Advantages of Neural Network Based Air Data Estimation for Unmanned Aerial Vehicles

Advantages of Neural Network Based Air Data Estimation for Unmanned Aerial Vehicles

Angelo Lerro, Manuela Battipede, Piero Gili, Alberto Brandl

***Abstract*—**Redundancy requirements for UAV (Unmanned Aerial Vehicle) are hardly faced due to the generally restricted amount of available space and allowable weight for the aircraft systems, limiting their exploitation. Essential equipment as the Air Data, Attitude and Heading Reference Systems (ADAHRS) require several external probes to measure significant data as the Angle of Attack or the Sideslip Angle. Previous research focused on the analysis of a patented technology named Smart-ADAHRS (Smart Air Data, Attitude and Heading Reference System) as an alternative method to obtain reliable and accurate estimates of the aerodynamic angles. This solution is based on an innovative sensor fusion algorithm implementing soft computing techniques and it allows to obtain a simplified inertial and air data system reducing external devices. In fact, only one external source of dynamic and static pressures is needed. This paper focuses on the benefits which would be gained by the implementation of this system in UAV applications. A simplification of the entire ADAHRS architecture will bring to reduce the overall cost together with improved safety performance. Smart-ADAHRS has currently reached Technology Readiness Level (TRL) 6. Real flight tests took place on ultralight aircraft equipped with a suitable Flight Test Instrumentation (FTI). The output of the algorithm using the flight test measurements demonstrates the capability for this fusion algorithm to embed in a single device multiple physical and virtual sensors. Any source of dynamic and static pressure can be integrated with this system gaining a significant improvement in terms of versatility.

***Keywords*—**Neural network, aerodynamic angles, virtual sensor, unmanned aerial vehicle, air data system, flight test.

I. Introduction

THE past 20 years has seen a rapid grown in the interest for UAV (Unmanned Aerial Vehicle) systems. According to [1], [2] nearly 80000 UAV should have been produced between 1994 and 2003, worth $3.9 billion. At the time of writing this article, the annual report from the AUVSI (Association of Unmanned Vehicle System International) of 2013 points out a total economic impact of $13.6 billion for 2015-2017 period for the USA only. The debate about safe integration of UAV into national airspace has gained importance during last years [3]. A recent study by Freeman et al. [4] recalled how ADS (Air Data System) can lead to catastrophic failure even in case of hardware redundancy combined with voting systems. At the same time, an increasing number of study developed an evolution in FDI (Fault Detection and Isolation) systems. A recent systematic literature review could be seen in [5]. Current research projects show an evolution of the FDI with sensor fault accomodation named SFDIA (Sensor Fault Detection, Isolation and Accomodation) system. As previously mentioned, the classical solution adopted in order to increase the fault tolerance of a complex system is the hardware redundancy with voting system. Current state-of-the-art air data sensors are made of several physical units, each of them requiring power supply, a de-icing system when needed and a certain number of conditioning and computing module. Furthermore, the external sensor should be as much as possible positioned in a clear aerodynamic environment. A number of authors have considered the analytical redundancy as useful solution to the problem [4]-[7]. However, in some cases the implementation of an analytical redundancy may not be enough to allow an efficient design of the entire architecture. The proposed approach (patented technology, [8]) allows a reduction in terms of physical external sensors thanks to an innovative fusion algorithm based on NN (Neural Network). This integration will simplify the entire architecture with resulting reduction in terms of cost and maintainability. The virtual sensor properly trained would be able to determine the entire set of inertial and air data including AOA (Angle of Attack) and AOS (Angle of Sideslip) with an accuracy suitable for an FCS (Flight Control System). These two angles, generally called aerodynamic angles, are defined as in Fig. 1. Previous research focused on the definition of suitable architectures together with analysis in simulated environment (see [9]-[11]). A further work, already accepted but not yet presented at the time of writing this article, showed the current Smart-ADAHRS system [12]. Section II will briefly describe the approach by a NN point of view, whereas Section III will depict the potential advantages in case of integration on UAV, comparing the proposed approach with current research. Section V will show training and test results in case of ideal simulated environment and real flight test scenario.

A. Lerro is with AeroSmart s.r.l., Caserta, Italy (e-mail: angelo.lerro@aerosmartsrl.it).

M. Battipede is Associate Professor at Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Turin, Italy (e-mail: manuela.battipede@polito.it).

P. Gili is Associate Professor at Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Turin, Italy (e-mail: piero.gili@polito.it).

A. Brandl is PhD student at Department of Mechanical and Aerospace Engineering, Politecnico di Torino, Turin, Italy (e-mail: alberto.brandl@polito.it).

II. Air Data System Based on Neural Network

Traditionally, aircraft have been equipped with several sensors in order to measure air data. This essential set of signals refers to static and dynamic pressure, aerodynamic angles and relative speed with respect to the wind. However, size and weight requirements may not be easy to meet during the design phase of UAV. Moreover, reliability performance have to be considered. Hence, reducing the number of external physical sensors implementing a software solution could bring

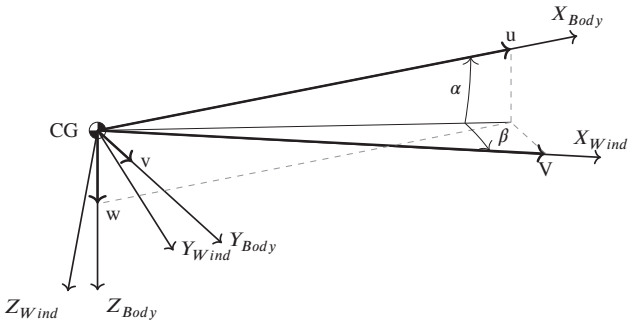

Fig. 1 Body and Wind reference frames with detail of aerodynamic angles α (Angle of Attack, also referred as AoA) and β (Sideslip Angle, also referred as AoS)

to important improvements. In this work the aerodynamic angles α (or AOA) and β (or AOS) are measured by splitting them in a linear estimation, evaluated by a LE (Linear Estimator), and a non-linear value obtained by means of an ANN (Artificial Neural Network) using data coming from a single conventional probe and GPS/INS measurements. In this way, only one external sensor is required. An MLP (Multilayer Perceptron) is a ANN considered as a universal function approximator. A lot of prior work has been carried out to support and proof this property (see [13], [14]). Equation (1) is a general mathematical model for an MLP with a single hidden layer.

$$y = \sum_{j=0}^{M} w_j^{(2)} g\left(\sum_{i=0}^{d} w_{ji}^{(1)} x_i\right) \tag{1}$$

where y is the ANN output, x_i is one of the d network input, $w_{ji}^{(1)}$ is the weight connecting the i-th input to the j-th node of the hidden layer with M neurons, $w_j^{(2)}$ is the weight connecting the j-th node to the output neuron and g is the activation function. Equation (1) modifies in (2) for two hidden layers, where M_2 is the number of neurons in the second hidden layer.

