JOURNAL OF AIRCRAFT
Vol. 44, No. 3, May–June 2007

Neuro-Fuzzy Techniques for the Air-Data Sensor Calibration

Marco Lando,* Manuela Battipede,† and Piero A. Gili‡
Politecnico di Torino, 10129 Torino, Italy

DOI: 10.2514/1.26030

The paper is concerned with an innovative air-data sensor calibration procedure, carried out through neuro-fuzzy techniques based on adaptive neuro-fuzzy inference system (ANFIS) and co-active neuro-fuzzy inference system (CANFIS) models. In particular, attention is focused on a β sideslip angle virtual sensor, and data used for the calibration are obtained through a series of simulations performed by means of the nonlinear dynamic model in 6 degrees of freedom of a high-performance combat aircraft. Several ANFIS and CANFIS architectures have been developed, tested, and compared with each other. Results of numerical simulations show the remarkable effectiveness of neuro-fuzzy techniques in the sensor calibration.

Nomenclature

C	=	fuzzy rule consequent part
E	=	quadratic output error
e_β	=	absolute signal error
f	=	activation function
G	=	scaling factor
h	=	altitude
k_p, k_d	=	proportional and derivative gain coefficients
M	=	Mach number
$\boldsymbol{P}$	=	covariance matrix
p	=	roll angular rate
r	=	yaw angular rate
$\boldsymbol{W}$	=	weight matrix
$\boldsymbol{x}$	=	input vector of fuzzy system
α	=	angle of attack
$\alpha_{\max}$	=	maximum estimated eigenvalue
$\alpha_{\min}$	=	minimum estimated eigenvalue
β_{ind}	=	indicated angle of sideslip
β_{pred}	=	estimated angle of sideslip
β_{ref}	=	reference angle of sideslip
γ	=	premise parameters (PP)
$\Delta\beta_{\text{pred}}$	=	fuzzy system output
$\boldsymbol{\Theta}$	=	consequent parameter matrix
θ	=	consequent parameters (CP)
μ	=	grade of membership (GOM), firing strength
$\tilde{\mu}$	=	normalized firing strength
$\boldsymbol{\Upsilon}$	=	firing strength vector
$\boldsymbol{\Psi}$	=	gradient descent matrix

I. Introduction

AIR-DATA sensors installed on high-performance modern aircraft must be carefully calibrated to achieve accurate onboard air-data measurements of dynamic pressure, temperature, Mach number, angle of attack α, sideslip angle β, and so forth [1,2]. Accuracy of these measurements should be always pursued for a multitude of tasks, including not only flight safety and aircraft performance evaluation, but also air traffic control, navigation, weapons delivery, and flight control systems. Generally, air-data sensor calibration problems are tackled with conventional regression methods [3–6], which process and combine data measured during several flight tests involving different maneuvers to make the aircraft reach the boundaries of its flight envelope. In this phase, the aircraft is equipped with reference sensors (standard flight test booms, inertial navigation systems, and so on) that provide the calibration system with reference signals, which are unavailable in operative conditions. These air-data measurements are affected by several flight variables that may vary in a very wide range, thus, air-data sensors calibration must be treated as a multidimensional and nonlinear problem. Conventional regression methods need a remarkable amount of appropriate data to evaluate the effects of each variable and the corrections due to the overlap of several concurrent variables. In particular, the classical approach implies that variables are varied one at a time, while the others are maintained at constant values. This approach is feasible in wind-tunnel and steady flight testing but might turn out to be impractical during unsteady flights, when dynamic effects prevails. A thorough calibration in unsteady conditions, in particular, requires a large number of appropriate flight maneuvers with a high expenditure of time, resources, and costs.

In the past, few alternative methods have been proposed to solve this multidimensional and nonlinear problem as well as to reduce risks and costs in flight tests. In particular, Tseng and Lan [7] suggested a technique based on fuzzy logic to improve the calibration procedure of the sideslip angle β sensor on a combat aircraft. The authors showed how the flight test number could be considerably reduced, with the achievement of a satisfactory accuracy at both subsonic and supersonic speed. In fact, calibration errors were less than 0.8 deg in most cases, except at the beginning and ending of maneuvers because of transient effects. However, the results were obtained testing the fuzzy calibrating system through the same four flight test maneuvers used for its design. In this way, some of the limitations of the fuzzy inference systems (FIS) were neglected, such as their poor capability as extrapolators, their lack of adaptability to deal with changing external environment, and their dependence from the human experience.

This paper attempts to overcome the aforementioned drawbacks by proposing an innovative neuro-fuzzy calibration method based on both fuzzy logic and neural learning concepts. In particular, starting from the basic fuzzy frames [8] the author wants to point out the essential contribution of neural networks in the development of a more powerful adaptive artificial intelligence technique. In fact, neural networks are combined with the fuzzy logic to build neuro-fuzzy models that can incorporate human expertise as well as adapt themselves through repeated training. These features make them optimal for the identification and approximation of multidimensional

Received 20 June 2006; revision received 29 November 2006; accepted for publication 19 November 2006.

*Ph.D. Researcher, Aeronautical and Space Department; marco.lando@polito.it.

†Ph.D. Researcher, Aeronautical and Space Department; manuela.battipede@polito.it. Member AIAA.

‡Associate Professor, Aeronautical and Space Department; piero.gili@polito.it. Senior Member AIAA.

and nonlinear dynamic systems. Using data obtained through flight simulations, it will be shown how the neuro-fuzzy sensor calibration might benefit both from the advantages claimed for neural networks and from the linguistic interpretability of fuzzy systems. Furthermore, different expedients are adopted to improve the performance of the neuro-fuzzy systems proposed and based on adaptive neuro-fuzzy inference system (ANFIS) [9] and co-active neuro-fuzzy inference system (CANFIS) [10,11] models. Finally, several ANFIS and CANFIS architectures are developed, analyzed, and compared with a summary of numerical results to emphasize the effectiveness, accuracy, and flexibility of these methods in performing the air-data sensor calibration with a restricted amount of data.

II. Data Sources

Data for the basic fuzzy frame setting of ANFIS and CANFIS models are carried out through a series of flight simulations, in which the aircraft and the real air-data sensor that must be calibrated are replaced, respectively, by a mathematical and an empirical model. In this way, the sensor error accounts just for the effects that are artificially introduced, meaning that dynamics, for example, may not affect sensor behavior, unless the sensor model provides for it. In particular, attention is focused on the virtual air-data sensor of the sideslip angle β, which is supposed to be affected by the Mach number M, the angle of sideslip β, the angle of attack α, the roll angular rate p, and the yaw angular rate r.

A. Aircraft Dynamic Model

Aerodynamic data are acquired through a flight simulator based on the nonlinear 13-state mathematical model in 6 DOF of the F-16 combat aircraft [12,13]. The model is augmented with a multi-input/multi-output (MIMO) autopilot [14–16], which enables to accomplish realistic maneuvers, planned on the basis of the fuzzy criteria that will be discussed in the next section. Flight variables are so stimulated to assume a wide range of combinations within each maneuver and the aircraft is supposed to move far from the trim condition and reach the boundaries of its flight envelope. In this way, one drawback revealed in the first attempt to design a neuro-fuzzy calibration method, as proposed in [17], can be overcome. In fact, in that work, flight simulations were accomplished by means of the F-16 flight simulator without any autopilot, thus, flight variables were recorded only for 20 s at each maneuver, which means that calibration parameters remained too close to trim condition and had little chance of developing in different arrangements. Otherwise, in this paper five different maneuvers of 60 s each are carried out from trim condition $M = 0.594$ and altitude $h = 9144$ m. All these maneuvers are accomplished by using the four control devices (elevators, ailerons, rudder, engine throttle), and the corresponding time histories are recorded in terms of the flight variables M, β, α, p, and r, which can be considered as the most appropriate minimum initial set of input variables for the neuro-fuzzy calibration system. For instance, Fig. 1 shows the noise-filtered time histories of these five flight variables for a rapid double-turn maneuver.

