JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 32, No. 4, July–August 2009

Active Flutter Suppression for a Three-Surface Transport Aircraft by Recurrent Neural Networks

Mattia Mattaboni,* Giuseppe Quaranta,† and Paolo Mantegazza‡
Politecnico di Milano, 20156 Milano, Italy

DOI: 10.2514/1.40774

This paper presents an effective approach for the design of a flutter-suppression system by means of recurrent neural networks. This system is used to move flutter instabilities outside the flight envelope of an unconventional three-surface transport aircraft. The design process requires a comprehensive aircraft model in which flight mechanics, structural dynamics, unsteady aerodynamics, and control-surface actuators are represented in state-space form, according to the modern aeroelastic approach. The implemented regulator is based on two recurrent neural networks: one is trained to identify the system dynamics and the other acts as a controller using an indirect inversion of the identified model. Keeping the training of both recurrent networks online leads to an adaptive control system. Extensive numerical tests are used to tune the neural network design parameters and to show how the neural controller increases system damping, widening the flutter-free flight envelope by more than 15% of the uncontrolled flutter velocity.

Nomenclature

$\boldsymbol{C}$	=	generalized damping matrix
c	=	reference chord, m
E	=	quadratic error
f	=	neuron activation function
$\boldsymbol{f}_a$	=	generalized aerodynamic forces due to the structural deformation
H_M	=	servo dynamic compliance
H_N	=	servo transfer function
$\boldsymbol{H}_{\mathrm{am}}$	=	generalized aerodynamic forces frequency response matrix
$\boldsymbol{H}_{\mathrm{pr}}$	=	proper part of the aerodynamic transfer function
$\boldsymbol{K}$	=	generalized stiffness matrix
K	=	flight mechanics controller proportional gains
k	=	reduced frequency $\omega c/2V_\infty$
$\boldsymbol{M}$	=	generalized mass matrix
M	=	Mach number
M_h	=	hinge moment
N_C	=	number of movable surfaces
N_M	=	number of measures
N_{in}	=	number of network inputs
N_{neuron}	=	number of network neurons
p	=	nondimensional Laplace variable $sc/2V_\infty$
$\boldsymbol{q}$	=	modal degrees of freedom
q	=	dynamic pressure, pa
s	=	Laplace variable
u	=	network input
V_∞	=	asymptotic velocity, m/s
v	=	neuron internal activity
$\boldsymbol{v}_g$	=	gust velocity, m/s
W_{ji}	=	network synaptic weights
$\boldsymbol{x}$	=	dynamic states
$\boldsymbol{y}$	=	state-space system output
y	=	neuron output
δ	=	movable surface deflection, rad
δ^C	=	servo command input
δ_{ij}	=	Kronecker symbol
η	=	learning rate
ρ	=	control penalization coefficient

Subscripts

a	=	aerodynamic states
B	=	states in the body-fixed reference
g	=	gust related quantity
I	=	states in the inertial reference
q	=	pitch rate
θ	=	pitch angle

Superscripts

B	=	body-fixed reference
CON	=	control network
D	=	desired signal
I	=	inertial reference
ID	=	identification network
M	=	measures

I. Introduction

FLUTTER often can be a critical problem for current aircraft design, due to the structure lightness, which leads to high structural compliance. Several approaches may be followed to solve this problem. The first, which may be denominated as passive, basically goes through a redesign of the aircraft to increase its stability boundaries. The second, which may be called active, tries to exploit the capabilities of control systems to improve stability properties, usually with a smaller weight increment if compared with the former approach. Furthermore, such controllers can be used to improve the performances and the cruise comfort.

In this paper, an active control strategy based on recurrent neural networks (RNNs) for flutter suppression is designed to improve the stability boundaries and performances of a transport aircraft with an unconventional configuration denominated X-DIA. The X-DIA, sketched in Fig. 1, is a conceptual design of a short-range 70-seat jet liner based on three main lifting surfaces: a front canard, a 15 deg forward swept wing, and a T-tail. The three-surface configuration is

Presented as Paper 2136 at the 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Honolulu, HI, 23–26 April 2007; received 3 September 2008; revision received 30 April 2009; accepted for publication 4 May 2009. Copyright © 2009 by Politecnico di Milano. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0731-5090/09 and $10.00 in correspondence with the CCC.

*Ph.D. Candidate, Dipartimento di Ingegneria Aerospaziale, via La Masa 34.

†Research Fellow, Dipartimento di Ingegneria Aerospaziale, via La Masa 34.

‡Professor, Dipartimento di Ingegneria Aerospaziale, via La Masa 34.

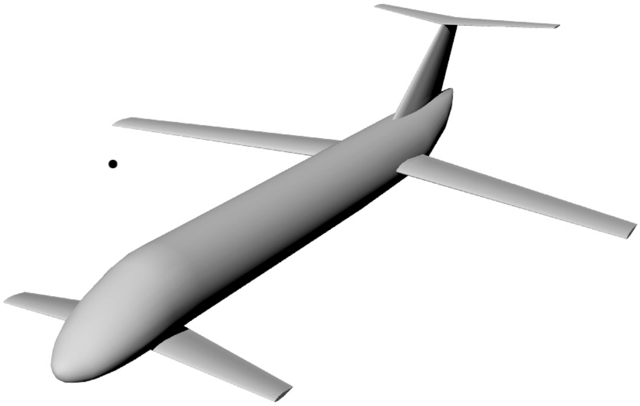


Fig. 1 X-DIA concept configuration.

expected to improve the aerodynamic efficiency of the aircraft [1–3], allowing a minimum induced-drag trim point for all center-of-gravity locations. The rear location of the main wing along the fuselage, allowed by the forward swept wing coupled with the canard surfaces, yields to a positive cooperation between structures and aerodynamics, allowing significant weight savings. The stability, from the flight mechanics and aeroelastic point of view, may be a problem and requires a thorough investigation [3,4]. However, the three-surface configuration is expected to mitigate the reduction of static margin of canard configurations [1].