$$y = \sum_{k=0}^{M_2} w_k^{(3)} g_2\left(\sum_{j=0}^{M} w_{kj}^{(2)} g_1\left(\sum_{i=0}^{d} w_{ji}^{(1)} x_i\right)\right) \tag{2}$$

$w_{kj}^{(1)}$ is the weight connecting the j-th node of the first hidden layer to k-th node of the second hidden layer, $w_k^{(3)}$ represents the weight connecting the output of the k-th unit of the second hidden layer to the output neuron whereas g_1 and g_2 are the generic activation functions of the neurons respectively for the first and the second hidden layer. Train the NN means to find the weight matrix W able to minimize an arbitrary error function, which is defined considering the target t and the network output y. In case of regression, a typical choice for the error function is the SSE (Sum-of-Squares Error) modeled in (3). The method applied to find the weight matrix W, named the training algorithm, is based on two processes. The first one is called Back Propagation (BP). This algorithm is needed in order to compute the error function derivatives with respect to the weights. The second process is an optimization algorithm used in order to determine the adjustments to be made to the weights (see [15]).

$$E = \sum_{n=1}^{N} \{y\left(\boldsymbol{x^n}; \boldsymbol{w}\right) - \boldsymbol{t^n}\}^2 \tag{3}$$

To train an MLP using SSE with a linear activation function on the output layer, in the limit of an infinite Training Set (TS), the residual error between target and output will be normally distributed. A proof that the output of an MLP could reach exactly the conditional average of the target data can be seen in [15]. This proof is of fundamental importance for this equipment. It means that in case of suitable training, the output of the ANN could match with the real non-linear static value of the aerodynamic angle. Because MLP is a valid method to obtain a non-linear regression, they could represent a valid substitute to physical sensor for the determination of the aerodynamic angles. Several optimization algorithms have been developed during last decades. One of the most common heuristic is the Levenberg-Marquardt (LM) method (see [16], [17]), which adapt a parameter λ to pass from a standard gradient descent approach for large value of λ to a Gauss-Newton formula for small value of λ. It represents an example of trust region approach applied to Gauss-Newton method. The main mathematical description for this method is reported in (4), where $\boldsymbol{Z}$ is the Jacobian matrix of the error function with respect to the weights whereas $\boldsymbol{w_{old}}$ and $\boldsymbol{w_{new}}$ represents respectively the old and new weight vectors expressed in the weight space. $\epsilon\left(\boldsymbol{w_{old}}\right)$ is the residual error applying $\boldsymbol{w_{old}}$.

$$\boldsymbol{w_{new}} = \boldsymbol{w_{old}} - \left(\boldsymbol{Z}^T\boldsymbol{Z} + \lambda\boldsymbol{I}\right)^{-1}\boldsymbol{Z}^T\epsilon\left(\boldsymbol{w_{old}}\right) \tag{4}$$

Although LM method avoids the calculation of the Hessian matrix, it is quite heavy in terms of memory and computational cost due to the evaluation of the Jacobian matrix. Moreover, there are some implications using the partial derivative of the error function for the direct modification of the weight matrix. The unforeseeable behaviour of the derivative itself could indeed bring to very slow learning or disturbances in the training procedure. A possible way to address this problem could be change the optimization algorithm with the RPROP (Resilient Propagation) in which the weight update step is function only of the sign of the derivative. For a complete description of the method please see [18]. General weight update step is reported in (5).

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} + \Delta w_{ij}^{(t)} \tag{5}$$

where

$$\Delta w_{ij}^{(t)} = \begin{cases} -\Delta_{ij}^{(t)} & \text{, if } \frac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ +\Delta_{ij}^{(t)} & \text{, if } \frac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ 0 & \text{, else} \end{cases} \tag{6}$$

and

$$\Delta_{ij}^{(t)} = \begin{cases} \eta^{+}\Delta_{ij}^{(t-1)} & \text{, if } \frac{\partial E^{(t-1)}}{\partial w_{ij}} \frac{\partial E^{(t)}}{\partial w_{ij}} > 0 \\ \eta^{-}\Delta_{ij}^{(t-1)} & \text{, if } \frac{\partial E^{(t-1)}}{\partial w_{ij}} \frac{\partial E^{(t)}}{\partial w_{ij}} < 0 \\ \Delta_{ij}^{(t-1)} & \text{, else} \end{cases} \quad (7)$$
$$\text{where } 0 < \eta^{-} < 1 < \eta^{+}$$

In [19] a comparison between different heuristic rules is carried out. Findings show how the LM algorithm is usually the best one in terms of speed convergence and training error but, at the same time, RPROP is able to obtain quick convergence with better residuals during a further test procedure. See [20] for further optimization algorithm comparison. At first, the selected heuristic was the LM and good results were obtained during previous research (see [9]-[12]). However, this work introduces in this project the training with multiple simultaneous TSs and initial trials required too much time and memory. For the reasons listed above, in this paper the RPROP method will be considered.

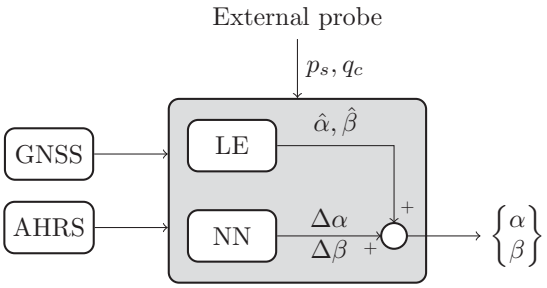

Fig. 2 General schematic of the Smart-ADAHRS

An infinite TS is obviously not available in a realistic application. The selection of a suitable set of input-target pair is one of the most difficult part of the NN design, together with the definition of which signals apply for the input pattern. Moreover, this procedure brings to the so-called bias-variance dilemma. To face this problem, the ANN has been simultaneously trained with multiple training sets. The benefits of this approach are the possibility of consider a larger quantity of cases with one singular training. In addiction, although this is a static network, the aircraft is a dynamic system involving time evolution and the ANN will learn from physically corrected trajectory. This method is particularly useful when learning from data logging coming from real flight tests, when the number of possible maneuvers is limited by fuel consumption, pilot experience, available time and flight test purpose. In the following TS will stay for a set of several trajectory in place of a singular one. The selection of the input signal has been described in previous research [9]-[12]. Starting from aircraft dynamics, it is possible to define a implicit redundancy of the measurements. The AOA and AOS are indeed strictly related to the inertial and the remaining air data values of the aircraft at each moment. The proposed input signal applied in this article is slightly changed from the previous ones. Keeping only on-board measurements, the derivative of the dynamic pressure is introduced as in (8).

$$\Delta\alpha = f_{\alpha}\left(\hat{\alpha}, \dot{q_c}, q_c, q, \phi, n_x, n_z, V_D\right) \quad (8)$$

where $\hat{\alpha}$ is the linear estimation of the AOA, q_c and $\dot{q_c}$ are the dynamic pressure and its time derivative, q is the pitching angular rate, ϕ is the roll angle, n_x and n_z are the accelerations measured along X_{Body} and Z_{Body} and V_D is the velocity component along the local Down versor.