B. Virtual β Air-Data Sensor

During the calibration phase, flight tests are usually carried out with the aircraft equipped with the β sensor (AOS transmitter) to be calibrated, for the indicated angle of sideslip β_{ind}, and with a standard flight test boom and an inertial navigation unit (INU), for the reference angle of sideslip β_{ref}. By using the flight simulator we assume that the sideslip angle β coming out from the simulation is the β_{ref} signal, whereas the indicated angle of sideslip β_{ind} is unknown. For this reason, a β sensor has been modeled in the simulator code to provide the β_{ind} signal. If β_{ind} is the angle of sideslip measured by the sensor and β_{ref} is the angle measured by the reference instrument, the relationship between β_{ref} and β_{ind} can be written as

$$\beta_{\text{ref}} = \beta_{\text{ind}} \pm \Delta\beta \tag{1}$$

where $\Delta\beta$ is the complete correction including *random* errors and

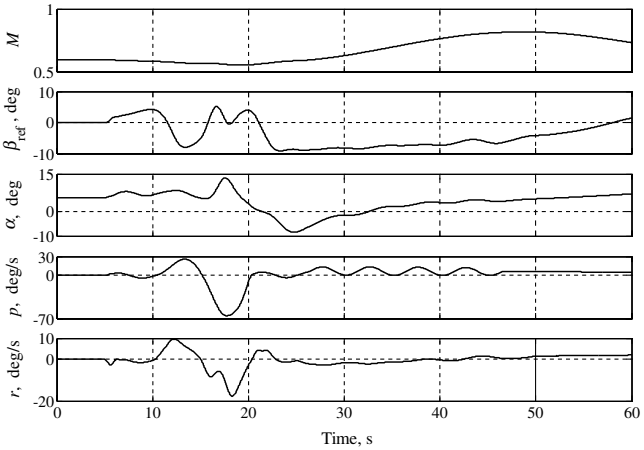

Fig. 1 Time histories for a rapid double-turn maneuver.

bias errors, which define, respectively, precision and accuracy [18]. Random errors include measure repeatability and noise, which can be filtered during real acquisitions. Bias errors comprehend different sources, such as the instrument resolution and precalibration uncertainties that, together with measure repeatability, can be evaluated for the whole measurement chain, but cannot be corrected. Bias includes also the most important source of error, that is, the sensor-aircraft coupling effects due to the kind of sensor location on aircraft and to the flight condition. In this work, the empirical model of β sensor accounts just for these coupling effects through the correction $\Delta\beta_{\text{ref}}$, which can be expressed by a function of the M, β, α, p, and r flight parameters as follows:

$$\Delta\beta_{\text{ref}} = f(M, \beta, \alpha, p, r) \tag{2}$$

On the basis of flight-test time histories, obtained from a previous study on a generic fighter aircraft in different flight conditions, the trend of the function $\Delta\beta_{\text{ref}}$ has been traced and used for the β_{ind} signal evaluation. In particular, $\Delta\beta_{\text{ref}}$ is evaluated as the superimposition of the effects of all the considered flight variables:

$$\Delta\beta_{\text{ref}} = \Delta\beta_M + \Delta\beta_\beta + \Delta\beta_\alpha + \Delta\beta_p + \Delta\beta_r \tag{3}$$

therefore, the indicated sideslip angle will be

$$\beta_{\text{ind}} = \beta_{\text{ref}} - \Delta\beta_{\text{ref}} \tag{4}$$

In this phase, the $\Delta\beta_{\text{ref}}$ correction must be considered just as a hypothetical correction to obtain the β_{ind} value from β_{ref}, because the task of the fuzzy calibration method consists of mapping the multidimensional target function β_{ref}, whatever it is. The fictitious β_{ind} signal becomes one of the inputs of the fuzzy system, together with M, α, p, and r, whereas the output is represented by the correction $\Delta\beta_{\text{pred}}$ needed to evaluate the predicted value β_{pred} of the target function β_{ref}:

$$\beta_{\text{pred}} = \beta_{\text{ind}} + \Delta\beta_{\text{pred}} \tag{5}$$

Finally, β_{ref} is used as reference value for the absolute error signal generation:

$$e_\beta = \beta_{\text{ref}} - \beta_{\text{pred}} = \Delta\beta_{\text{ref}} - \Delta\beta_{\text{pred}} \tag{6}$$

III. Basic Fuzzy Inference System (FIS)

The fuzzy frame is based, to a great extent, on the choice of the input variables. Those must be selected, depending on the sensor features and on the aircraft-sensor coupling, as the most representative of the particular aircraft configuration, of the sensor location and of the type of sensor. As shown in Sec. IV, this generic fuzzy frame can then be specialized for the particular aircraft, through the calculation of the system parameters. Once the parameters have been estimated, the resulting calibration system can

be applied only to the considered aircraft-sensor configuration. Nevertheless, when the most influent input variables are selected, the method is still generic and can be successfully employed on a wide range of sensors, which shows similarities with the proposed case-study. Particularly critical or nonconventional aircraft architectures, however, may require the introduction of additional input variables. For instance, some sensors might be affected by aerodynamic lag or by strong engine disturbances; these situations can be addressed by including a proper set of input variables, which can take into account these effects. The most sensible approach consists of starting with an initial minimal set of variables, such as the one of Eq. (2), and increasing the system dimension whenever the final results show lack of accuracy. The possibility of expanding the system, without jeopardizing the existing frame, is one of the advantages of the fuzzy method, which is particularly functional to handle multivariable problems.

Although mathematical concepts behind fuzzy logic are very simple, it is not trivial to define a FIS which reflects the actual functional relationships among the several variables involved in a sensor calibration process. All the possible variable combinations, in fact, should be considered for the fuzzy rules definition or, at least, the *human expert* should provide FIS with precise and detailed information. However, due to the complexity of the problem, relationships among the calibration variables and the sensor error are hardly incorporated in a logic framework by the human expert, who generally finds it very tricky to manage problems in more than three dimensions. For this reason, in the previous phases of the FIS definition, when the fuzzy IF-THEN rules have to be computed, the human experience can be substituted by numerical examples in the form of time histories [19]. Thus, from the acquisition of time-histories of proper maneuvers, a general frame for the FIS can be set up, with appropriate universes of discourse (UD), scaling factors (G), membership functions (MFs) and fuzzy IF-THEN rules [20]. In particular, the assessment and optimization of the FIS structure for this sensor calibration has been carried out through only four of the ad-hoc 60-s maneuvers mentioned in Sec. II.A. In this way, the neuro-fuzzy calibration requirement of operating with a restricted amount of data is surely fulfilled.

A. Fuzzy Sets and Membership Functions

The human expert activity is basically focused on the system fuzzification, in which the universe of discourse, the scaling factor, and the membership functions must be determined for each variable. Initially, the domain intervals of the selected input and output variables M, β_{ind}, α, p, r, $\Delta\beta_{\text{ref}}$ are defined as $[M^-, M^+]$, $[\beta^-_{\text{ind}}, \beta^+_{\text{ind}}]$, $[\alpha^-, \alpha^+]$, $[p^-, p^+]$, $[r^-, r^+]$, and $[\Delta\beta^-, \Delta\beta^+]$, respectively. Assuming that the aforementioned four maneuvers cover the whole UD of each variable, the previous domain intervals represent the ranges where the variables lie, but, however, in real applications the same variables could even lie outside these defined domain intervals. The respective UDs will be $[M^-_{\text{UD}}, M^+_{\text{UD}}]$, $[\beta^-_{\text{indUD}}, \beta^+_{\text{indUD}}]$, $[\alpha^-_{\text{UD}}, \alpha^+_{\text{UD}}]$, $[p^-_{\text{UD}}, p^+_{\text{UD}}]$, $[r^-_{\text{UD}}, r^+_{\text{UD}}]$, and $[\Delta\beta^-_{\text{UD}}, \Delta\beta^+_{\text{UD}}]$. Successively, fuzzy variables are scaled as elements of their UD and expressed as follows:

$$M_F = \frac{M}{G_M}, \qquad \beta_{\text{ind}_F} = \frac{\beta_{\text{ind}}}{G_\beta}, \qquad \alpha_F = \frac{\alpha}{G_\alpha}, \qquad p_F = \frac{p}{G_p}$$
$$r_F = \frac{r}{G_r}, \qquad \Delta\beta_F = \frac{\Delta\beta}{G_{\Delta\beta}} \tag{7}$$

where the subscript F corresponds to the fuzzified variables, whereas the scaling factor G is defined as the ratio between the maximum absolute value of the variable domain interval and the upper bound of its respective UD. The scaling factors G can be chosen as a consequence of the UD extension, which is directly connected to the variable sensitivity. If a variable is expected to be more influent than the others, in fact, improvements can be obtained by expanding the UD range of this variable. The scaling factor is thus reduced as a consequence. Once the relative range of each UD has been optimized, then, increasing or decreasing all the scaling factors to the same extent do not bring further improvements. The UD of each variable is divided into N fuzzy regions, where the parameter N can be different depending on the influence of variables on calibration. The optimal choice of N can be achieved through a trial-and-error method: starting from an initial configuration, N is increased for each variable until the calibration mean error reaches a constant value. This means that the calibration problem has a defined sensitivity towards the fuzzy discreteness, and below this sensitivity the FIS becomes overdimensioned.

