust logics for the failure management.

The air data system studied in this paper refers to the architecture used in the new jet trainer Aermacchi M-346, based on a pseudo-quadruplex redundancy, which employs four self-aligning air data probes named Integrated Multi Function Probe (IMFP) [4] symmetrically installed on the fuselage, two starboard and two port (Fig. 1).

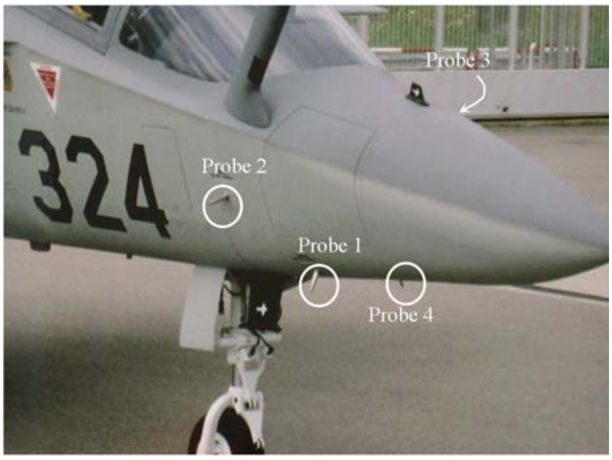

Fig. 1. Probe installation on Aermacchi M-346.

Each probe provides [1] three outputs: the local flow angle λ_i measured by a rotary transducer (where subscript i= 1,…,4 refers to the probe number); the frontal pressure $P_{front\ i}$ provided by a frontal slot aligned with the local flow direction and the slot pressure $P_{slot\ i}$. The last is obtained as the average of the pressures measured by two slots at 90° from the local flow direction.

The air data algorithms have to solve the problem illustrated in Fig. 2. The air data parameters must be determined on the basis of the twelve signals provided by the four probes (four local flow angles and eight local pressures), assuring a correct redundancy management by identifying possible failures and by providing an adequate system reconfiguration. In addition, the algorithms have to manage situations in which one or more probes do not provide reliable measurements since they are in the wake of the fuselage. Finally, the algorithms have to take into account (Fig. 2) both aircraft manoeuvres and configuration effects (landing gear extraction, position of the flaps, etc).

In the paper two different approaches to the air data computation are presented: one based on polynomial calibration functions and another one based on neural networks. The latter approach is by now limited to the determination of static pressure and Mach number. The monitoring and voting algorithms, that are the same for the two approaches, are also presented. Finally, the results of the different approaches are presented by comparisons with wind tunnel data and preliminary flight test data.

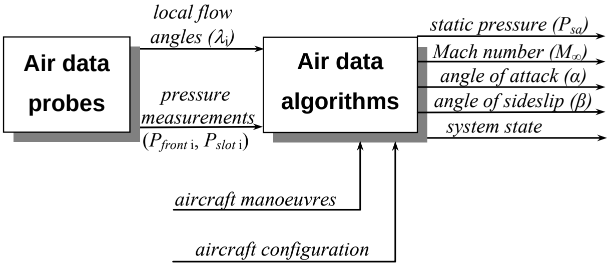

Fig. 2. Inputs and outputs of the air data algorithms.

2 Air data computation using polynomial function

The air data procedure is characterized by two sequential processes: computation of angles and computation of pressures [1]. The first process (Fig. 3) computes six values of α and six values of β by using the four local flow angles measured by the IMFP probes.

With reference to a condition of rectilinear motion and to a fixed aircraft configuration, the i-th local flow angle is a function of the actual values of the α and β angles and of the Mach number. Considering two generic probes (i, j), it is then possible to define the following functions:

$$\lambda_i = f_i(\alpha, \beta, M_\infty) \tag{1}$$

$$\lambda_j = f_j(\alpha, \beta, M_\infty) \tag{2}$$

By using the consolidated Mach number $(\overline{M})$ provided by the procedure at the previous time step, the system of eq.s (1) and (2) allows the calculation of one estimate of the angles of attack and sideslip. The six possible couples (λ_i, λ_j) allow six different couples (α_{ij}, β_{ij}) to be estimated. The six values of angles of attack and sideslip are then forwarded to the monitoring and voting algorithms which identify possible failures and provide a consolidated value ($\overline{\alpha}$, $\overline{\beta}$) for both the parameters.