III. Comparison between Classical ADS and Smart-ADAHRS

Current research in UAV revealed additional limitations for designers. The restriction of available space and weight may be a question on system design, especially when talking about sensors. Today, several solutions can be applied for the estimation of attitude parameters and air data. However, integrated solution are not so common. In Table I a list of commercial sensors for AOA is shown. As the values reported in the table suggest, there is not a unique architecture. Simpler sensors are very light but often they are not provided with an anti-ice or de-icing system. However, a general classification could be depicted to show the main advantages of a NN-based ADS. Current state-of-the-art ADS may be divided in two main groups:

1) *Conventional probes*: a different external unit for AOA, AOS, static and dynamic pressure (usually combined in the well-known Pitot tube), temperature, recently analitically integrated by multi-sensor fusion techniques;
2) *Multi-Function probes*: several multi-hole probes placed in particular position of the aircraft, usually the nose, integrated with a complex algorithm based on curve calibration

In both cases, the classical hardware redundancy will multiply the number of units and connection by at least three, for a triplex physical redundancy, or even four. Moreover, in some cases the increased number of external units might not avoid reliability issues. As reported in [21], an investigation conducted by Airbus and Thales showed that an incorrect removal of machining oil during the manufacturing process of AOA resolver can bring to delayed or reduced AOA vane movement. This kind of fault could affect more than one sensor and hence could lead to delayed activation or non-activation of the AOA protection system. Eventually, the aircraft could exhibit a reduced controllability. In [22] the blockage of two AOA probes during climb led to the activation of a protection system on Airbus A321. In a worst case scenario, pilots could become not able to oppose to a nose down command if the Mach number increases. In the same Airworthiness Directive [22] the AOA sensors is claimed as necessary to maintain the required safety level of the aircraft. Pros and cons of sensor equipped with moving parts with respect to fixed multi-hole probe rely on accuracy of angle determination by means of several techniques. Current state-of-the-art solution involves potentiometers, RVDTs and synchro. Permanent magnet solution is presented in [23]. However, as reported in previously cited EASA Airworthiness Directive ([21], [22]) moving parts might be subjected to

TABLE I
EXAMPLES OF COMMERCIAL AIR DATA PROBES

Manufacturer	Model	Weight [g]	Heater Power [W]
UTC Aerospace	0012 AOA Transmitter	567	425
SpaceAge Control	4239-01	454	100
SpaceAge Control	101100 (micro air data boom)	142	unheated
SpaceAge Control	100900	5440	not available
AMETEK	Total Air Probe	900	not available
AMETEK	AOA transducer 25147A	816	270
AMETEK	AOA transducer 2568A	1814	270
Aerosonic	AOA	1360	150
Aerosonic	SWT	1360	190
Aerosonic	SWT	1360	450
Ack Emma LLC	CYA-100	56	unheated

TABLE II
EXAMPLES OF COMMERCIAL ADAHRS

Manufacturer	Model	Provides AOA/AOS
Cobham	ADAHRS	no
Northropp-Grumman	LCR-300	yes
Honeywell	KSG 7200	no
Archangel	AHR300A	no

TABLE III
ADS AND AHRS RAMS PERFORMANCE

Item	FR [10^{-6}]	MTBF
AOA sensor	50	20000
Air Data Probe	20	50000
Electrical Connector	0.0163	$61.35 \cdot 10^6$
Pneumatic Tube	0.1104	$9.05 \cdot 10^6$
Air Data Computer	130	7692
Gyroscope or accelerometer	64	15625
GPS Antenna	20	50000
GPS Receiver	20	50000
Power Supply	31	32000

delayed motion or even blockage. Alternative solutions have been discussed in the past literature as can be seen in [25], where moving vane and fixed fin equipped with strain gauge have been analysed. Another patent related to a multi-hole probe can be seen in [24]. Where possible, the external structure could be aerodynamically designed to passively avoid the ice build-up without heating (see [26]). Except for those rare cases, the external sensor must comply with safety regulations about de-icing. Typical regulation could be the MIL-STD-810. As seen in Table I values between 150 W and 400 W per probe could be considered valid for the electrical heater consumption (see [27]-[31]). Hardware redundancy will multiply the power requirements. Hence, a reduction on the number of external probes might be considered a possible alternative to the current state-of-the-art solutions.

Previous discussion mainly focused on probe for AOA and ADS but actually an ADC (Air Data Computer) is generally required, except for highly federated architecture where the unit is already provided with its own processing unit. Due to the high number of combinations between sensors and processing unit, comparing the ADAHRS architectures is not easy. Some examples of ADAHRS units are reported in Table II. To the best of our knowledge, only one is able to provide AOA or AOS signal among the ones listed. Often an additional equipment is required. RAMS performance of the entire ADS architecture must be taken into account as well. Table III shows data related to FR (Failure Rate) taken from [32], [33].

Define the position of the external probes is not easy as well. They should be placed in way of avoid any aerodynamic influence with other aircraft parts. For instance, propeller will produce a turbulent aerodynamic field that will induce oscillations on both an AOA vane and multi-hole AOA probe. Moreover, thinking about small UAV applications, a typical payload could be a camera and the position of the external parts should not interfere or obstruct the FOV (Field of View) of the camera itself. The proposed algorithm is hence particularly useful when there is a control system which needs a reliable AOA/AOS signal and it is difficult to meet redundancy requirements or, in the worst case scenario, it is not even possible to place the traditional sensor in the right position. The only needed external sources are static and dynamic pressure, that are usually some of the most common probes. The remaining input signals needed by this virtual sensor could be given by an inertial MEMS-based platform and GNSS receiver. These equipments are very common and validated in modern aircraft and UAVs, usually properly treated by a multi-sensor fusion techniques as the most famous Kalman Filter (see [34], [35]). In this way, an important simplification of the entire architecture is obtained. Moreover, all kinds of redundancy could be implemented easily, also in difficult situations as in case of small and medium UAV.