Table 1 Significant parameters of the basic FIS

	Max value	UD	MFs	G
M	1	0–2.4	4	0.417
β_{ind}	20 deg	−36–36	5	0.555 deg
α	43 deg	−24–48	5	0.896 deg
p	72 deg/s	−15–15	5	4.80 deg/s
r	24 deg/s	−15–15	5	1.60 deg/s
$\Delta\beta_{\text{ref}}$	13 deg	−45–45	17	0.289 deg

Each fuzzy region is characterized by a simple triangular MF, whose parameters are defined by the human expert. However, due to the large size of the available input-output data set, the MFs shaped by user may contain vague information about the system or not be precise enough and, therefore, have to be optimized manually through the trial-and-error method as well. Finally, the basic fuzzy frame is set up by a combination of 24 triangular MFs for the five input variables and 17 MFs for the output, respectively, four for M, five for β, α, p, r, and 17 for $\Delta\beta$. In particular, the number of output MFs is increased to achieve more accuracy in the error correction during the calibration process. Table 1 summarizes the final parameters of the FIS architecture after the trial-and-error approach. For each variable, the maximum value defines the UD boundary, which is set to a given value depending on the resolution assigned to that variable. Symmetry criterion has been applied in defining UD, except for the Mach number and the angle of attack α, where UD has been narrowed to the boundaries of the actual range of data obtained from simulations. In real applications, UD should be adjusted and widened to cover the whole aircraft flight envelope.

B. Fuzzy IF-THEN Rules

Once the number and shape of MFs are defined, fuzzy IF-THEN rules are extracted from the four mentioned 60-s maneuvers with a sampled time of 0.2 s. In this way, each maneuver gives rise to 300 AND rules, that is, rules in which conditions stated in the *antecedent* (IF) part must be met simultaneously to consider the *consequent* (THEN) part. After several attempts and simulations, the sampled time of 0.2 s was found suitable to capture with accuracy even the fast variations of the flight parameters involved in the calibration process, avoiding gaps in the rule set.

At this point, it is necessary to evaluate the effects of the rule number on performance and choose the most appropriate set of fuzzy rules. Normally, no relationship links the number of collected rules with the number of describing rules. It is highly probable, in fact, that input/output data pairs produce both repeated rules and conflicting rules, that is, rules that have the same IF part but a different THEN part. Although literature does not report examples of known pathologies caused by the number of rules, it was noticed that the more rules are supplied the less the system is able to generalize. This behavior is assimilable to the *overfitting* problem; in fact, increasing the number of rules means increasing system dimensions, namely, the number of internal parameters. Consistent with expectations, the system has shown a different behavior towards the rule elimination: whereas the repeated rule cancellation brings beneficial effects, because system dimensions are reduced without loss of information, the conflicting rule preservation allows the system to be more tolerant towards uncertainties, and thus more capable of generalizing. Hence, according to the rule-checking process mentioned previously, conflicting rules are maintained whereas repeated rules are excluded through a meritocratic process: a degree is assigned to each rule

generated from data pairs and the rule with the maximum degree is chosen to represent the whole group of identical rules. In this way, the final number of rules decreases to a $10 \div 15$ percentage of the total collected rules and a complete set of 156 describing rules is used to set up the basic fuzzy frame. The generic kth fuzzy IF-THEN rule can be expressed in the Mamdani form [21]:

$$\begin{aligned} &\text{If } M \text{ is } M, \quad \beta_{\text{ind}} \text{ is PS}, \quad \alpha \text{ is AZ}, \quad p \text{ is AZ}, \quad \text{and} \quad r \text{ is AZ} \\ &\text{then } \Delta\beta \text{ is NMS} \quad \text{and} \quad \text{the weight is DOR} \end{aligned} \tag{8}$$

where the value DOR is the degree of the kth rule, which is defined as follows:

$$\text{DOR} = \mu_M(M)\mu_{\text{PS}}(\beta_{\text{ind}})\ \mu_{\text{AZ}}(\alpha)\ \mu_{\text{AZ}}(p)\ \mu_{\text{AZ}}(r)\ \mu_{\text{NMS}}(\Delta\beta) \tag{9}$$

The terms $\mu_M(M)$, $\mu_{\text{PS}}(\beta_{\text{ind}})$, $\mu_{\text{AZ}}(\alpha)$, $\mu_{\text{AZ}}(p)$, $\mu_{\text{AZ}}(r)$, and $\mu_{\text{NMS}}(\Delta\beta)$ represent the grades of membership (GOM) of the calibration variables, respectively, in the fuzzy sets M, PS, AZ, AZ, AZ, NMS defined by the fuzzy rule, and in which the variables themselves have maximum degree.

For each fuzzy rule, the firing strength that represents the degree to which the antecedent part of the rule is satisfied, is specified by using the fuzzy operator *T-norm* (min) and is then used to implement the implication process reshaping the corresponding consequent MF. Successively, all these fuzzy rule outputs are combined into a single fuzzy set according to the compositional rule of inference [22], which defines the aggregation process based on the *T-conorm* (max) fuzzy operator. The input of the aggregation process is the list of reshaped output MFs returned by the implication process for each rule. Otherwise, the output, which represents the final result of the fuzzy reasoning process, is the aggregate fuzzy set $\mu_{\Delta\beta}$. The max-min composition of this Mamdani-type FIS can be summarized as follows:

$$\mu_{\Delta\beta} = \max\left\{\min_{k=1}^{N_r}\left(\mu_M^k, \mu_{\beta_{\text{ind}}}^k, \mu_\alpha^k, \mu_p^k, \mu_r^k\right)\right\} \tag{10}$$

where N_r is the total number of fuzzy IF-THEN rules. Finally, the final output fuzzy set $\mu_{\Delta\beta}$ is defuzzified through the centroid method, which returns the calibration crisp correction value $\Delta\beta_{\text{pred}}$ according to the following expression:

$$\Delta\beta_{\text{pred}} = \frac{\int_Z \mu_{\Delta\beta}(z)\, z\, \mathrm{d}z}{\int_Z \mu_{\Delta\beta}(z)\, \mathrm{d}z} \tag{11}$$

IV. Neuro-Fuzzy System Assessment

Because of the complexity of the calibration problem and the large size of available input-output data, the simple FIS frame of Sec. III, principally based on human experience, may contain vague information or may not be precise enough. Consequently, remarkable results, which will be presented in Sec. V, can be achieved only after a long manually performed trial-and-error approach. This weakness can be overcome through the introduction of neuro-fuzzy systems based on the coupling of fuzzy logic with neural networks [23,24]. The neural training algorithms imply a completely different approach to pursue the calibration system optimization: the effects of those parameters of Table 1, which are considered significant for the performance of the basic fuzzy system, are no longer investigated. The fuzzy frame is simply made suitable to interact with the neural training algorithms, which are at the base of the optimization procedure. In these systems, human experience is firstly translated by fuzzy logic into modifiable MFs and fuzzy IF-THEN rules, and then postprocessed by a neural network learning algorithm, which fine-tunes the parameterized MFs and fuzzy rules to obtain better performance. In particular, during the learning process the parameters associated with MFs, named *premise* parameters (PP), and the parameters of the fuzzy rules, named *consequent* parameters (CP), will change to reduce the error measure expressed by Eq. (6). The computation or adjustment of these parameters is facilitated by a gradient vector, which supplies a measure of how well the augmented FIS is modeling the input/output data for a given set of parameters.