This configuration is currently the object of numerous investigations at the Dipartimento di Ingegneria Aerospaziale of Politecnico di Milano (DIAPM), both numerical and experimental [5–7]. A $1/10$- Froude-scaled, wind-tunnel model, shown in Fig. 2, has been built and tests have been already carried out on it. Two different all-movable canard planforms have been evaluated for the reduction of fuselage vibrations [7]; a wing aeroelastic control has been implemented and tested [8]. Because several configurations are currently under investigation for the wind-tunnel model, it has been chosen to refer, as a proof of concept, to the full-scale aircraft model for the controller design.

Neural networks (NNs) are currently widely used for processing and identification of signals and for control system design [9,10]. Usually, NNs used for identification are static when the parameters of the model are fixed and it is not necessary to have online learning. However, for aeroelastic control, the use of an adaptive controller seems the most appropriate choice because of largely varying flight conditions, which means different dynamic pressures, Mach numbers, mass configurations, and so on [11]. Static NNs have been used for active adaptive aeroelastic control by Scott and Pado [12] on the benchmark active controls technology wind-tunnel model using a single-input/single-output architecture. Different control strategies were tested to obtain adaptivity: a gain scheduled control law, using Mach number and dynamic pressure as input, and an inverse model control based on a static NN with the input to the network made by the current and delayed values of sensor signals.

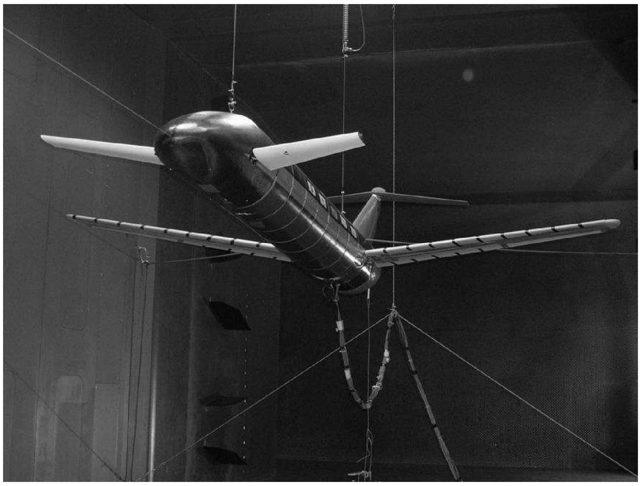


Fig. 2 X-DIA wind-tunnel model.

However, to cope with time variations in dynamics, it is necessary to *self-tune* the controller, adjusting automatically to the current operating conditions. To do so, it is necessary to continuously update the synaptic weights (i.e., the parameters of the network). This capability can be achieved by using RNNs, which add feedback connections to classical NNs by creating internal states that develop internal dynamics. This approach gives rise to a real active learning system [13,14]. The application of RNNs to aeroelastic control has already been pursued in previous works [15,16]. However, this time, the approach is applied to a complete free-flying aircraft.

The basic system is made of two RNNs: one dedicated to the identification of the system using the information collected by few accelerometers on the aircraft and one dedicated to the control by an indirect inversion of the identified model. Because aircraft are nonminimum-phase dynamic systems, a penalty function is introduced in the objective function to avoid the divergence of network weights. The block obtained by grouping identification and control networks works as a sort of black box, which can be designed without specific information on the system to be governed. However, to choose the best network architecture and topology, a numerical test campaign, such as the one here presented, is needed. The RNN training phase is conducted by means of time-domain numerical simulations, and so a state-space model of the entire system based on a finite state rational approximation of unsteady aerodynamic loads is required.

The paper is organized as follows. The first section is dedicated to the description of the arrangement of the system including the movable surfaces used to control and the presentation of the comprehensive aeroservoelastic model built. The second section investigates the stability properties of the model and the introduction of a simple proportional controller to stabilize the flight mechanics modes. The third section presents the architecture of the RNN control system in detail, and the fourth section shows the numerical results obtained by applying such a system to the aircraft under investigation.

II. Aeroservoelastic Model

For this assessment of the RNN controller, only instabilities related to symmetric modes have been taken into account. The flutter controller must be designed relying on the following aerodynamic surfaces already included in the wind-tunnel model: 1) the all-movable canard surfaces; 2) four surfaces on the main wing, two on the trailing edge, and two on the leading edge (see Fig. 3); and 3) the elevators.

The structural model of the X-DIA is based on a finite element (FE) stick model built using Nastran. The unsteady aerodynamic loads are obtained using a doublet-lattice model (DLM) for the aircraft lifting surfaces [17], coupled to interference bodies for the fuselage (see Fig. 4). Because only symmetric modes are considered, it has not been necessary to apply any correction for in-plane movements of the T-tail [18].

To design the RNN controller, it is necessary to build a comprehensive, time-domain aeroservoelastic model of the X-DIA aircraft concept. Such a model must include a detailed description of the following items: 1) dynamics of the aircraft reference frame

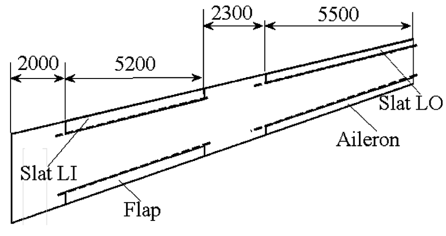


Fig. 3 Control surfaces on the main wing; measures are in millimeters.