Different solutions have been proposed in literature to obtain a virtual sensor. Some of them are model-based, hence requiring a phase of definition of the model parameter. For instance [6] proposed a virtual sensor for the Angle of Attack (AoA) that split in three parts the signal to estimate: a trimmed angle of attack obtained by means of a Takagi-Sukeno fuzzy model, a short period AoA from linear short period approximation and a third part obtained by means of a neural network. Reference [7] developed an Adaptive Kalman Filter (AEKF) to estimate AoA and Calibrated Air Speed (CAS). Another proposed approach consider an aerodynamic model inversion (AMI) and was previously described in [36]. Model-free solutions exist as shown in [37], where an identification of the aerodynamic coefficient from sparse data has been conducted using ANN trained as described in [38]. A further different algorithm is described in [39] where a Functional Pooling Nonlinear AutoRegressive with eXogenous excitation (FP-NARX) is applied in order to directly obtain the AoA signal. Although the system proposed in this paper is static, its simplicity should be a great advantage. The

application of a learning procedure able to autonomously define a model avoids a lot of computation, giving a light and fast response equipment.

IV. Methodology

In order to demonstrate the capability of the Smart-ADAHRS of substitute state-of-the-art ADS, two different test cases will be shown. For a general validity of the algorithm, an ideal scenario will be simulated. Eventually, real signals coming from flight tests will be applied for training and test procedures. Aircraft dynamics simulations and NN operations have been conducted by means of a properly written MATLAB code. The Neural Network Toolbox has been implemented in this work. The NN has been simultaneously trained with different TS, each one lasting for 2000 s so that the dynamic of the aircraft should be generally covered. In this way, the NN can learn as much number of states as possible for the AOA signal. Being the training procedure subjected to initial condition, which are generally randomly selected, several run will be carried out. When all training operations will be concluded, the one with minimum global NSSE (Normalized Sum-of-Squares Error) will be taken into account for testing. Using NSSE allows to compare different training operations without considering the length of the training set. For each run, the TS has been partitioned in three parts respectively for training, test and validation (see [40]). The first partition has been actually used for weight updating, the validation only to check over-fitting on the training partition and the latter one has been applied to compare different models. This method will reduce the risk of over-fitting on training data. For global NSSE the entire TS is considered. The SSE definition reported in (3) can hence be re-written for the three partitions as in (9). Preferring NSSE to compare the obtained models, results will be given as in (10).

$$SSE_i = \sum_{n=1}^{N} \Omega_i(n)\{y(\boldsymbol{x^n};\boldsymbol{w}) - \boldsymbol{t^n}\}^2 \tag{9}$$

$$NSSE_i = \frac{1}{N_i}\sum_{n=1}^{N} \Omega_i(n)\{y(\boldsymbol{x^n};\boldsymbol{w}) - \boldsymbol{t^n}\}^2 \tag{10}$$

where

$$\Omega_i(n) = \begin{cases} 1 & \text{if } n \in \boldsymbol{v}_i \\ 0 & \text{if } n \notin \boldsymbol{v}_i \end{cases} \tag{11}$$

and $\boldsymbol{v}_i$ represents a vector containing the indexes of samples selected for the i-th set and i can represent training, test or validation. Actually, a NN will be trained with the same TS for a number of times selected by means of trial-and-error procedure. Previous investigations showed 10 as a good compromise between computational time and good learning. Hence, for a singular training NSSE doesn't give any additional information, but could allow the comparison between different TS, as in case of simulated and flight test scenario. Moreover, authors found that a normalization of the input and output is a promising pre-processing for the NN data. Once the NN has been trained, a different trajectory has been put in input of the Smart-ADAHRS. This operation is needed in order to check how this equipment behaves if a situation not presented during the supervised learning appears. This is one of the easiest way to check the ability of the NN to generalize what it has learnt. The time needed by the entire training procedure has been measured to compare the computational cost of the various architectures applied. All training and test operations have been conducted on a double 4-core processors Intel® Xeon® 2.27 GHz with 16 GB of RAM. For what concern the real flight test scenario, a prototype of Smart-ADAHRS has been developed. This prototype has been mounted on a ULV (Ultra Light Vehicle) manufactured by Ing. Nando Groppo srl, named G-70, together with a fully-equipped FTI (Flight Test Instrumentation). Several flight tests have been carried out between January and June 2016. The issues quickly rise implementing an algorithm on a real aircraft. The reliability of the recorded signals might be affected by a lot of factors. Physical sensor accuracy together with a proper calibration could deteriorate the measurements. One of the main problem could be structural vibrations affecting both the inertial transducer and the AOA vanes. Eventually, the data logger system must be adequate so as measurements taken at the same time would be synchronized. Table IV shows a brief summary of the FTI. All measurements have been elaborated in post-processing after the data log have been downloaded from the prototype.

TABLE IV
FTI Description

System	Model (Producer)	Role
ADAHRS	Spatial (Advanced Navigation)	Main
ADS	Spatial (Advanced Navigation)	Main
AHRS	MTi (Xsens)	Redundancy
GPS	LEA-6R (ublox)	Redundancy

V. Results

In this section some NN will be compared in terms of performance and response to training operation over different TS. For sake of clarity, NN is indicated by a row vector in bracket notation, where each element represents the number of neurons in the i-th hidden layer. For instance, [15 20] is a compact notation for an MLP with 15 neurons on the first hidden layer and 20 on the second one. Moreover, for a single TS is considered a set of different flight recordings at which the NN is trained simultaneously for 10 times with different initial conditions. Similarly the FT acronym will stay for the TS obtained from Flight Tests. Some examples of response of the virtual sensor based on a [20] NN for training procedure using the first TS can be seen in Figs. 3 and 4. The NSSE trend during various training can be seen in Fig. 6.

It is apparent from this figure that the effect of different initial condition could be pronounced. At the same time, marked variations in the final global NSSE could implicate an incomplete training. Although this could mean that the NN has not learned enough from the available TS, one of the

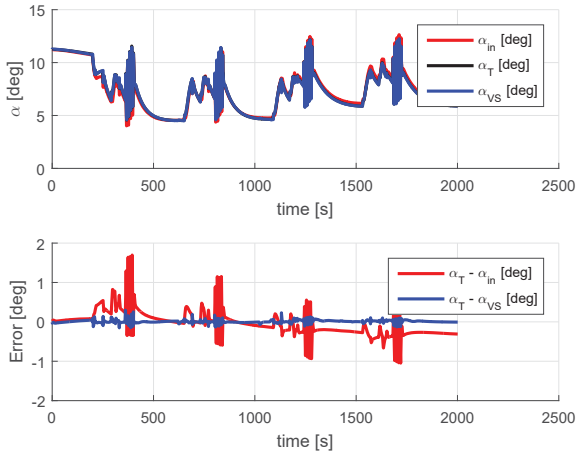

Fig. 3 Training comparison between simulated signal, linear estimation and virtual sensor output using [20] NN on simulated data and TS #1