A. ANFIS Model

The simplest neuro-fuzzy system used in the β-sensor calibration process is based on an ANFIS model with linear fuzzy rules. To fine-tune the premise and consequent parameters through neural learning algorithms, the derivative of the output error $\Delta\beta_{\text{pred}}$ with respect to each system internal parameter must be supplied. However, *T-norm* and *T-conorm* fuzzy operators, triangular MFs, and defuzzification process make the differential calculus awkward and sometimes impossible. For this reason, the Mamdani-type FIS described in Sec. III is never used to implement the fuzzy core of ANFIS models; generally it is simplified and reparameterized to become a hybrid between a fuzzy system and a neural network. In particular, the MFs of input variables (M, β_{ind}, α, p, r) are expressed by continuous analytical functions, such as Gaussian, bell-shaped, or sigmoidal curves. Whereas the triangular MFs of the consequent part of the fuzzy IF-THEN rules are substituted by a polynomial form in the FIS input variables, that is, a linear combination of M, β_{ind}, α, p, and r. In this way, it is possible to obtain crisp values from the fuzzy reasoning process based on the *product* composition rule, and consequently, the defuzzification stage can be replaced by a simpler weighted average. This new FIS is commonly defined as first-order Takagi–Sugeno model [25] and the related expression of the kth fuzzy IF-THEN rule can be rewritten as follows:

$$\begin{aligned} &\text{If } M \text{ is } M, \quad \beta_{\text{ind}} \text{ is PS}, \quad \alpha \text{ is AZ}, \quad p \text{ is AZ}, \quad \text{and} \quad r \text{ is AZ} \\ &\text{then } \Delta\beta \text{ is } C^k = \{\theta_0\theta_1\theta_2\theta_3\theta_4\theta_5\}^k\{1M\ \beta_{\text{ind}}\ \alpha\ p\ r\}^T \end{aligned} \tag{12}$$

where the consequent parameters θ, suitably gathered into the matrix $\boldsymbol{\Theta}$, express a linear relationship between the ANFIS output $\Delta\beta$ and the five input variables. Therefore, they can be used as further degrees of freedom in the ANFIS optimization process. Finally, the defuzzification expression of Eq. (11) is replaced by the following weighted average:

$$\Delta\beta_{\text{pred}} = \frac{\sum_{k=1}^{N_r}\mu^k C^k}{\sum_{k=1}^{N_r}\mu^k} = \sum_{k=1}^{N_r}\tilde{\mu}^k C^k = \boldsymbol{\Upsilon}_{\text{PP}}^T \cdot \mathbf{C} \tag{13}$$

where N_r is the total number of fuzzy IF-THEN rules and $\tilde{\mu}^k$ is the normalized firing strength of the kth fuzzy rule. As it can be noticed from Eq. (13), the premise parameters, defining the position and shape of the parameterized MFs, affect the $\Delta\beta_{\text{pred}}$ calculus in a nonlinear way through the firing strengths μ^k. Because of this formal difference between premise and consequent parameters, a hybrid neural learning method will be proposed in Sec. IV.C to speed up the optimization process: premise and consequent parameters are updated in turns through two different algorithms. In addition, the optimization process demands also a normalization of the MFs to increase the effectiveness of neural learning on the premise parameters. For this reason, UD has been normalized to keep the domains between -1 and $+1$. The general ANFIS architecture implemented in the β-sensor calibration process follows the functional scheme of Fig. 2.

An important issue in the ANFIS training is how to preserve the physical meaning and other intuitive features of the fuzzy rules, avoiding the occurrence of nonsense in the MF reshaping, which may jeopardize the FIS effectiveness. In particular, it was found out that *Gaussian* and *sigmoidal* MFs could change unexpectedly their shape and position from the initial condition during the neural training. In fact, some MFs were null for a given range of UD, sigmoid slope could change its sign, and Gaussian could slide under their neighbor or became excessively wide or thin. Figure 3, for example, shows the initial MFs of the input variable p and the corresponding MFs after

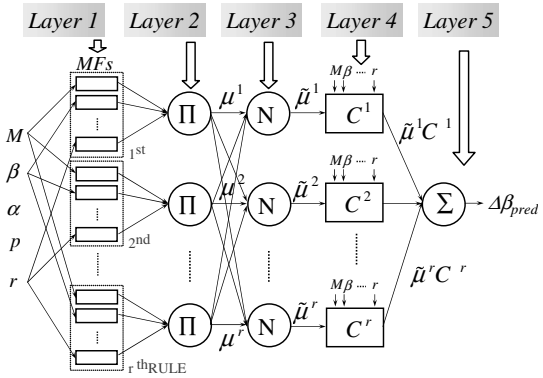

Fig. 2 ANFIS and CANFIS architecture.

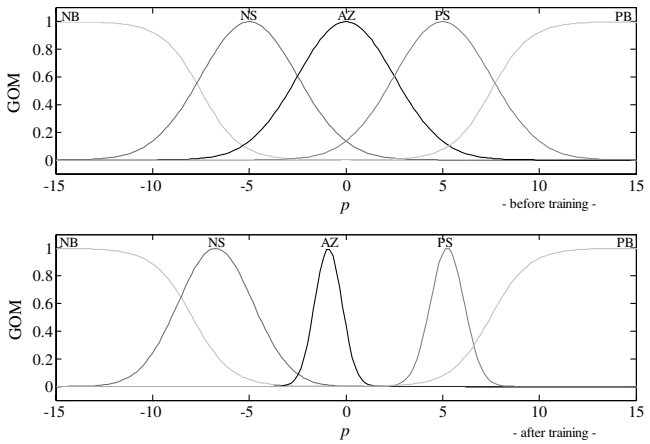

Fig. 3 Gaussian MFs of input variable p before and after training.

the procedure of training: a portion of the UD of p is uncovered, namely, the GOM of all the MFs is null for a given range of p, but this situation is a physical nonsense. In fact, p, as a motion variable, has a continuous domain and the undefinedness of its MFs on a portion of the UD is physically unacceptable.

Unlike Gaussian MFs, other MFs such as the *generalized bell* are more robust, because they are defined with a greater number of internal parameters: this enables to split the information gained through the training without affecting dramatically any of them. The generalized bell function is defined by the equation

$$\mu(x) = \left(1 + \left|\frac{x-c}{a}\right|^{2b}\right)^{-1} \tag{14}$$

where c determines the center of the MF, a is half the width, and b controls the slopes at the crossover points (where the MF value is 0.5). With 3 DOF, the generalized bell MFs seem to be more effective during training than the Gaussian MFs. In fact, as it can be noticed in Fig. 4, the generalized bell MFs relative to the input variable p change smoothly their shape and position during the training phase, avoiding uncovered portions of the normalized UD.

Another expedient to preserve the FIS linguistic interpretability consists of stiffening the learning algorithm with proper constraints, which makes the MFs satisfy the condition of *ε-completeness* [26,27]. This property guarantees that the whole input space UD is covered properly with the condition that at least a MF has grade greater than or equal to ε. For instance, with $\varepsilon = 0.5$ the ε-completeness condition is satisfied if

$$c_i + a_i \geq c_{i+1} - a_{i+1} \tag{15}$$

where (a_i, b_i, c_i) and $(a_{i+1}, b_{i+1}, c_{i+1})$ are the sets of parameters of two adjacent generalized bell MFs. The ε-completeness property can be easily fulfilled throughout the training process by using a constrained gradient descent method [28,29]. In this manner, the

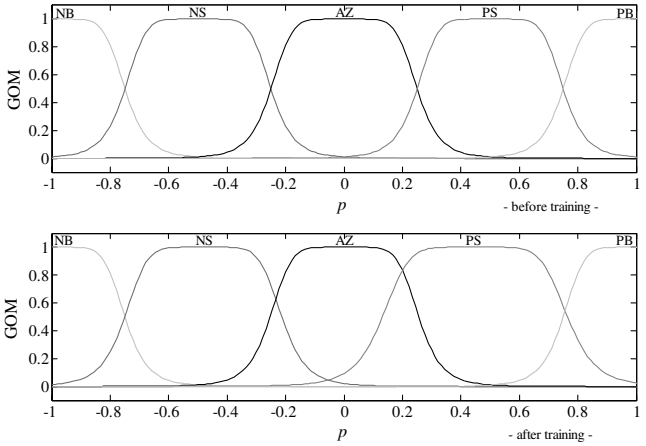

Fig. 4 Generalized Bell MFs of input variable p before and after training.

fuzzy inference system can provide smooth transition and sufficient overlap from one MF to another and, in particular, avoid the occurrences of undesirable physical nonsense. However, this constrained algorithm proved to be too conservative in the evolution of MFs shape and position during training because it could not make premise parameters reach the global minimum but just a local minimum [30].