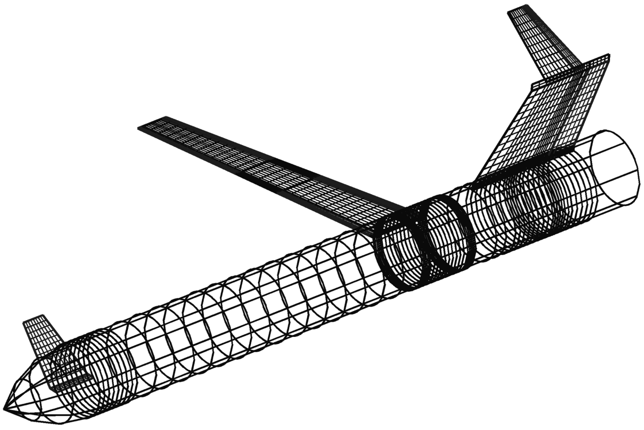


Fig. 4 Aerodynamic DLM model grid.

(i.e., flight mechanics), 2) structural dynamics, 3) unsteady aerodynamics, and 4) control-surface actuator dynamics.

Using these data, two aeroservoelastic models are defined: a *design* model and a *verification* model, the latter using a larger modal basis to represent structural dynamics. Of course, both must be able to correctly represent the aircraft flutter behavior. The first model is used in the control system design, when many numerical simulations are performed to tune all control parameters. The larger and more accurate verification model is used to clear the design from possible spillover effects.

A. Aeroservoelastic State-Space Modeling

The FE model is used to compute the symmetric modes with the lowest frequencies and includes the rigid-body modes. These modes are used as input to the integral DLM aerodynamics. The results are the unsteady aerodynamic loads due to harmonic boundary oscillations of the structural modes, which lead to the generalized aerodynamic forces (GAFs) frequency response matrix $\boldsymbol{H}_{\mathrm{am}}(k, M)$. With this aerodynamic formulation (DLM plus interference bodies), it is not possible to obtain the speed derivatives along the flight direction, which are essential to represent the aircraft phugoid mode. However, such a contribution can be properly added to the $\boldsymbol{H}_{\mathrm{am}}$ using data gathered from a flight mechanics study of the aircraft and following the indications included in [19]. As a result, a frequency-domain aeroelastic model is obtained:

$$(s^2\boldsymbol{M} + s\boldsymbol{C} + \boldsymbol{K} - q\boldsymbol{H}_{\mathrm{am}}(k, M))\boldsymbol{q} = \boldsymbol{0} \tag{1}$$

To implement a simulator for the synthesis of the neural controller, Eq. (1) must be transformed into an appropriate time-domain system. For this reason, it is necessary to build a finite state approximation [20] of the GAFs transfer matrix $\boldsymbol{H}_{\mathrm{am}}$. A simple rational approximation can be obtained using Roger's approach [21], which states that $\boldsymbol{H}_{\mathrm{am}}$ can be approximated as

$$\boldsymbol{H}_{\mathrm{am}}(p) = \boldsymbol{D}_0 + \boldsymbol{D}_1 p + \boldsymbol{D}_2 p^2 + \sum_{i=0}^{m} \frac{p}{p + \beta_i}\boldsymbol{E}_i \tag{2}$$

The system poles included in Eq. (2), $-\beta_i$, can be chosen arbitrarily by the user in the range of interest of the reduced frequency k and must be real and negative to ensure asymptotic stability of the aerodynamic subsystem. All coefficients of the matrices in Eq. (2) are evaluated using a least-squares technique on each individual term. The resulting model for the GAFs has a dimension equal to $m \times n$, where n is the number of structural modes, and m is called the order of the rational approximation. Roger's technique is straightforward, robust, easy to implement, and inexpensive in terms of computational burden. However, an increase in the number of states should be expected when high accuracy is required. System poles $-\beta_i$ are assigned as evenly distributed points in the range of reduced frequencies for which the $\boldsymbol{H}_{\mathrm{am}}$ matrix is known. When this range is large, further improvements in the fitting of $\boldsymbol{H}_{\mathrm{am}}$ coefficients may be obtained by running a simple optimization on pole positioning. In any case, given a large Roger's approximation of the GAFs transfer matrix, it is possible to recover a smaller model adopting classical model-reduction techniques, such as the balanced-truncation approach [22]. In this way, the user does not have to a priori decide the dimension of the finite state approximation, because the right system dimension is identified automatically through the balanced-truncation process. For more details on such a technique, see [7]. The same identification process can be applied to gust GAFs [23]. As a result, a state-space model for GAFs is obtained,

$$\begin{aligned}\frac{c}{2V_\infty}\dot{\boldsymbol{x}}_a &= \boldsymbol{A}_a\boldsymbol{x}_a + \boldsymbol{B}_a\boldsymbol{q} + \boldsymbol{B}_{ag}\frac{\boldsymbol{v}_g}{V_\infty}\\ \boldsymbol{f}_a + \boldsymbol{f}_g &= \boldsymbol{C}_a\boldsymbol{x}_a + \boldsymbol{D}_0\boldsymbol{q} + \frac{c}{2V_\infty}\boldsymbol{D}_1\dot{\boldsymbol{q}} + \left(\frac{c}{2V_\infty}\right)^2\boldsymbol{D}_2\ddot{\boldsymbol{q}}\\ &\quad + \boldsymbol{D}_{0g}\frac{\boldsymbol{v}_g}{V_\infty} + \frac{c}{2V_\infty}\boldsymbol{D}_{1g}\frac{\dot{\boldsymbol{v}}_g}{V_\infty} + \left(\frac{c}{2V_\infty}\right)^2\boldsymbol{D}_{2g}\frac{\ddot{\boldsymbol{v}}_g}{V_\infty}\end{aligned} \tag{3}$$

which can be connected to the structural model

$$\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{C}\dot{\boldsymbol{q}} + \boldsymbol{K}\boldsymbol{q} = q(\boldsymbol{f}_a + \boldsymbol{f}_g) \tag{4}$$

to build the aeroelastic aircraft model in the time domain.