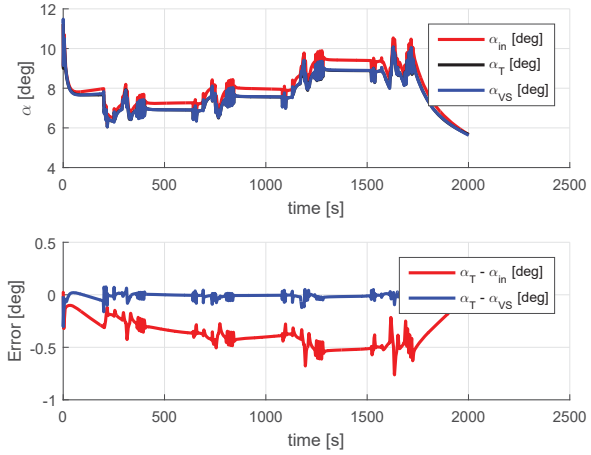

Fig. 4 Training comparison between simulated signal, linear estimation and virtual sensor output using [20] NN on simulated data and TS #1

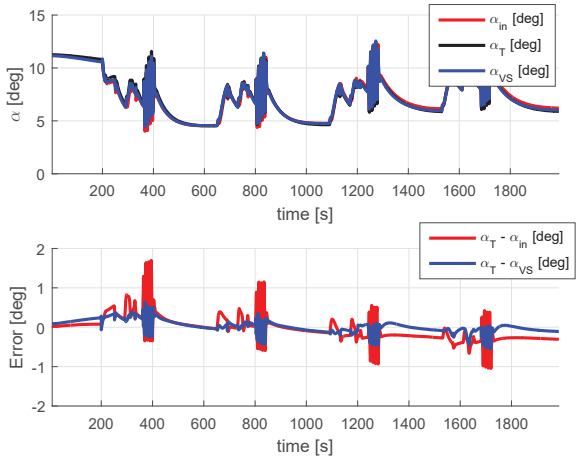

Fig. 5 Training comparison between simulated signal, linear estimation and virtual sensor output using [20] NN on simulated data and TS #2

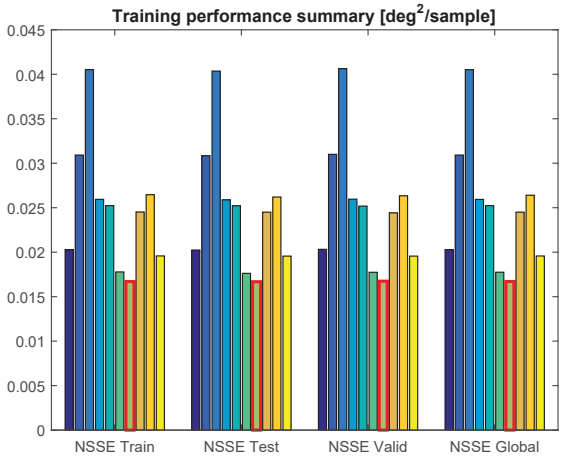

Fig. 6 Training performance over 10 training operations using [20] NN on simulated data and TS #2

most difficult part in NN training is avoid over-fitting. Further analysis showed that an increment on the number of maximum training epochs seems not to bring striking improvement on performance. The effect of changing TS adding a trajectory including more lateral-directional dynamics can be seen in Figs. 5-7. To verify training has not been stopped too early, Fig. 7 provides the SSE plot at each epoch reporting a trend to limit the effective performance. Further information can be seen in Table V.

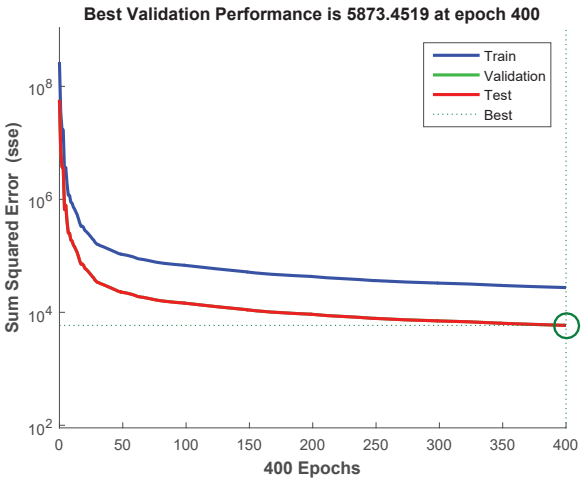

Fig. 7 Training performance in terms of SSE [deg^2/samples] over 400 training epochs using [20] NN on simulated data and TS #2

As previously mentioned, training with flight test measurements is more difficult than using simulated data. Sensor calibration, signal noise and structural vibration are only some of the factors influencing measurements. To obtain suitable trajectories is another issue more easily faced with a simulator. Three different NNs have been trained, respectively with [20], [20 20] and [30 30] structure. Examples of non-linearity of AOA and how the NN can learn is shown in Fig. 11-13. A detail on the test trajectory is shown in Fig. 14. Fig. 10 allows to discuss the necessity of

TABLE V
TRAINING PERFORMANCE OF [20] NN WITH TS #1 AND #2

NN	Opt. Alg.	TS	Mean training error [deg]	NSSE [deg^2/samples] Train	Test	Validation	Global
20	RPROP	1	8.37e-3	2.175e-3	2.175e-3	2.175-3	2.175e-3
20	RPROP	2	5.91e-2	1.67e-2	1.668e-1	1.674e-1	1.67e-1
20	LM	2	2.08e-3	2e-4	2e-4	2e-4	2e-4

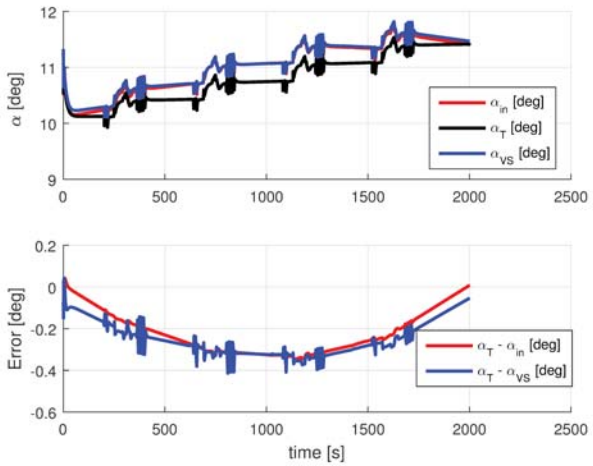

Fig. 8 Test comparison between simulated signal, linear estimation and virtual sensor output using [20] NN on simulated data and TS #1