B. CANFIS Model

In the neuro-fuzzy spectrum [20], ANFIS models stand between a completely understandable FIS and a black box *neural network* (NN). The learning process should improve their input-output mapping precision maintaining the FIS interpretability; however, it often improves precision to the detriment of interpretability, increasing conspicuously the so-called dilemma between interpretability and precision. In addition, when efficient and advanced learning algorithms are applied, a higher input-output mapping precision is achieved but it might lead to meaningless fuzzy rules. Therefore, the dilemma increases and the original interpretability may be lost. Similar observations can be found when there is a high number of fuzzy rules. In fact, in this case the resulting MFs may not lend themselves to good linguistic interpretation and should be carefully set up so that fuzzy rules can be held to meaningful limits.

The first approach to alleviate the *dilemma* is the reduction of fuzzy rules, starting from the conflicting rules defined in Sec. III.B. Conflicting rules, in fact, are useful to define the system uncertainties but deteriorate the FIS interpretability at the same time. Secondly, linear fuzzy rules are replaced by nonlinear and neural fuzzy rules, whose more learning power allows a higher mapping precision with the same computational effort while interpretability is maintained. This extension emphasizes characteristics of a more fused neuro-fuzzy system called CANFIS. The evolution of CANFIS models during the learning phase is compared with the ANFIS behavior in the neuro-fuzzy spectrum of Fig. 5. As already outlined, ANFIS models should follow the vertical dotted route but they often take the diagonal solid route. Otherwise, CANFIS models are more effective during training and follow the vertical solid route. In fact, they work more on the consequent parameters θ, preventing the MFs from varying a lot during the learning phase. In this way, CANFIS models can get a better condition in terms of interpretability and mapping precision than the other models, even if they start from a worse condition in terms of linguistic interpretability due to their neural network structure.

The CANFIS architecture can be illustrated through a scheme similar to the analogous ANFIS model of Fig. 2: layers from 1 to 3 and layer 5 are identical, whereas layer 4 has to be redefined. In particular, for CANFIS models with nonlinear rules, the consequent part C^k of the kth linear fuzzy rule of Eq. (12) is replaced by a nonlinear combination of the consequent parameters $\boldsymbol{\theta}^k$:

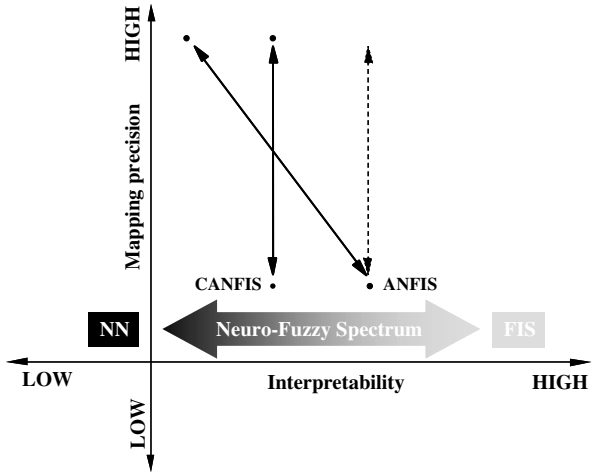

Fig. 5 Evolution of ANFIS and CANFIS models in the neuron-fuzzy spectrum.

$$\text{If } M \text{ is } M, \quad \beta_{\text{ind}} \text{ is PS}, \quad \alpha \text{ is AZ}, \quad p \text{ is AZ}, \quad \text{and} \quad r \text{ is AZ}$$
$$\text{then } \Delta\beta \text{ is } C^k = f^k(\{\theta_0\theta_1\theta_2\theta_3\theta_4\theta_5\}^k\{1M\beta_{\text{ind}}\ \alpha pr\}^T) \tag{16}$$

where f^k is a *bipolar hyperbolic tangent* activation function in the form

$$f^k(x_k) = 1 - 2(e^{2x_k} + 1)^{-1} \tag{17}$$

with $x_k = \{\theta_0\theta_1\theta_2\theta_3\theta_4\theta_5\}^k\{1M\beta_{\text{ind}}\alpha pr\}^T$ representing the input of the kth activation function at the consequent layer. If f^k is the identity function, the consequent part is linear and CANFIS turns into the simpler ANFIS model. Finally, if the consequent part of each fuzzy rule is accomplished by a layered NN, the corresponding CANFIS model will be based on neural rules and layer 4 will be redefined through a network-layered representation. In this manner, the neural consequent part C^k of the kth fuzzy rule is represented by the output of the relative neural network. In particular, using three-layer perceptrons[§] (MLP with one hidden layer) with bipolar hyperbolic tangent activation functions f of Eq. (17) in the hidden layer, the neural consequent part C_k at the ith time step can be expressed as follows:

$$C^k(i) = \boldsymbol{W}_{2_k} \cdot \begin{bmatrix} f[\boldsymbol{W}_{1_k} \cdot \phi(i)] \\ 1 \end{bmatrix} \tag{18}$$

where $\phi(i) = \{1M(i)\beta_{\text{ind}}(i)\alpha(i)p(i)r(i)\}^T$ is the network input and the matrices $\boldsymbol{W}_{1_k}[n_h \times (n_i + 1)]$ and $\boldsymbol{W}_{2_k}[1 \times (n_h + 1)]$ represent, respectively, the input-to-hidden-layer weights and the hidden-to-output-layer weights of the kth network, in which n_i is the dimension of the input vector and n_h is the number of hidden-layer neurons. The *bias* elements are considered by adding 1 to n_i and n_h. Accordingly, the consequent parameters $\boldsymbol{\theta}^k$ of the kth neural rule are the elements of the weight matrices $\boldsymbol{W}_{1_k}$ and $\boldsymbol{W}_{2_k}$. The total matrix $\boldsymbol{\Theta}\{[n_h + n_h \cdot (n_i + 1) + 1] \times N_r\}$ of consequent parameters is defined by the following expression:

$$\boldsymbol{\Theta} = \begin{bmatrix} \boldsymbol{W}_{2_1}(1,1) & \cdots & \boldsymbol{W}_{2_{N_r}}(1,1) \\ \vdots & \ddots & \vdots \\ \boldsymbol{W}_{2_1}(1, n_h + 1) & \cdots & \boldsymbol{W}_{2_{N_r}}(1, n_h + 1) \\ \boldsymbol{W}_{1_1}(1,1) & \cdots & \boldsymbol{W}_{1_{N_r}}(1,1) \\ \vdots & \ddots & \vdots \\ \boldsymbol{W}_{1_1}(n_h, n_i + 1) & \cdots & \boldsymbol{W}_{1_{N_r}}(n_h, n_i + 1) \end{bmatrix} \tag{19}$$

where N_r is the total number of neural fuzzy rules.