1. Servo-Actuators

The usual assumption of infinitely stiff and infinitely powerful linear servo-actuators, able to drive the position of the control surface exactly, may not be adequate for aeroservoelastic analyses. It is usually necessary to add the dynamics of actuators to represent the correct bandwidth of the surface position servo and the actuator dynamic compliance, which may affect the whole system aeroelastic behavior. In the most general case, the dynamics of an actuator for a movable surface can be represented in the frequency domain by the following expression [24]:

$$\delta = H_N(s)\delta^C + H_M(s)M_h \tag{5}$$

H_N and H_M are, respectively, the transfer functions of the servo and its dynamic compliance. The servo, driven by the signal δ^C, supplies a moment that acts on the hinge. To connect in feedback the servo-actuator model to the aeroelastic system, it is necessary to find the relation that gives the hinge moment as a function of δ and the command signal:

$$M_h = \frac{1}{H_M(s)}(\delta - H_N(s)\delta^C) \tag{6}$$

Equation (6) can be easily transformed in a state-space model only if the inverse of the dynamic compliance [i.e., the dynamic stiffness $1/H_M(s)$] is a proper transfer function. For this reason, the dynamic compliance must be a proper, but not strictly proper, transfer function; that is, the degree of the polynomial denominator must be equal to the degree of the numerator. This condition is usually true if only the low-frequency behavior is considered [25]. If an electrohydraulic position control servo is considered, the closed-loop dynamic stiffness presents a numerator with a degree 2 times higher than the denominator [24], whereas the transfer function $M_h/\delta_c = H_N(s)/H_M(s)$ remains strictly proper [24]. In this case, a second-order residualization, similar to the one already used for the generalized aerodynamic forces, can be employed for dynamic compliance without breaking the causality of the overall system. This is possible because the control-surface deflections, for which the dynamics are naturally second order in time, are already included in the structural model. So the dynamic stiffness can be represented as the sum of a proper part plus the terms associated with the derivatives of the movable surface deflection:

$$\frac{1}{H_M(s)} = \frac{N_M^p(s)}{D_M^p(s)}\delta + sD_{1M}\delta + s^2D_{2M}\delta \tag{7}$$

Equation (6) can be easily transformed into a time-domain system [26] considering, in general, four inputs (δ, $\dot{\delta}$, $\ddot{\delta}$, and δ^C):

$$\begin{aligned}\dot{\boldsymbol{x}}_{\text{act}} &= \boldsymbol{A}\boldsymbol{x}_{\text{act}} + \boldsymbol{B}_{\delta}\delta + \boldsymbol{B}_{\delta^C}\delta^C\\ M_h &= \boldsymbol{C}\boldsymbol{x}_{\text{act}} + D\delta + D_{1M}\dot{\delta} + D_{2M}\ddot{\delta}\end{aligned} \tag{8}$$

Of course, the generalized degrees of freedom related to control-surface deflections must be included into the structural model and the related GAFs must be computed using the DLM.

The final result is a state-space system that represents the aeroservoelastic dynamics by a linear dynamic system of equations:

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{\delta}^C + \boldsymbol{B}_g\boldsymbol{v}_g \qquad \boldsymbol{y}^M = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{\delta}^C + \boldsymbol{D}_g\boldsymbol{v}_g \tag{9}$$

where the vector $\boldsymbol{x}$ is composed of the structural degrees of freedom $\boldsymbol{q}$ and their first derivative in time $\dot{\boldsymbol{q}}$, the control-surface rotations $\boldsymbol{\delta}$ and $\dot{\boldsymbol{\delta}}$, the aerodynamic internal states $\boldsymbol{x}_a$, and all the states of each actuator dynamic system $\boldsymbol{x}_{\text{act}}$.

2. *Inertial-to-Body-Fixed Transformation*

Following the flight mechanics approach, the reference system chosen to represent the rigid dynamics of the aircraft (i.e., the dynamics of rigid-body modes) should be a body reference frame. The state-space model obtained using Nastran modal data refers to an inertial reference frame instead. Thus, a transformation between the two reference frames is required. The inertial-to-body-fixed transformation is performed using the procedure described by Lind and Brenner [27] for small perturbations around a reference condition restricted to symmetric modes.

The only degrees of freedom involved in such transformation are the displacements and rotations of the aircraft center of mass and their derivatives. The small perturbation vector expressed in the inertial reference is

$$\boldsymbol{x}_I = \{x \quad z \quad \theta \quad \dot{x} \quad \dot{z} \quad \dot{\theta}\}^T \tag{10}$$

and the same vector in the body reference frame is

$$\boldsymbol{x}_B = \{\acute{x} \quad \acute{z} \quad \acute{\theta} \quad u \quad w \quad q\}^T \tag{11}$$

using classical flight mechanics notation [28]. The transformation from the body reference to the inertial one is

$$\boldsymbol{x}_I = \boldsymbol{T}_1\boldsymbol{x}_B \qquad \dot{\boldsymbol{x}}_I = \boldsymbol{T}_2\dot{\boldsymbol{x}}_B + \boldsymbol{T}_3\boldsymbol{x}_B \tag{12}$$

where the exact definition of matrices $\boldsymbol{T}_1$, $\boldsymbol{T}_2$, and $\boldsymbol{T}_3$ can be found in [27]. By properly expanding the transformation matrices on all the degrees of freedom, it is possible to obtain an aeroservoelastic model in which only the effective rigid degrees of freedom (i.e., $\acute{\theta}$, u, w, and q) are included. This transformation is applied to the state-space aeroservoelastic model (9) to obtain the corresponding system in the body reference:

$$\dot{\boldsymbol{x}}_B = \boldsymbol{A}^B\boldsymbol{x}_B + \boldsymbol{B}^B\boldsymbol{\delta}^C \tag{13}$$

where

$$\boldsymbol{A}^B = \boldsymbol{T}_2^{-1}(\boldsymbol{A}^I\boldsymbol{T}_1 - \boldsymbol{T}_3) \qquad \boldsymbol{B}^B = \boldsymbol{T}_2^{-1}\boldsymbol{B}^I \tag{14}$$