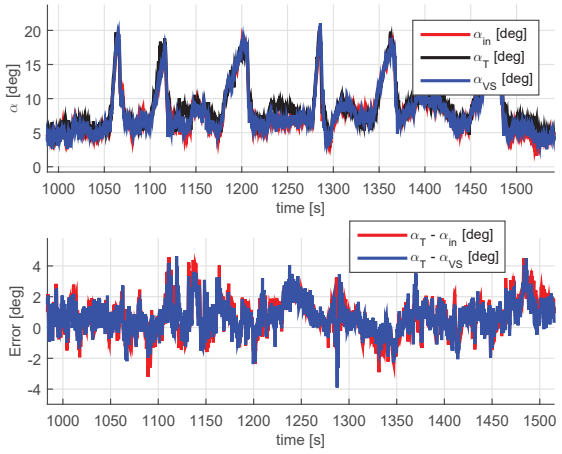

Fig. 9 Examples of non-linearity during test of [20] NN trained with 600 epochs in operative environment

TABLE VI
TRAINING TIME WITH DIFFERENT ARCHITECTURES AND TS (FT STAYS FOR FLIGHT TEST)

NN	TS	Optimization Algorithm	Max epochs	Time [s]
20	1	RPROP	600	1420
20	2	RPROP	2000	2774
20	2	LM	400	55531
20	FT	RPROP	600	1114
20	FT	LM	400	out-of-memory
20 20	FT	RPROP	600	1568
30 30	FT	RPROP	2000	6603

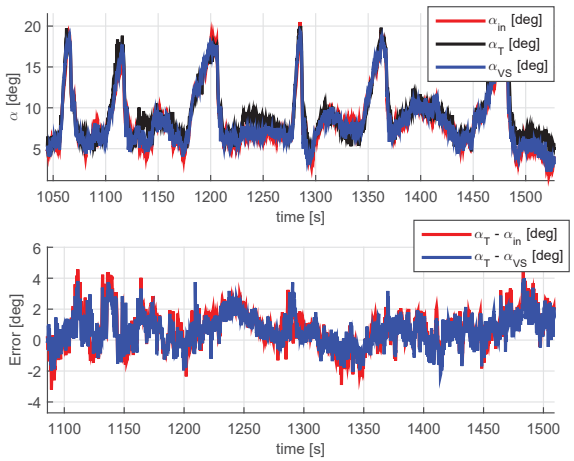

Fig. 10 Examples of non-linearity during test of [20] NN trained with 2000 epochs in operative environment

an extended training until 2000 epochs have been reached. A comparison on the same time window with Fig. 9 indicates that no strinking improvements have been obtained. Although RPROP is an efficient way to train simultaneously with multiple trajectories, the learning procedure seems to become slower to obtain more accurate results. Table VI provides the training time needed by the algorithms. From this table is clear how the LM optimization is much more heavy with respect to RPROP. For a smaller number of epochs (400 vs 600) the total duration is 20 times the one required by RPROP. However, the results obtained with LM were more accurated then those obtained by RPROP. The residual error in Table V obtained with LM is lower than that obtained by RPROP of one order of magnitude whereas the global NSSE is 3 orders of magnitude lower. This might suggest that the RPROP algorithm could be extended to more iteration numbers without the risk of over-fitting. The same measurement has not been possible for FT, because the memory requirements exceeded 26 GB, giving stronger support to our findings.

A marked difference can be noted between Tables V and VII. Training the neural network using simulated data allows to obtain very low NSSE value if compared with those obtained by means of flight test. Although NSSE coming from flight test is quite larger than expected, the residual between measured AOA and estimated AOA is always limited and bounded between some degrees. This difference should be attributed to the imperfection of the measurements. The Pitot-boom oscillations superimposed to the measurements and there are some good points to say that the AOA vane is quite subjected to these structural vibrations. Moreover, deflection of the boom itself due to aerodynamic field was evaluated around $\pm 2deg$. All these factors bring the authors to claim the virtual ability

TABLE VII
TRAINING PERFORMANCE OF [20], [20 20] AND [30 30] NN WITH TS COMING FROM FLIGHT TEST

NN	Max epochs	Mean training error [deg]	NSSE [deg^2/samples] Train	Test	Validation	Global
20	600	3.51e-1	2.165	2.154	2.156	2.162
20	2000	3.38e-1	2.051	2.044	2.044	2.049
20 20	600	4.21e-1	3.998	3.979	4.071	4.006
20 20	2000	3.38e-1	2.046	2.037	2.040	2.044
30 30	2000	2.59e-1	1.625	1.625	1.625	1.625

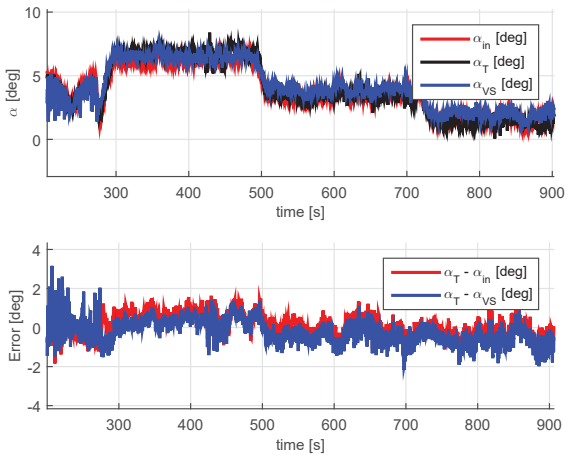

Fig. 11 Details of training trajectory of [20] NN using FT

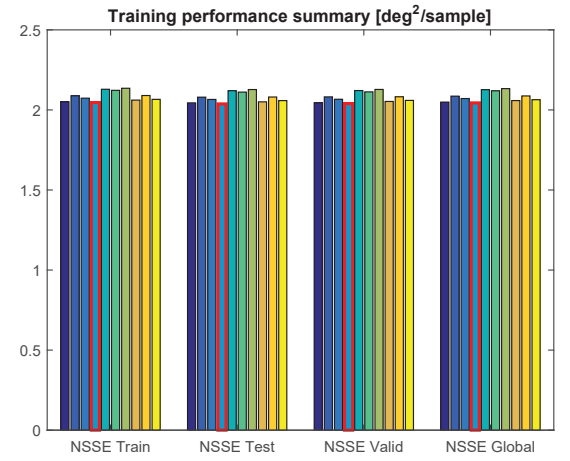

Fig. 12 NSSE over 10 training operations using [20] NN trained with 2000 epochs on FT