The second process calculates four couples ($P_{sa\ i}$, $M_{\infty\ i}$) using the local pressures measured by the IMFP probes. The pressure measurements of the i-th probe depend on the actual values of the α and β angles, of the Mach number and of the static pressure:

$$P_{front\ i} = P_{sa}\left[1 + (\gamma/2) M_\infty^2 Cp_{front\ i}(\alpha, \beta, M_\infty)\right] \tag{3}$$

$$P_{slot\ i} = P_{sa}\left[1 + (\gamma/2) M_\infty^2 Cp_{slot\ i}(\alpha, \beta, M_\infty)\right] \tag{4}$$

where γ is the ratio of specific heats of air and $Cp_{front\ i}$ and $Cp_{slot\ i}$ are the frontal and slot pressure coefficient of the i–th probe. If such coefficients are determined by using $\overline{\alpha}$ and $\overline{\beta}$ from the first process and $\overline{M}$ from the previous time step of the procedure, the i-th value P_{sa} and the i-th value M_∞ are determined by solving the system the system of eq.s (3) and (4).

The f_i, $Cp_{front\ i}$ and $Cp_{slot\ i}$ functions that appear in eq.s (1), (2), (3) and (4) have been determined in the form of look-up-tables by means of wind tunnel tests. The storing of such look-up-tables in the FCCs memory is a challenging problem due to the large amount of data associated to them. To overcome this problem, the look-up-tables have been approximated by means of third or fourth degree polynomial functions whose coefficients have been determined by a least squares technique.

Before the voting and monitoring phase, Mach number and static pressure are corrected to compensate for surface deflection effects, such as leading edge flap (δ_{LE}) and trailing edge flap (δ_{TE}), in order to consider the different aircraft configurations.

Concerning the maneuver effects, Fig. 3 shows two corrections: the first one directly acts on the local flow angles measured by the probes (roll rate P effects), the second one evaluates the angles of attack and sideslip $(\bar{\alpha}, \bar{\beta})_{C.G.}$ at the aircraft's centre of gravity (pitch rate Q and yaw rate R effects). A detailed discussion about all the computation algorithms can be found in [1].

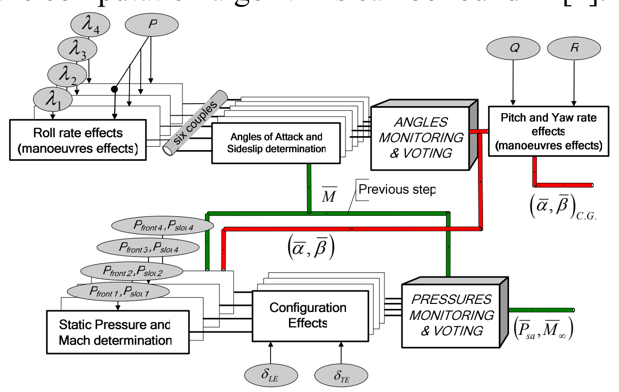

Fig. 3. Flow diagram of the air data procedure.

3 Air data computation using neural networks

The alternative approach [5] based on neural networks herein presented is only relevant to the computation of P_{sa} and M_∞, while the computation of the angles α and β is, at present, the same of the previous section.