The signals measured by the accelerometers on the aircraft are also in the body reference frame, and so a transformation must also be applied to the output equations.

The direct application of the inertial to body transformation may lead to an incorrect results for the aircraft rigid dynamics. It may happen that the system poles that correspond to the short period and the phugoid mode undergo modifications after applying such a transformation, whereas they should remain unchanged.

Looking in depth into the applied transformations, it appears that the error must be due to missing coefficient cancellations in the part of the $\boldsymbol{A}^B$ matrix of the state-space model of Eq. (13) associated with rigid modes. To ensure that the resulting model is correct, it is necessary to add a few constraints to the aerodynamic state-space realization to force the cancellation of the coefficients necessary to obtain the correct rigid dynamics in the body reference. Applying Eq. (12) in level flight, the rigid mode's GAFs transfer function becomes

$$\begin{aligned}\boldsymbol{H}_{\text{am}}^B(p) &= \boldsymbol{C}_a(p\boldsymbol{I} - \boldsymbol{A}_a)^{-1}\boldsymbol{B}_a + \boldsymbol{D}_0 + p\boldsymbol{D}_1 + p^2\boldsymbol{D}_2\\ &\quad + \frac{c}{2V_\infty}\boldsymbol{D}_1\boldsymbol{T} + p\frac{c}{2V_\infty}\boldsymbol{D}_2\boldsymbol{T} = \boldsymbol{H}_{\text{am}}^I(p) + \frac{c}{2V_\infty}\boldsymbol{D}_1\boldsymbol{T}\\ &\quad + p\frac{c}{2V_\infty}\boldsymbol{D}_2\boldsymbol{T}\end{aligned} \tag{15}$$

where T is equal to

$$\boldsymbol{T} = \begin{bmatrix} 0 & 0 & 0\\ 0 & 0 & -V_\infty\\ 0 & 0 & 0\end{bmatrix} \tag{16}$$

The transfer function in the body reference system is therefore the sum of that in the inertial reference system plus two terms related to the nonproper part of the same transfer function. Using the *final value theorem*, the asymptotic value of GAFs due to a step input in the body-reference-frame pitch rotation $\acute{\theta}$ can be computed as

$$\begin{aligned}\lim_{t\to\infty}\boldsymbol{f}_a(t) &= \lim_{p\to 0}\boldsymbol{H}_{\text{am}}^B(p)\{0 \quad 0 \quad 1\}^T\\ &= \left(\boldsymbol{H}_{\text{am}}^I(0) + \frac{c}{2V_\infty}\boldsymbol{D}_1\boldsymbol{T}\right)\{0 \quad 0 \quad 1\}^T\\ &= \left(\boldsymbol{H}_{\text{am}}^I(0) + \frac{c}{2V_\infty}\begin{bmatrix} 0 & 0 & -V_\infty D_1(x,z)\\ 0 & 0 & -V_\infty D_1(z,z)\\ 0 & 0 & -V_\infty D_1(\theta,z)\end{bmatrix}\right)\begin{Bmatrix}0\\0\\1\end{Bmatrix}\end{aligned} \tag{17}$$

which must be zero, because the corresponding variation of angle of attack is null. As a consequence, the first constraint to be imposed is

$$D_1(*,z) = \frac{2\boldsymbol{H}_{\text{am}}^I(0)(*,\theta)}{c} \tag{18}$$

where the symbol $*$ is used to indicate all rigid-mode rows. If the system is forced by a vertical ramp input in the inertial reference frame

$$z(t) = wt$$

the asymptotic value of the aerodynamic generalized force is equal to

$$\begin{aligned}\lim_{t\to\infty}\boldsymbol{f}_a(t) &= \lim_{s\to 0}\frac{\boldsymbol{H}_{\text{am}}^I(s)}{s}\{0 \quad w \quad 0\}^T\\ &= w\frac{c}{2V_\infty}\lim_{p\to 0}\frac{\boldsymbol{H}_{\text{am}}^I(p)(*,z)}{p}\\ &= w\frac{c}{2V_\infty}\lim_{p\to 0}\left(\frac{\boldsymbol{H}_{\text{pr}}^I(p)(*,z)}{p}\right) + w\frac{c}{2V_\infty}D_1(*,z)\end{aligned} \tag{19}$$

where $\boldsymbol{H}_{\text{Pr}}^I$ is the proper part of the aerodynamic transfer function: that is,

$$\boldsymbol{H}_{\text{Pr}}^I = \boldsymbol{C}_a(p\boldsymbol{I} - \boldsymbol{A}_a)^{-1}\boldsymbol{B}_a + \boldsymbol{D}_0 \tag{20}$$