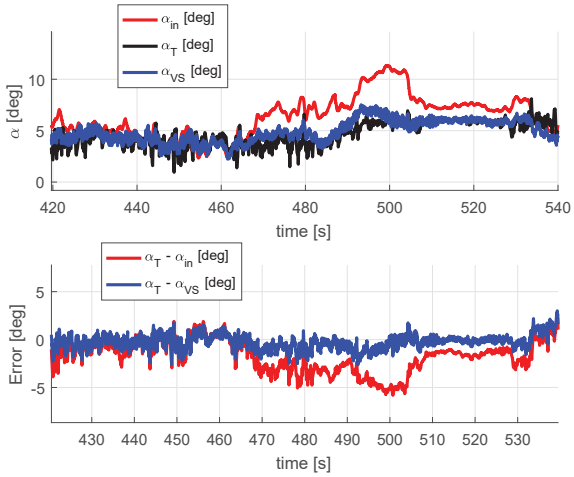

Fig. 13 Examples of non-linearity during training of [20 20] NN using FT

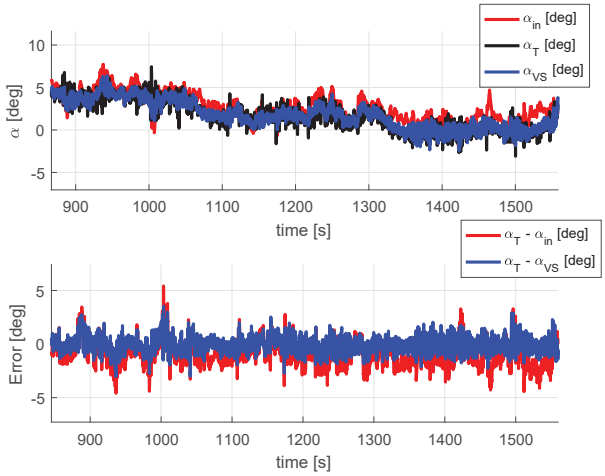

Fig. 14 Examples of non-linearity during test of [20 20] NN using FT

of the equipment to reach some fraction of degree of accuracy if the training set and input sensors are properly set. After the NN has been trained, the equipment has been tested with different input data to see how much it is able to generalize. In Table VIII the results in terms of mean residual error are shown. Although the test error obtained with a NN trained with RPROP seems to be slightly higher, the time needed for training is still considered a great advantage. In Table IX the test results in case of operative environment are depicted. They might be considered similar with those obtained in simulated environment. However, the quantity of cases available in a TS coming from a real FT are surely lower than those given by a set of simulations. This could be the reason for the higher test residual error.

TABLE VIII
MEAN RESIDUAL ERRORS ON SIMULATED ENVIRONMENT

NN	Opt. Alg.	Mean error [deg] Test #1	Test #2	Test #3	Test #4
20	LM	8.22e-2	6.26e-2	5.53e-2	1.86e-2
20	RPROP	1.195e-1	5.539e-1	6.131e-1	2.491e-1

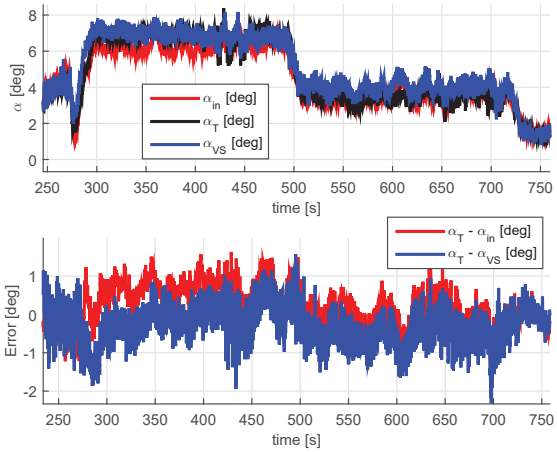

Fig. 15 Examples of non-linearity during training of [30 30] NN using FT

TABLE IX
MEAN RESIDUAL ERRORS ON OPERATIVE ENVIRONMENT

	Mean error [deg]		
NN	**FT #1**	**FT #2**	**FT #3**
20	2.166e-1	1.1863	1.1398
20 20	3.333e-1	5.076e-1	4.120e-1
30 30	1.147e-1	9.304e-1	1.1313

VI. CONCLUSION

Air data stays for a fundamental set of measurements needed by a lot of control systems. Current state-of-the-art sensors imply a lot of physical units and current redundancy multiplies those parts. Numerous studies have attempted to define an efficient and reliable virtual sensor for air data estimation. Some of them consist in model-based approaches where the main issue is to define the model parameters, whereas others are model-free approaches. Previous research frequently focused on FDI systems in order to propose analytical redundancy for current state-of-the-art sensors. This study focuses on the reasons why a NN-based ADS could be a possible substitution for current commercial ADAHRS, describing advantages over other techniques. This solution fits well with design requirements of small and medium UAVs, permitting to reach high integration of the main aircraft sensors. This innovative fusion technique can bring to an important simplification on a fundamental system as the ADAHRS. Simulations and operative scenario measurements have been applied to training and test procedures obtaining very good results. The training operations have been conducted with simultaneous trajectories showing improvements in final residuals. A discussion has been carried out on the optimization algorithm, which allowed to define pros and cons of the RPROP method over the most common LM. Additional studies should be conducted to define a suitable training set in order to reduce the final residual error. Validation of the equipment in operative environment will become objective of the following research.

ACKNOWLEDGMENT

The authors would like to thank Ing. Nando Groppo srl and Politecnico di Milano for their collaboration during the flight test campaign.