The computation of Psa and M∞ is performed by two independent neural networks for each probe (Fig. 4). The two networks have a similar structure. The network relevant to the estimation of M∞ has three input signals and three layers of neurons: an input layer and a hidden layer of 20 neurons each, and an output layer of a single neuron, as shown in Fig. 4. Such an architecture needs 521 parameters to be stored in FCCs, for each probe. The output neuron provides the estimate of the Mach number, while the three input signals are: the angles α and β given by the angles computation, and the ratio $P_{front\ \mathrm{i}}$ / $P_{slot\ \mathrm{i}}$ of the i-th probe. This pressure ratio provides important information because it highly depends on the Mach number and less on the angles of attack and sideslip. For example, the pressure ratio related to probe 2, is plotted in Fig. 5 as a function of both α and β for various Mach number values.

The network relevant to the estimation of P_{sa} has the same architecture, except for the

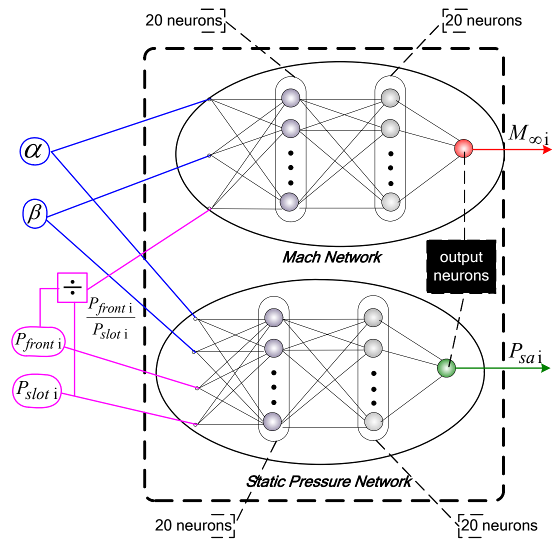

Fig. 4. Mach and static pressure network (i-th probe).

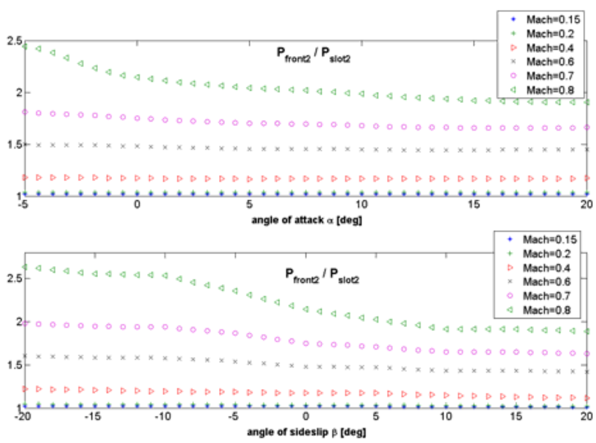

Fig. 5. Pressure ratio vs α, β and *Mach* (probe 2).

input data that are, in this case: the angles α and β given by the angles computation, and the pressures $P_{front\ \mathrm{i}}$ and $P_{slot\ \mathrm{i}}$ measured by the i-th probe. The static pressure network needs 541 parameters to be stored in FCCs for each probe.

The eight networks (two for each probe) need a total of (521+541) x 4 = 4248 parameters to be stored in the FCCs. This number of parameters is to be compared with the about 1200 coefficients needed for the computation of pressures by using the polynomial functions [1].

The neural networks were trained in two ways: by using wind tunnel test data relevant to the new jet trainer Aermacchi M-346 and by using preliminary flight test data.

The training based on flight test data was carried out in order to verify the ability of the neural networks to be tuned on actual flight data, and to determine the accuracy that can be achieved.

The training based on wind tunnel data was carried out in order to compare the performance of the neural networks with the performance of the method based on polynomial functions which, at present, has not yet been tuned on flight test data.

The results of such analyses are reported in § 5.

4 Air data failure management

The algorithms for failure management, are the same for the both the polynomial functions and the neural network approaches presented in § 2 and § 3.

4.1 Failure detection for angles computation

As mentioned in § 2, the procedure estimates six different couples (α_{ij}, β_{ij}) on the basis of the six possible couples of local flow angles (λ_i, λ_j) measured by the four probes.

It is worst noting that a failure in the measurement of one local flow angle will affect three couples (α_{ij}, β_{ij}), so it is not possible to adopt a standard monitoring algorithm to identify such a failure.