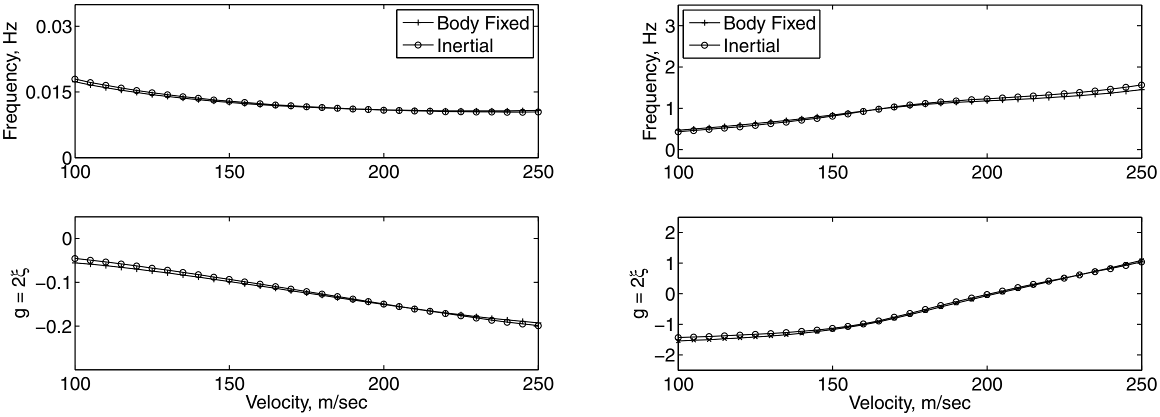
The z ramp input is equivalent to a step variation of the angle of attack $\alpha = w/V_\infty$, which corresponds to a variation of θ. In the inertial reference frame, it is straightforward to show that

$$\lim_{t\to\infty}\boldsymbol{f}_a(t) = \lim_{p\to 0}\boldsymbol{H}_{\text{am}}^I(p)(*,\theta)\alpha = \boldsymbol{H}_{\text{am}}^I(0)(*,\theta)\frac{w}{V_\infty} \tag{21}$$

Considering Eqs. (18), (19), and (21) together, it follows that

$$\lim_{p\to 0}\frac{\boldsymbol{H}_{\text{pr}}^I(0)(*,z)}{p} = 0 \tag{22}$$

This means that the proper part of the aerodynamic transfer function associated with the z translation input must have two zeros placed at



a) Phugoid **b) Short period**

Fig. 5 Comparison between rigid system roots in the inertial frame and in the body frame.

the origin of the complex plane. By imposing these two constraints. (18) and (22), the system poles remain practically unchanged after the reference system transformation, as shown in Fig. 5.

B. Analysis of the Configuration Without the Aeroelastic Controller

In this section, the results of some preliminary analyses are presented to illustrate the aeroelastic behavior of the uncontrolled symmetric X-DIA model.

1. Flight Mechanics

A simple flight mechanics control system is designed to augment aircraft rigid-mode damping. Thanks to the frequency separation between the first elastic mode and the rigid modes, it is possible to split up the control system in two distinct parts: one designed to control the aircraft flight dynamics and another designed to control its deformable dynamics. Hence, the flight mechanics control system is conceived as an inner loop independent of the aeroelastic control system, as shown in the block diagram of Fig. 6. Clearly, it must be ensured that the two control loops do not interact adversely, leading to a reduction of the aircraft stability.

For the flight mechanics, a simple proportional control law on the canards and elevators is applied [28,29]. The surface deflections are driven by the aircraft pitch and pitch rate, which are supposed to be measured by a vertical gyro and a rate gyro, respectively. The following two simple relations are added to the aeroservoelastic state-space model of the aircraft:

$$\delta^{\mathrm{can}} = K_{\theta}^{\mathrm{can}}\theta + K_{q}^{\mathrm{can}}q \qquad \delta^{\mathrm{eq}} = K_{\theta}^{\mathrm{eq}}\theta + K_{q}^{\mathrm{eq}}q \tag{23}$$

To adapt the controller to changing flight conditions, a gain scheduling strategy is implemented; the control law gains are determined for several design flight conditions and then linearly interpolated at intermediate points. The gains are calculated using a rigid model of the aircraft to augment the damping of the rigid modes without changing their frequency significantly. The obtained results are shown in Fig. 7, in which the root locus of the rigid modes at different flight velocities is presented. Because the short-period mode is already adequately damped (see the lines tagged as "control off" in Fig. 7b), the control system mainly increases the phugoid damping.

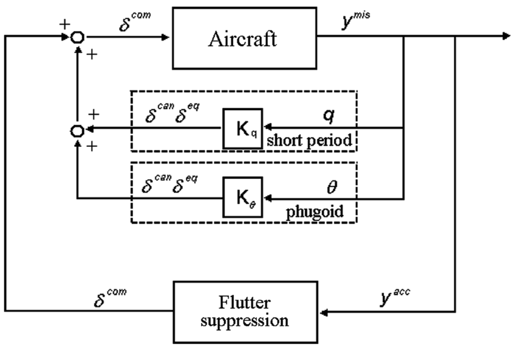
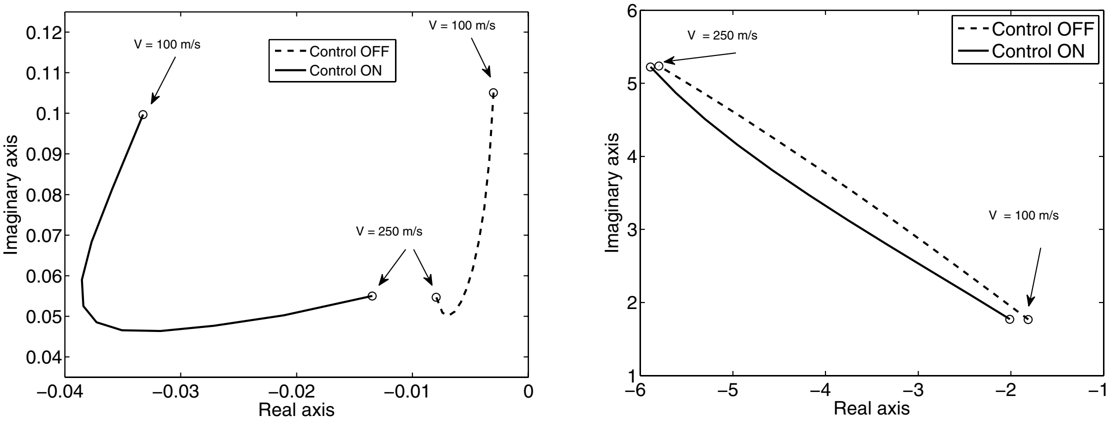