REFERENCES

[1] I. Samy, I. Postlethwaite, and D. W. Gu, Survey and application of sensor fault detection and isolation schemes, Control Eng. Pract., vol. 19, no. 7, pp. 658674, 2011.
[2] K. C. Wong, Aerospace industry opportunities in Australia-unmanned aerial vehicles (UAVs). Department of Aeronautical Engineering, University of Sydney, 2007.
[3] FAA Modernization and Reform Act (2012). H.R. 658, 112th Congress, 2nd Session.
[4] P. Freeman, P. Seiler, and G. J. Balas, Air data system fault modeling and detection, Control Eng. Pract., vol. 21, no. 10, pp. 12901301, 2013.
[5] J. Marzat, H. Piet-Lahanier, F. Damongeot, and E. Walter, Model-based fault diagnosis for aerospace systems: a survey, Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng., vol. 226, no. 10, pp. 13291360, 2012.
[6] M. Oosterom, R. Babuska, Virtual Sensor for the Angle-of-Attack Signal in Small Commercial Aircraft, 2006 IEEE International Conference on Fuzzy Systems, pp. 1396-1403, Sept 2006.
[7] G. Hardier, C. Seren, P. Ezerzere, Model Based Techniques for Virtual Sensing of Longitudinal Flight Parameters, International Journal of Applied Mathematics and Computer Science, vol. 25, no. 1, pp. 23-28, Mar 2015.
[8] A. Lerro, M. Battipede, P. Gili, "Sistema e procedimento di misura e valutazione di dati aria e inerziali", TO2013A000601, 2013.
[9] M. Battipede, P. Gili, A. Lerro, S. Caselle, P. Gianardi, "Development of Neural Networks for Air Data Estimation: Training of Neural Network Using Noise-Corrupted Data", 3rd CEAS Air & Space Conference, 21st AIDAA Congress, ISBN: 9788896427187, 2011.
[10] M. Battipede, M. Cassaro, P. Gili, A. Lerro, "Novel Neural Architecture for Air Data Angle Estimation", In: L. Iliadis, H. Papadopoulos, C. Jayne (eds) Engineering Applications of Neural Networks, EANN 2013, Communications in Computer and Information Science, vol 383, pp. 313-322, Springer, Berlin, Heidelberg, DOI: 10.1007/978-3-642-41013-0_32, Sept 2013.
[11] P. Gili, M. Battipede, A. Lerro, "Neural networks for air data estimation: test of neural network simulating real flight instruments", In: C. Jayne, S. Yue, L. Iliadis (eds) Engineering Applications of Neural Networks, EANN 2012, Communications in Computer and Information Science, vol 311, Springer, Berlin, Heidelberg, ISBN:978-364232908-1, DOI: 10.1007/978-3-642-32909-8_29, Sept 2012.
[12] A. Lerro, M. Battipede, P. Gili, A. Brandl, "Survey on a Neural Network for Non Linear Estimation of Aerodynamic Angles", accepted but not yet presented for Intelligent Systems Conference 2017, London, 2017.
[13] J. G. Attali and G. Pags, Approximations of Functions by a Multilayer Perceptron: a New Approach, Neural Networks, vol. 10, no. 6, pp. 10691081, Aug. 1997.
[14] J. L. Castro and C. J. Mantas, Neural networks with a continuous squashing function in the output are universal approximators, vol. 13, pp. 561563, 2000.
[15] C. M. Bishop, Neural networks for pattern recognition, J. Am. Stat. Assoc., vol. 92, 1995.
[16] K. Levenberg, A method for the solution of certain non-linear problems in least squares. Quarterly Journal of Applied Mathematics I I (2), 164-168, 1944.
[17] D. W. Marquardt, An algorithm for least-squares estimation of non-linear parameters. Journal of the Society of Industrial and Applied Mathematics 11 (2), 431-441, 1963.
[18] M. Riedmiller and H. Braun, A direct adaptive method for faster backpropagation learning: The RPROP algorithm, in IEEE International Conference on Neural Networks - Conference Proceedings, 1993.
[19] . Kisi and E. Uncuoglu, Comparison of three back-propagation training algorithm for two case study, Indian J. Eng. Mater. Sci., vol. 12, no. October, pp. 434442, 2005.
[20] A. R. Webb, D. Lowe, and M. D. Bedworth, A comparison of non-linear optimisation strategies for feed-forward adaptive layered networks. RSRE Memorandum 4157, Royal Signals and Radar Establishment, St Andrew's Road, Malvern, UK, 1988.
[21] EASA Airworthiness Directive AD No.: 2013-0068, 29 March 2013.
[22] EASA Airworthiness Directive AD No.: 2015-0135, 15 July 2015.

[23] T. Golly, D. M. Holm, "Magnetic Angle of Attack Sensor", WO 01/77622 A2, 2001.
[24] G. A. Seidel, D. J. Cronin, J. H. Mette, M. R. Koosmann, J. A. Schmitz, J. R. Fedele, D. A. Kromer, "Multi-function Air Data Sensing Probe Having an Angle of Attack Vane", US 6941805 B2, 2005.
[25] D. H. Lenschow, Vanes for Sensing Incidence Angles of the Air from an Aircraft, J. Appl. Meteorol., vol. 10, no. 6, pp. 13391343, Dec. 1971.
[26] UTC Aerospace Systems, Outside Air Temperature (OAT) Sensor Series 0129, Burnsville, USA.
[27] UTC Aerospace Systems, Angle of Attack (AOA) Sensors, Burnsville, USA.
[28] Aerosonic Corporation, Sensors, Clearwater, USA.
[29] AMETEK Aerospace, Angle of Attack Transducer, Wilmington, USA.
[30] AMETEK Aerospace, Aircraft Sensors and Systems Total Air Probe, Wilmington, USA.
[31] SpaceAge Control, State-of-the-Art Air Data Products Solution Guide, Palmdale, USA.
[32] W. Denson, G. Chandler, W. Crowell, A. Clark, & P. Jaworski, Nonelectronic Parts Reliability Data 1991 (No. RAC-NPRD-91). Reliability Analysis Center Griffiss AFB NY, 1991.
[33] S. Chiesa, S. C. Aleina, G. A. Di Meo, R. Fusaro, N. Viola Autonomous Take-off and Landing for Unmanned Aircraft System: Risk and Safety Analysis, 29th Congress of the International Council of the Aeronautical Sciences, ICAS 2014.
[34] F. De Vivo, M. Battipede, P. Gili, A. Brandl, "Ill-conditioned problems improvement adapting Joseph covariance formula to non-linear Bayesian filters". WSEAS Trans. Electron. 7, 1825, DOI:10.13140/RG.2.1.3027.0960, 2016.
[35] F. De Vivo, A. Brandl, M. Battipede, and P. Gili, Joseph covariance formula adaptation to Square-Root Sigma-Point Kalman filters, DOI: 10.1007/s11071-017-3356-x, Nonlinear Dynamics, Jan. 2017.
[36] G. Hardier, C. Seren, P. Ezerzere and G. Puyou, "Aerodynamic Model Inversion for Virtual Sensing of Longitudinal Flight Parameters", 2013 Conference on Control and Fault-Tolerant Systems (SysTol), pp. 140-145, Oct 2013.
[37] T. Rajkumar, J. Bardina, "Prediction of Aerodynamic Coefficients using Neural Networks for Sparse Data", FLAIRS Conference, pp. 242-246, 2002.
[38] T. Rajkumar, J. Bardina, "Training data requirement for a neural network to predict aerodynamic coefficients", Independent Component Analyses, Wavelets, and Neural Networks, vol. 5102, pp.92-103, Apr 2003.
[39] P. A. Samara, G. N. Fouskitakis, J. S. Sakellariou, & S. D. Fassois, "Aircraft angle-of-attack virtual sensor design via a functional pooling NARX methodology". European Control Conference (ECC), 2003, pp. 1816-1821, Sept 2003.
[40] D. Kriesel, A Brief Introduction to Neural Networks, http://www.dkriesel.com, 2007.