To solve the problem of detecting the first failure, the six (α_{ij}, β_{ij}) estimates are grouped into four groups (Tab. 1). The identification number ID of the generic group refers to the probe whose local flow angle measurement is not used in the evaluation of the α_{ij} and β_{ij} included in the group itself. For example, the local flow angle λ_1 does not appear in "Group 1" (see Tab. 1) that contains the estimates of the angles of attack and sideslip related to the couples (λ_2, λ_3), (λ_2, λ_4), and (λ_3, λ_4).

Two standard "cross-channel" monitoring algorithms are then applied in parallel to each group: one acts on the values of the angle of attack, while the other one operates on the values of the angle of sideslip.

Group ID	angle of attack estimates	angle of sideslip estimates	couples of local flow angles
1	α_{23}, α_{24}, α_{34}	β_{23}, β_{24}, β_{34}	$(\lambda_2,\lambda_3)(\lambda_2,\lambda_4)(\lambda_3,\lambda_4)$
2	α_{13}, α_{14}, α_{34}	β_{13}, β_{14}, β_{34}	$(\lambda_1,\lambda_3)(\lambda_1,\lambda_4)(\lambda_3,\lambda_4)$
3	α_{12}, α_{14}, α_{24}	β_{12}, β_{14}, β_{24}	$(\lambda_1,\lambda_2)(\lambda_1,\lambda_4)(\lambda_2,\lambda_4)$
4	α_{12}, α_{13}, α_{23}	β_{12}, β_{13}, β_{23}	$(\lambda_1,\lambda_2)(\lambda_1,\lambda_3)(\lambda_2,\lambda_3)$

Tab. 1. Groups and associated local flow angles

For example, concerning the angle of attack (Fig. 6a), the algorithm orders the three values α_{ij} of each group and it verifies if the differences between each extreme value and the middle value are below a preset monitoring threshold. The crossover of the threshold points out the occurrence of an angle failure. Such a

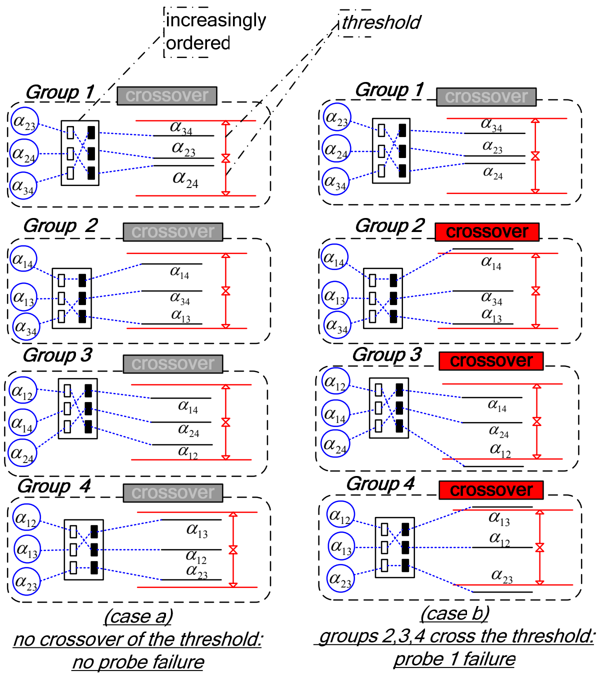

Fig. 6. Monitoring algorithms for the identification of the first angle failure.

crossover should occur in all groups, except in the one not correlated with the failed probe. Fig.

6b shows the case in which a failure occurs in the measurement of the local flow angle of probe 1. Some non-standard situations could occur, in which only one or two groups cross the threshold. This should happen during the transient after a failure. If such a situation continues for a long period of time, due to signal noise, to probe dynamics or to the value chosen for the threshold, some further tuning of the algorithms parameters can be necessary.