C. Hybrid Neural Learning Algorithm

The training phase of both ANFIS and CANFIS models is carried out through an *online* hybrid learning method [31], which can speed up the learning process avoiding substantially the slowness and the tendency of the simple gradient method to become trapped in local minima. Premise and consequent parameters are updated in turns after each data presentation through two different algorithms, respectively, a *gradient descent* algorithm and a *linear recursive least square* algorithm. In the first step, premise parameters are kept fixed while consequent parameters are adjusted through the linear recursive least square method. For each fuzzy rule, the ith time step of this algorithm, applied according to the *constant-trace* technique, can be expressed by the following equations:

$$\boldsymbol{K}^k(i) = \boldsymbol{P}^k(i-1)\boldsymbol{\Psi}^k_{\text{CP}}(i) \cdot [I + K_p\boldsymbol{\Psi}^{kT}_{\text{CP}}(i)\boldsymbol{P}^k(i-1)\boldsymbol{\Psi}^k_{\text{CP}}(i)]^{-1} \tag{20}$$

$$\theta^k(i) = \theta^k(i-1) + \boldsymbol{K}^k(i) \cdot \text{PI}(i) \tag{21}$$

$$\tilde{\boldsymbol{P}}^k(i) = [I - \boldsymbol{K}^k(i)K_p\boldsymbol{\Psi}^{kT}_{\text{CP}}(i)] \cdot \boldsymbol{P}^k(i-1) \tag{22}$$

$$\boldsymbol{P}^k(i) = \frac{\alpha_{\max} - \alpha_{\min}}{\text{tr}[\tilde{\boldsymbol{P}}^k(i)]}\tilde{\boldsymbol{P}}^k(i) + \alpha_{\min}I \tag{23}$$

where $\alpha_{\max}$ and $\alpha_{\min}$ are, respectively, the maximum and minimum estimated eigenvalues of the covariance matrix $\boldsymbol{P}^k(i)$, $\boldsymbol{\theta}^k(i)$ is the kth column of the consequent parameter matrix $\boldsymbol{\Theta}$, and PI(i) is the performance index at the ith time step:

$$\text{PI}(i) = K_p[\Delta\beta_{\text{ref}}(i) - \Delta\beta_{\text{pred}}(i)] + K_d[\Delta\dot{\beta}_{\text{ref}}(i) - \Delta\dot{\beta}_{\text{pred}}(i)] \tag{24}$$

K_p and K_d are, respectively, proportional and derivative gain coefficients [15], whereas $\Delta\beta_{\text{pred}}(i)$ is the system output computed by Eq. (13) at the ith time step. $\boldsymbol{\Psi}_{\text{CP}}(i)$ is the matrix that determines the gradient descent direction with respect to the consequent parameters through a backpropagation procedure:

$$\boldsymbol{\Psi}_{\text{CP}}(i) = \nabla_{\boldsymbol{\Theta}}\Delta\beta_{\text{pred}}(i) = \boldsymbol{\Upsilon}^T_{\text{PP}}(i) \cdot \nabla_{\boldsymbol{\Theta}}\boldsymbol{C}(i) \tag{25}$$

where $\boldsymbol{\Upsilon}^T_{\text{PP}}(i)$ is the vector of the normalized firing strengths $\tilde{\mu}^k$ and $\boldsymbol{C}(i)$ is the output vector of the rule consequent parts. For ANFIS models with linear fuzzy rules, $\boldsymbol{\Psi}_{\text{CP}}(i)$ can be written as

$$\boldsymbol{\Psi}_{\text{CP}}(i) = [\tilde{\mu}^1(i)\boldsymbol{x}(i) \cdots \tilde{\mu}^k(i)\boldsymbol{x}(i) \cdots \tilde{\mu}^{N_r}(i)\boldsymbol{x}(i)]^T \tag{26}$$

with $\boldsymbol{x}(i)$ the input vector $\{1M(i)\beta_{\text{ind}}(i)\alpha(i)p(i)r(i)\}^T$. Otherwise, for CANFIS models the gradient $\nabla_{\boldsymbol{\Theta}}\boldsymbol{C}(i)$ of Eq. (25) is a $N_r \times [n_h + n_h(n_i + 1) + 1]$ matrix, whose kth row will be expressed as

$$\nabla_{\theta^k}C^k(i) = \left[\frac{\partial C^k}{\partial\boldsymbol{\Theta}(1,k)} \cdots \frac{\partial C^k}{\partial\boldsymbol{\Theta}[n_h + n_h(n_i + 1) + 1, k]}\right] \tag{27}$$

where $\boldsymbol{\Theta}$ is the total matrix of consequent parameters defined by Eq. (19).

[§]The three-layer perceptron is the simplest neural network with nonlinear capabilities and associates calculation simplicity with result accuracy. Besides, it has good extrapolation properties, which make it particularly attractive to complement a fuzzy system.

The next step involves the renewing of the premise parameters, gathered into the vector $\boldsymbol{\gamma}$, through a procedure based on the gradient descent method, which is the same used for the neural network backpropagation procedure. This means that the quadratic output error $E(i)$ expressed as

$$E(i) = \frac{1}{2} e_{\Delta\beta}(i)^2 = \frac{1}{2}[\Delta\beta_{\text{ref}}(i) - \Delta\beta_{\text{pred}}(i)]^2 \tag{28}$$

is backpropagated through each layer, up to the MFs:

$$\Psi_{\text{PP}}(i) = \nabla_{\gamma} E(i) = -e_{\Delta\beta}(i)[\nabla_{\mu}\Delta\beta_{\text{pred}}(i)]^T[\nabla_{\gamma}\boldsymbol{\mu}(i)] \tag{29}$$

and then used in a recursive updating formula similar to Eq. (20–23), in which $\boldsymbol{\Psi}^k_{\text{CP}}$ and $\theta^k(i)$ are substituted, respectively, by $\boldsymbol{\Psi}_{\text{PP}}(i)$ and $\boldsymbol{\gamma}$.

Although alternative learning methods have been implemented, such as the resilient propagation algorithm (RPROP) [32,33], the quickprop algorithm (QP) [34] and the Levenber–Marquardt algorithm (LM) [35], the aforementioned hybrid learning algorithm seems to be the best choice to maintain interpretability and train the fuzzy rules without leading to meaningless MFs.

V. Numerical Results

The training phase of ANFIS and CANFIS models is accomplished by means of just one of the four maneuvers used for the FIS optimization in Sec. III. A further maneuver is used to test and evaluate performance of different FIS, ANFIS, and CANFIS configurations in terms of maximum and mean calibration errors e_{β} and standard deviations. Results of this comparison are summarized in Table 2, in which the "*% incr*" represents the percentage of the error increment with respect to the best result. These results can be then compared with the β accuracy level required by the highly maneuverable aircraft flight control systems, for which the angle of sideslip can be used as an important feedback variable to control and/or compensate the lateral-directional stability. For these control systems the angle of sideslip accuracy requirement ranges between 0.5 and 0.75 deg. This threshold must be correlated to the static calibration accuracy, which, for high-precision devices, is about ± 0.15 deg.

Firstly, two Mamdani-type FIS, labeled Mamdani 1 and Mamdani 2, are implemented according to the FIS parameters of Table 1: the first is based on the 156 fuzzy rules defined automatically with the checking process, whereas the second results from the first one through some adjustments and/or exclusions of conflicting rules after a long trial-and-error process. As it can be seen from Table 2, Mamdani 2 model is more effective, reducing the maximum error of about 12% and the mean error of about 9%.

Table 2 Performance of different fuzzy and neuro-fuzzy systems for the β sensor calibration (*% incr* is the percentage of the error increment with respect to the best result)

Fuzzy and NF models	Max error, deg (*% incr*)	Mean error, deg (*% incr*)	STD, deg (*% incr*)
Mamdani 1	0.77231	0.2775	0.3299
	(+43.12)	(+452.79)	(+251.33)
Mamdani 2	0.6831	0.2547	0.2940
	(+26.59)	(+407.37)	(+213.10)
Sugeno 1	1.1213	0.4545	0.4566
	(+107.80)	(+805.38)	(+386.26)
ANFIS 1	0.8692	0.1197	0.1829
	(+61.08)	(+138.45)	(+94.78)
ANFIS 2	0.9790	0.1643	0.1824
	(+81.43)	(+227.29)	(+94.25)
CANFIS 1	**0.5396**	0.1811	0.1575
	(N/A)	(+260.76)	(+67.73)
CANFIS 2	0.7521	**0.0502**	**0.0939**
	(+39.38)	(N/A)	(N/A)
CANFIS 3	0.6679	0.0939	0.0982
	(+23.78)	(+87.05)	(+4.58)

Table 3 Number of fuzzy rules and fitting parameters for the fuzzy and neuron-fuzzy systems

Fuzzy and NF models	Fuzzy rules	Fitting parameters	
		PP	CP
Mamdani 1	150	72	——
Mamdani 2	151	72	——
Sugeno 1	100 linear	81	600
ANFIS 1			
ANFIS 2	57 nonlinear	81	342
CANFIS 1			
CANFIS 2–3	57 neural	81	2850

Successively, ANFIS models are implemented starting from a first-order Sugeno model (Sugeno 1), which is a modified version of the previous Mamdani 1 FIS structure, based on 27 generalized bell MFs for the input variables, respectively, five for M, α, p, r and seven for β, and 17 MFs for the output $\Delta\beta$. Sugeno 1 produces worse performance if compared with Mamdani 1 because the MF number is increased and generalized bell MFs have more degrees of freedom than triangular ones. In addition, the fuzzy IF-THEN rules come down to 100. Nevertheless, the ANFIS extension, ANFIS 1, generated from Sugeno 1 through the premise and consequent parameter training, seems to benefit from the hybrid learning procedure described in the preceding section. In fact, comparing Sugeno 1 and ANFIS 1 in Table 2, it can be noticed how the maximum error is reduced about 23% and the mean error about 74%. Considering a cut-down number of fuzzy rules, ANFIS 1 turns into ANFIS 2, in which conflicting rules have been removed and the FIS manages just 57 rules instead of the 100 fuzzy rules of ANFIS 1. From the neuro-fuzzy spectrum viewpoint, it means that linguistic interpretability increases while a lower mapping precision is achieved. Results in Table 2 underline this concept, showing a degraded performance of ANFIS 2 with respect to ANFIS 1.