Fig. 6 Split configuration of the aircraft control systems.



a) Phugoid **b) Short period**

Fig. 7 Root locus of rigid modes with and without the flight mechanics controller.

2. Aeroelastic Behavior

Figure 8 depicts the diagrams of the evolution of structural roots, in terms of frequency and g damping, at different flight velocities. The plotted data refer to the aircraft model with no active aeroelastic control. In this configuration, the aircraft presents three different flutters in a very small velocity range (i.e., between 195.5 and 203.5 m/s):

1) The first flutter at 195.5 m/s is a classical bending-torsion flutter of the wing, as shown in Fig. 9a.

2) The second flutter at 195.5 m/s is a bending-torsion flutter of the horizontal tail plane, as shown in Fig. 9b.

3) The third flutter at 203.5 m/s is characterized by a coupling between the rigid-body pitch rotation (i.e., the short-period mode) and the first wing bending mode, as shown in Fig. 9c; this is a peculiar flutter mechanism related to the X-DIA negative-swept wing configuration, already reported on another aircraft with a somewhat similar architecture: the X-29A.

The flutter-suppression task is particularly complex, both because of the closeness of the three flutter velocities and because of the high tail flexibility, as demonstrated by the flutter modal shapes shown in Fig. 9b. Moreover, the tail bending-torsion flutter is more difficult to stabilize than the wing flutter, because there is a reduced controllability of the related unstable mode through the elevators. As a consequence, it has been decided to try to move this flutter at an higher velocity by stiffening the stabilizer structure, as shown in the Results section.

3. Actuators and Sensors

The flutter-suppression control system uses accelerometers as input. The accelerometers setup is presented in Fig. 10; positions are chosen, according to preliminary flutter analyses, so that a good observability of all flutter modes is obtained. The signals are high-pass-filtered (with a cut frequency of 1 Hz) to avoid interactions between aeroelastic and aircraft flight dynamic controllers, then low-pass-filtered (with a cut frequency of 50 Hz) to reduce noise and avoid aliasing. Both high- and low-pass filters are based on a second-order Butterworth frequency shaping.

The characteristics of the movable surface actuators have been inferred by an appropriate scaling of their counterparts used on the X-DIA wind-tunnel model. They are driven by a force proportional–integral–derivative controller on the surface position error [8]:

$$M_h = \mathrm{PID}(s)(\delta^C - \delta) \tag{24}$$

III. Control Methodology

The aeroelastic active control implemented for flutter suppression is an indirect adaptive control system made up by an *identifier*, needed to collect information on the system from the measures of its input and output, and a *controller*, which exploits the identifier to stabilize the aircraft. The system architecture is shown in the block diagram of Fig. 11. Such a control strategy is suitable when a black-box approach is chosen, because all information on the system under control is inferred by the identifier without any prior knowledge of the system. The identifier and the controller are realized by means of two different RNNs: the former network takes the name of *neuroidentifier*, and the latter is a *neurocontroller*. Figure 12 shows the structure of a single-layer RNN. The jth neuron output at the kth time step is calculated as

$$y_j(k) = f(v_j(k)) \tag{25}$$

where the neuron internal activity is equal to

$$v_j(k) = \sum_{i=1}^{N_{\mathrm{in}}} W_{ji}(k)u_i(k) \tag{26}$$

In Eq. (26), N_{in} is the number of network input u. Among the network input there is a constant value set to -1, named *bias*, which is used to shift the origin of their activity value. In this application, all neurons use the hyperbolic tangent activation function; that is,

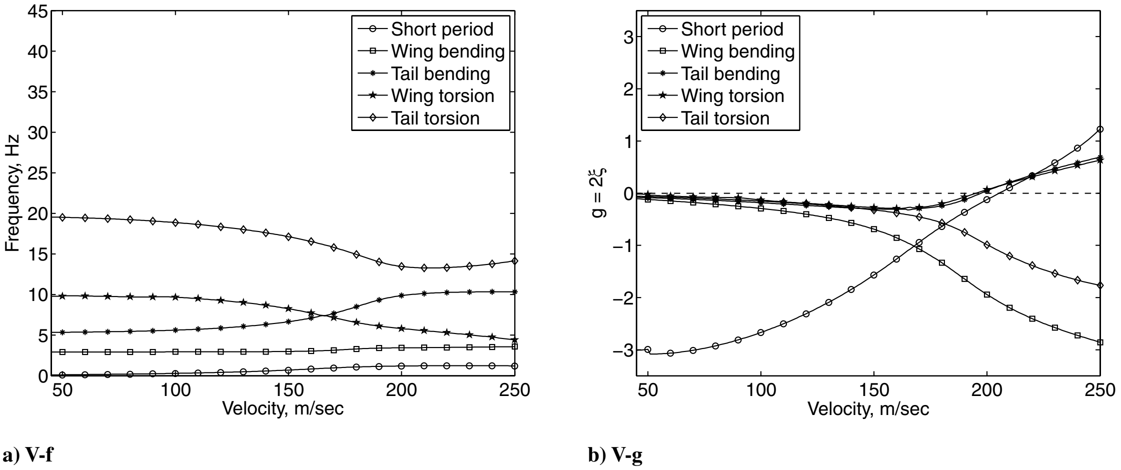
$$y_j(k) = \alpha \tanh(\beta v_j(k)) \tag{27}$$