Once the first failure occurred, only three (α_{ij}, β_{ij}) estimates are still available (see Tab. 2). In this case, the same standard "cross-channel" monitoring algorithms are applied to such estimates in order to detect a possible further failure. However, in this case it is not possible to recognize which probe led to the second failure. As a consequence, a consolidated solution ($\bar{\alpha}, \bar{\beta}$) cannot be determined and the process for the computation of angles of attack and sideslip is no longer operative.

available couples	probe 1 failure	probe 2 failure	probe 3 failure	probe 4 failure
(λ_1,λ_2)			X	X
(λ_1,λ_3)		X		X
(λ_1,λ_4)		X	X	
(λ_2,λ_3)	X			X
(λ_2,λ_4)	X		X	
(λ_3,λ_4)	X	X		

Tab. 2. Remaining couples when one probe is failing.

4.2 Failure detection for pressures computation

Concerning the computation of P_{sa} and M_∞, the measurements of the local pressures of each probe allow one couple ($P_{sa\ i}$, $M_{\infty\ i}$) to be estimated.

The procedure performs two standard monitoring algorithms in parallel: one acts on the four values of the static pressure, while the other one operates on the four values of the Mach number. A failure in the measurement of a local pressure is latched if the associated $P_{sa\ i}$ and $M_{\infty\ i}$ values do not pass at least one of the two controls.

The monitoring algorithms calculate a reference value for the Mach number (*Mach_m*) and a reference value for the static pressure (*Psa_m*) as the average of the relative two middle estimates [6]. The algorithms evaluate then the differences between the four estimates of Mach (or P_{sa}) and *Mach_m* (or *Psa_m*) in order to verify if one of them is greater than an opportune threshold value *th_Mach* (or *th_Psa*).

Once the first failure occurred, the procedure sets *Mach_m* (or *Psa_m*) equal to the middle of the three remaining values of Mach (or P_{sa}). The monitoring is still possible after two pressure failures, so the system is two-fail operative with respect to the pressure measurements of the probes. In this case the procedure verifies if the difference between the two remaining values is smaller than the preset threshold, otherwise the third pressure failure is declared, although it is not possible to know which probe led to such a failure.

It must be pointed out that this monitoring algorithm is based on the assumption that two failures can not occur at the same time. For this reason, if more than one value crosses the threshold at the same time the pressures computation process is immediately declared not-operative. However, the probability of occurring of such an event is considered extremely remote.

It is worth noting that the procedure to determine P_{sa} and M_∞ needs a previous determination of consolidated values of the angles of attack and sideslip ($\bar{\alpha}, \bar{\beta}$). In the case of loss of such information, due to failures on the measurement of the local flow angles, it is still possible to determine P_{sa} and M_∞ if alternative procedures, based for example on measurements from inertial sensors, are implemented in order to estimate $\bar{\alpha}$ and $\bar{\beta}$.

4.3 Voting algorithms

The aim of the voting algorithms is to provide a consolidated value for each air data parameter. Such algorithms change depending on the number of the estimates that are available. If the monitoring identifies a failed probe, the voting algorithms do not consider the associated parameters. In the full-operative condition, the computation procedure calculates six values of the angle of attack, six values of the angle of sideslip, four values of the Mach number, and four values of the static pressure. The generic voted value (V_{voted}) is the average of the two middle ones.

$$V_{voted} = (V_{III} + V_{IV})/2 \quad \text{for the six values of } \alpha \text{ and } \beta; \tag{5}$$

$$V_{voted} = (V_{II} + V_{III})/2 \quad \text{for the four values of } P_{sa} \text{ and } M_\infty; \tag{6}$$

where sub-scripts *II*, *III* and *IV* indicate the position order. The first failure on the angle measurements causes the loss of three couples (α_{ij}, β_{ij}) of the six available, while the first pressure failure determines the loss of only one couple ($P_{sa\ i}$, $M_{\infty\ i}$) of the four available.
Once the first failure occurred in the measurement of a local pressure or local flow angle, the voted value is set to be equal to the middle of the three remaining estimates.