If linguistic interpretability is a concern and we want to pursue a higher input-output mapping precision without more computational effort, linear rules have to be replaced with nonlinear and neural rules by using CANFIS systems. In particular, CANFIS 1 represents the first step of this extension process: the linear consequent part of each fuzzy rule is replaced by a hyperbolic tangent activation function. CANFIS 1 uses a total of 423 fitting parameters, 81 of which are premise parameters and 342 are consequent parameters, as summarized in Table 3. Without a priori knowledge, initial setup of consequent parameters is carried out with random numbers; in this way, significant results can be achieved only after a large amount of learning epochs. Table 2 presents the good results accomplished after 200 training epochs: the maximum error is reduced about 45% in comparison to ANFIS 2 model with the same number of fuzzy rules. In addition, Fig. 6 shows the root mean square error (RMSE) trend for training and checking data during learning epochs. The RMSE curves indicate that most of the learning is done in the first 50 epochs.

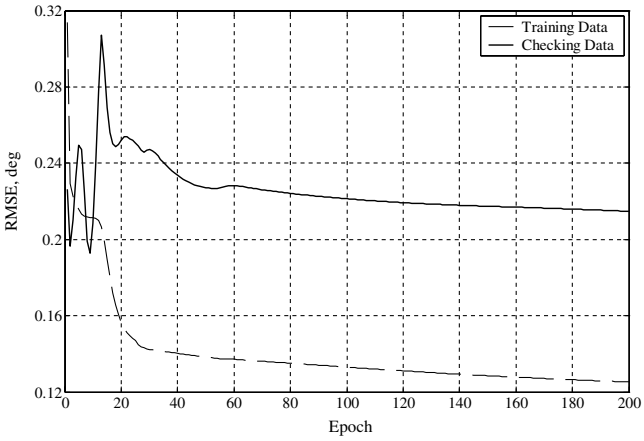

Fig. 6 RMSE for training and checking data in CANFIS 1.

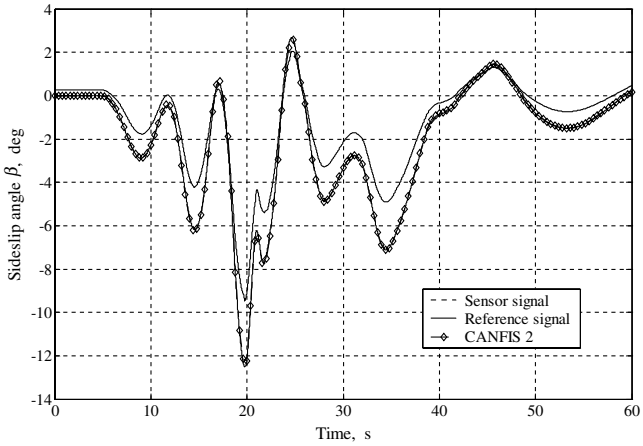

Fig. 7 Test set for CANFIS 2.

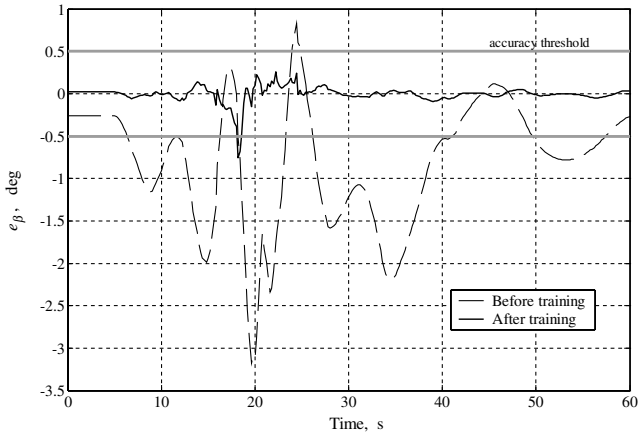

Fig. 8 Absolute error for CANFIS 2 before and after learning.

The simulation has been stopped after 200 epochs with $\mathrm{RMSE}_{\mathrm{trn}} = 0.1252$ and $\mathrm{RMSE}_{\mathrm{chk}} = 0.1929$, while both RMSE trends were still slowly decreasing. In the next step, nonlinear rules are replaced by neural rules. The new CANFIS model, CANFIS 2, seems to have more learning power without losing interpretability. Each rule consequent part is accomplished by a three-layered NN with seven bipolar hyperbolic tangent neuron functions in the hidden layer and one identity neuron function in the output layer. The fitting parameters are 2931, 81 of which are still premise parameters and 2850 are consequent parameters. Figures 7 and 8 illustrate the results for CANFIS 2: after 110 learning epochs the simulation has been stopped with $\mathrm{RMSE}_{\mathrm{trn}} = 0.0378$ and $\mathrm{RMSE}_{\mathrm{chk}} = 0.0939$, whereas the maximum error is reduced about 76% and the mean error about 95%. CANFIS 3 is a variant of CANFIS 2, in which 50 learning epochs are accomplished by using data from the four maneuvers instead of just the first one; the results do not show remarkable improvements.

CANFIS models achieve the best results in terms of maximum/mean errors and fulfill the β accuracy requirements (0.5–0.75 deg); however, the maximum error occurs when the slope of β trend is maximum and not for high β_{ind} values. In this way, the maximum percentage error attains to about 25% of the β sensor indication. This problem could be solved by including the dynamic effects $\dot{\beta}$ in the calibration input variables. In particular, the new Mamdani-type FIS, based on six calibration variables increasing both the total number of MFs and fuzzy rules, achieves better results in comparison to Mamdani 1 with the reduction of the maximum and the mean error of about 14 and 50%, respectively. In addition, applying to this basic fuzzy frame the CANFIS model with nonlinear rules, the learning process presents remarkable results after 78 epochs, with a mean error below 0.1 deg and a maximum error of about 0.63 deg, occurring for the maximum value of β sensor, that is $\beta_{\mathrm{ind}} = -9.5$ deg. Therefore, the maximum percentage error is below the 7% of the sensor indicated value and this result is accomplished through a reduced number of learning epochs with respect to the other CANFIS models of Table 2.

VI. Conclusions

The purpose of this work has been to show the effectiveness of the neuro-fuzzy methodology in the air-data sensor calibration. In particular, the innovative neuro-fuzzy calibration system has been designed and implemented within the Matlab/Simulink simulation environment. Calibration results are promising and remarkable: four short maneuvers of 60 s each allow the $\Delta\beta$ calibration errors to fulfill the accuracy requirements. Once the neuro-fuzzy calibration system has been trained and tuned for the selected aircraft-sensor configuration, the correction is nonadaptive, which means that the training algorithm can be dismissed and the calibration error calculation can be performed with simple direct algebraic operations, typical of the neuro-fuzzy logic. The calculation times are, hence, fairly compatible with the acquisition frequency of 100 Hz of the onboard flight control computers [36]; this makes the neuro-fuzzy methodology suitable to real-time applications. Finally, particular attention has been focused on the improvement of the neuro-fuzzy models to achieve more accuracy and mapping precision and preserve, at the same time, the physical meaning and other intuitive features of the fuzzy rules, namely, the interpretability of the system. New improvements could result from different methods to generate fuzzy rules from numerical input-output data, such as hyperboxes representation [37], genetic algorithms, or clustering techniques [38,39]. This is a very critical phase in the FIS process and more describing rules would be needed to increase performance of the neuro-fuzzy systems.