The following figure shows the way of functioning of the voting algorithms acting on the four values of Mach (or P_{sa}) in the full-operative condition ($t_0<t<t_1$), in the failure condition ($t>t_5$), and during the failure transition ($t_1<t<t_5$).

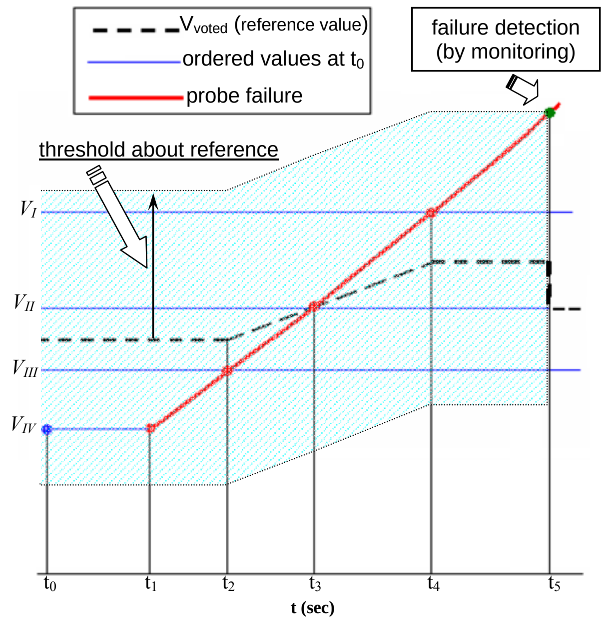

Fig. 7. Voting algorithms (full-operative and first failure).

In the case of a second pressure failure, it is still possible to generate consolidated values for the Mach number and the static pressure because two couples ($P_{sa\ i}$, $M_{\infty\ i}$) are still available. The voted value of the Mach number (or P_{sa}) is the average of the remaining estimates.

Concerning the angles computation process, as mentioned in § 4.1, when the procedure detects a second angle failure, the process is no longer operative.

5 Results

All the air data algorithms have been implemented in the Matlab/Simulink® environment and extensively tested in order to assess their performance. For this purpose, such algorithms have been interfaced with a flight simulator which includes a model of IMFP probes based on the look-up-tables coming from wind-tunnel tests. The algorithms developed given good results in the entire flight envelope. As an example, Fig. 8 reports the absolute value of Mach number and static pressure errors provided by the two approaches (polynomial functions and neural networks) during a generic manoeuvre generated with the flight simulator. The errors refer to the voted values calculated, in both the approaches, by means of the voting algorithms described in § 4.3.

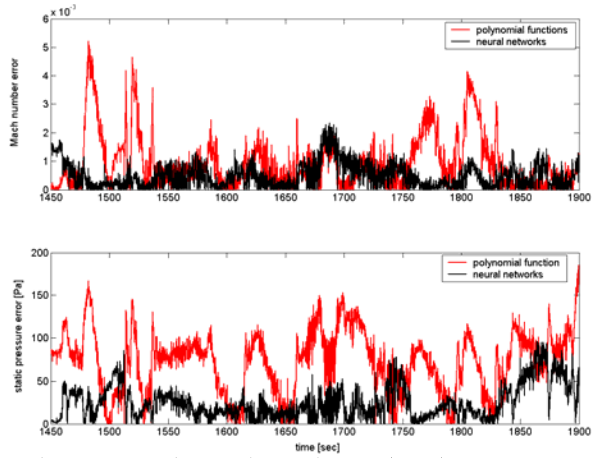

Fig. 8. Errors in Mach number and static pressure

The neural networks have also been trained on preliminary flight test data. The methodology used for training is that described in [7] and it allowed extracting from the original database an opportune subset able to represent the entire domain of the flight tests examined. Fig. 9 plots the absolute value of the static pressure errors of the neural networks trained on both flight data and the wind tunnel database and the same errors of the polynomial functions whose coefficients are also tuned on the wind tunnel database. Such errors have been calculated with respect to the measurements of a nose boom installed on the aircraft during the flight tests, in order to calibrate the air data algorithms. The nose boom provided the total and static pressure, together with the angles of attack and sideslip.

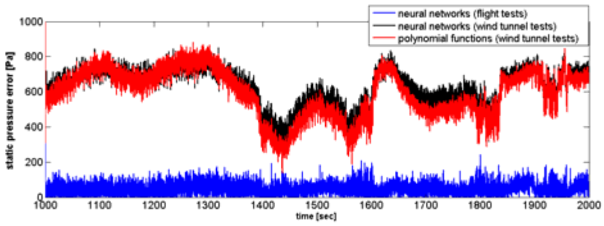

Fig. 9. Static pressure error during a flight test.

Notice that the neural networks calibrated on the flight data reduce the maximum errors to 200 Pa, instead of 800 Pa provided by the two algorithms which refer to the wind tunnel database.

6 Conclusions

A fault-tolerant air data computation procedure has been developed to estimate flight parameters.

As far as the angles of attack and sideslip are concerned, each estimate of such parameters is based on the measurements of local flow angles by two probes. Six estimates are then available, corresponding to the six possible couples of the four probes. For this reason, particular monitoring and voting algorithms were developed to manage the failures on the measurements of the local flow angles. In this case the system is fail-operative / fail-safe.

On the contrary, standard monitoring and voting algorithms have been used to manage the failures in the measurements of the local pressures. As a matter of fact, an estimate of the Mach number and the static pressure can be obtained on the basis of the measurements of local pressures by a single probe. In this case the system is two-fail-operative / fail-safe.

Two different algorithms were developed for the air data computation: one based on polynomial calibration functions and another based on neural networks. The first one was developed in collaboration with the societies *Teleavio* (now *AleniaSIA*) and *Aermacchi* (now *AleniaAermacchi*), within the framework of the program for the development of the Flight Control System of the new jet trainer Aermacchi M346. The second approach, that is by now limited to the determination of static pressure and Mach number, demonstrated to be an interesting alternative.

The accuracy of the two methods are comparable when the neural networks are trained on the same data used to tune the polynomial calibration functions; the neural networks are even a little more accurate. The number of coefficients to be stored in the FCCs is larger for the neural networks (about four times). However, this can be no further true if other inputs are added (i.e. flap deflection, landing gear position, etc.), or if the flight envelope is enlarged with respect to the one considered in the present work; this is because in this case the number of polynomial coefficients can increase very much. In addition, the number of coefficients needed by the neural networks can be reduced by optimizing the architecture, for example by using multiple output networks

On the other hand, the neural networks show dramatic advantages in terms of time to be spent to tune the system on new data, coming either from flight tests or from modifications of aircraft configuration.

References

[1] Cervia F, Denti E, Galatolo R and Schettini F. Air data computation in fly-by-wire flight control systems. *Journal of Aircraft*, Vol. 43, No. 2, pp. 450-455, 2006.

[2] Collinson R P G. Fly-by-wire flight control. *Computing & Control Engineering Journal*, Vol. 10, pp 141-152, 1999.

[3] Mc Lean D. Aircraft flight control system. *The Aeronautical Journal*, Vol. 103, No. 1021, pp 159-165, 1999.

[4] Menzies M A. Integrated air data sensors. *The Aeronautical Journal*, Vol. 105, pp 223-229, 2001.

[5] Calia A and Schettini F. Air data computation using neural networks (in Italian). *Proc. 18th AIDAA National Congress*, Volterra, Italy, 2005.

[6] Houck D. Air data sensor failure detection. *Proc. 17th AIAA/IEEE/SAE Digital Avionics Systems Conference*, Bellew, WA, Vol. 1, 1998.

[7] Rohloff T J, Whitmore S A and Catton I. Air data sensing from surface pressure measurements using a neural network method. *AIAA Journal*, Vol. 36, No. 11, pp 2094-2101, 1998.