References

[1] Haering, E. A., Jr., "Airdata Measurement and Calibration," NASA TM-104316, 1995.

[2] Haering, E. A., Jr., "Airdata Calibration of a High-Performance Aircraft for Measuring Atmospheric Wind Profiles," NASA TM-101714, 1990.

[3] Lawford, J. A., and Nippress, K. R., "Calibration of Air-Data Systems and Flow Direction Sensors," AGARD AG-300, Vol. 1, 1983.

[4] Whitmore, S., "Reconstruction of the Shuttle Reentry Air Data Parameters Using a Linearized Kalman Filter," AIAA Paper 83-2097S, 1983.

[5] Whitmore, S., Larson, T. J., and Ehrenberger, L. J., "Air-Data Position-Error Calibration Using State Reconstruction Techniques," NASA TM-86029, 1984.

[6] Whitmore, S., "Formulation and Implementation of a Nonstationary Adaptive Estimation Algorithm with Applications to Air-Data Reconstruction," NASA TM-86727, 1985.

[7] Tseng, Y. C., and Lan, C. E., "Calibration of an Aircraft Sideslip Sensor by Fuzzy Logic," AIAA Paper 97-3626, 1997, pp. 1144–1149.

[8] Zadeh, L. A., "Fuzzy Sets," *Information and Control*, Vol. 8, No. 3, June 1965, pp. 338–353.

[9] Jang, J.-S. R., "ANFIS: Adaptive-Network-Based Fuzzy Inference System," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 23, No. 3, 1993, pp. 665–685.

[10] Mizutani, E., Jang, J.-S. R., Nishio, K., Takagi, H., and Auslander, D. M., "Coactive Neural Networks with Adjustable Fuzzy Membership Functions and Their Applications," *Proceedings of the International Conference on Fuzzy Logic, Neural Networks, and Soft Computing*, Fuzzy Logic Systems Institute, Iizuka, Fukuoka, Japan, 1994, pp. 581–582.

[11] Mizutani, E., and Jang, J.-S. R., "Coactive Neural Fuzzy Modeling," *Proceedings of the IEEE International Conference on Neural Networks*, Vol. 2, IEEE Computer Society Press, Piscataway, NJ, 1995, pp. 760–765.

[12] Stevens, B. L., and Lewis, F. L., *Aircraft Control and Simulation*, John Wiley & Sons, New York, 1992.

[13] Nguyen, L. T., Ogburn, M. E., Gilbert, W. P., Kibler, K. S., Brown, P. W., and Deal, P. L., "Simulation Study of Stall/Post Stall Characteristics of a Fighter Airplane with Relaxed Longitudinal Static Stability," NASA TP-1538, 1979, pp. 43–93.

[14] Psaltis, D., Sideris, A., and Yamamura, A., "A Multilayered Neural Network Controller," *IEEE Control Systems Magazine*, Vol. 8, No. 2,

1988, pp. 17–21.
[15] Gili, P., and Battipede, M., "An Adaptive Neurocontroller for Nonlinear Combat Aircraft Model," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 5, 2001, pp. 910–917.
[16] Gili, P., and Battipede, M., "Adaptive Features of a MIMO Full-Authority Controller," AIAA Paper 2000-4282, 2000.
[17] Battipede, M., Gili, P. A., and Lando, M., "Neuro-Fuzzy Calibration of Air-Data Sensors," AIAA Paper 01-4375, 2001.
[18] *Assessment of Experimental Uncertainty with Application to Wind Tunnel Testing*, AIAA Standards Series, S-071A, AIAA, Reston, VA, 1999.
[19] Wang, L.-X., *Adaptive Fuzzy Systems and Control, Design and Stability Analysis*, Prentice Hall, Upper Saddle River, NJ, 1994, pp. 65–82.
[20] Jang, J.-S. R., Sun, C.-T., and Mizutani, E., *Neuro-Fuzzy and Soft Computing*, Prentice Hall, Upper Saddle River, NJ, 1997.
[21] Mamdani, E. H., and Assilian, S., "An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller," *International Journal of Man-Machine Studies*, Vol. 7, No. 1, 1975, pp. 1–13.
[22] Zadeh, L. A., "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 3, No. 1, 1973, pp. 28–44.
[23] Horikawa, S., Furuhashi, T., and Uchikawa, Y., "On Fuzzy Modeling Using Fuzzy Neural Networks with the Back-Propagation Algorithm," *IEEE Transactions on Neural Networks*, Vol. 3, No. 5, 1992, pp. 801–806.
[24] Wang, L.-X., and Mendel, J. M., "Back-Propagation Fuzzy System as Nonlinear Dynamic System Identifier," *IEEE International Conference on Fuzzy Systems*, IEEE Computer Society Press, Piscataway, NJ, 1992, pp. 1409–1418.
[25] Takagi, T., and Sugeno, M., "Fuzzy Identification of Systems and its Applications to Modeling and Control," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 15, Jan.–Feb. 1985, pp. 116–132.
[26] Lee, C.-C., "Fuzzy Logic in Control Systems: Fuzzy Logic Controller —Part 1," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 20, No. 2, 1990, pp. 404–418.
[27] Lee, C.-C., "Fuzzy Logic in Control Systems: Fuzzy Logic Controller —Part 2," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 20, No. 2, 1990, pp. 419–435.
[28] Gill, P. E., and Murray, W., *Numerical Methods for Constrained Optimization*, Academic Press, London, 1974.
[29] Wismer, D. A., and Chattergy, R., *Introduction to Nonlinear Optimization: a Problem Solving Approach*, North-Holland, Amsterdam, 1978, pp. 163–178, Chap. 7.
[30] Battipede, M., Gili, P. A., and Lando, M., "Virtual Air-Data Sensor Calibration Through a CANFIS Model," AIAA Paper 02-3788, 2002.
[31] Jang, J.-S. R., "Fuzzy Modeling Using Generalized Neural Networks and Kalman Filter Algorithm," *Proceeding of the 9th National Conference on Artificial Intelligence*, AAAI Press, Menlo Park, CA, 1991, pp. 762–767.
[32] Riedmiller, M., "RPROP—Description and Implementation Details," Technical Report, Institut für Logik, Komplexität und Deduktionssyteme, Karlsruhe University, Karlsruhe, Germany, 1994.
[33] Chen, M. S., and Liou, R. J., "An Efficient Learning Method of Fuzzy Inference System," *Proceedings of the IEEE International Fuzzy Systems Conference*, Vol. 2, IEEE Computer Society Press, Piscataway, NJ, 1999, pp. 634–638.
[34] Fahalman, S., "An Empirical Study of Learning Speed in Backpropagation Networks," Technical Report CMU-Cs-88-162, Computer Science Dept., Carnegie-Mellon University, Pittsburgh, PA, 1988.
[35] Marquardt, D. W., "An Algorithm for Least Squares Estimation of Nonlinear Parameters," *Journal of the Society for Industrial and Applied Mathematics*, Vol. 11, No. 2, June 1963, pp. 431–441.
[36] Menzies, M. A., "Integrated Air Data Sensors," *The Aeronautical Journal*, Vol. 105, April 2001, pp. 223–229.
[37] Abe, S., and Lan, M. S., "Fuzzy Rules Extraction Directly from Numerical Data for Function Approximation," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 25, No. 1, 1995, pp. 119–129.
[38] Mikhailov, L., Nabout, A., Lekova, A., Fischer, F., and Nour Eldin, H. A., "Method for Fuzzy Rules Extraction from Numerical Data," *Proceedings of the 12th IEEE International Symposium on Intelligent Control*, IEEE Computer Society Press, Piscataway, NJ, 1997, pp. 61–65.
[39] Thawonmas, R., and Abe, S., "Function Approximation Based on Fuzzy Rules Extracted from Partitioned Numerical Data," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 29, No. 4, 1999, pp. 525–534.