with $\alpha = \sqrt{3}$ and $\beta = \frac{2}{3}$, as suggested by Haykin [30]. The peculiarity of recurrent networks is that all neurons' output are used as input with a time-step delay. A neural network is de facto a nonlinear map from the N_i-dimensional input space to the N_o-dimensional output space; with a recurrent architecture, this map becomes a dynamic map. Dealing with dynamic systems, the use of a dynamic neural network is the natural choice. Moreover, with a recurrent architecture, it is possible to limit the number of neurons (compared with a static network) with a great benefit for the real-time implementation of the controller. In such a network architecture, it is possible to find two different types of neurons: the visible and the hidden neurons. The former are neurons for which the desired output value is known during the network training phase; the latter represent the network internal states, and their desired output values are not known.

A. Neuroidentifier

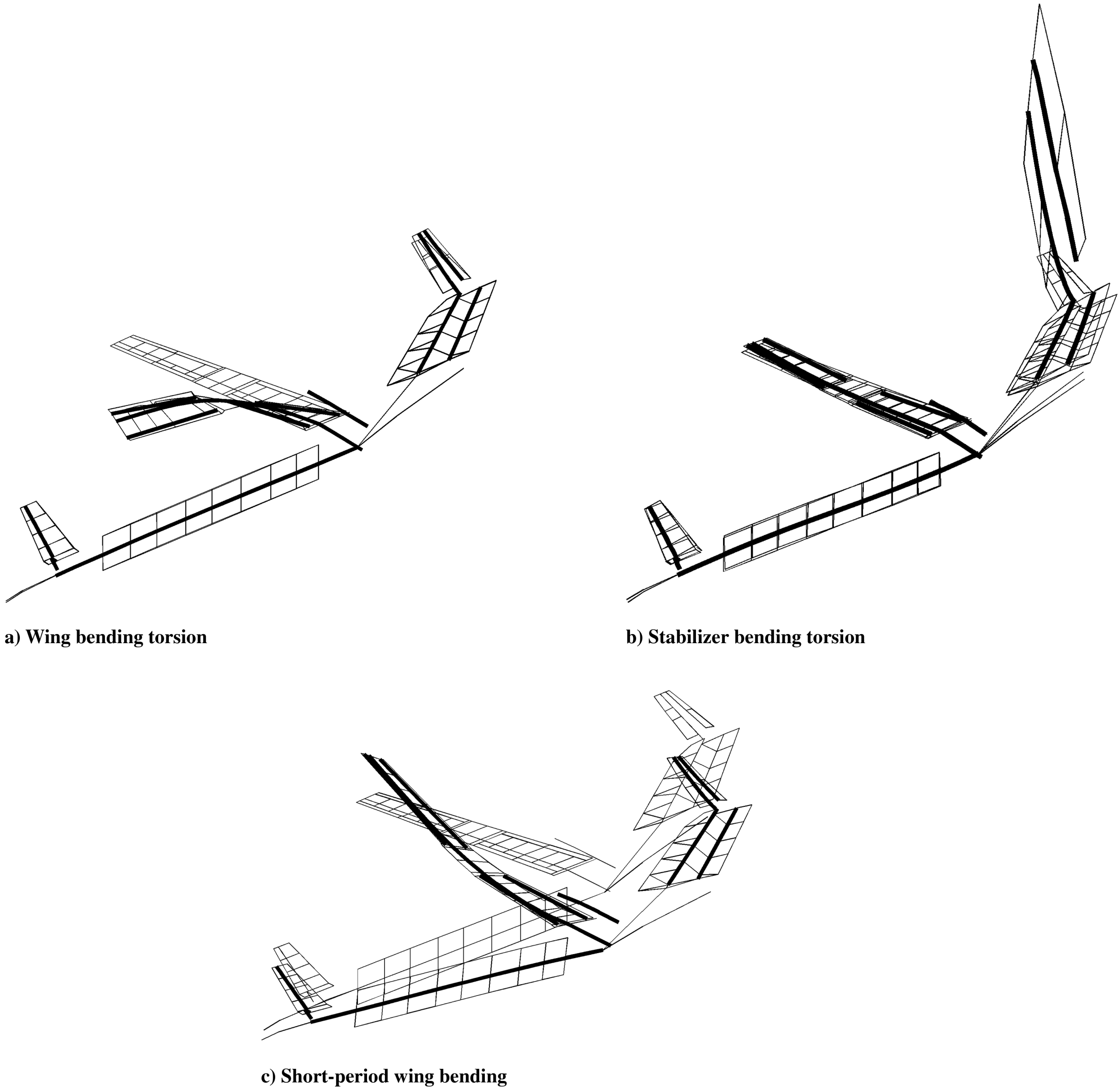
Neural networks are quite common in system identification due to their well-known capability to be a universal approximator of generic nonlinear functions within a desired accuracy. In an adaptive control system, the identifier is designed to reproduce the input–output relationship of the system. Therefore, it can be seen as an operator P, which represents a map from the input $\boldsymbol{u}$ to the output $\boldsymbol{y}$:

$$\boldsymbol{y}^{ID}(t) = \boldsymbol{P}(\boldsymbol{u}(t)) \tag{28}$$



a) V-f **b) V-g**

Fig. 8 Flutter diagrams without the aeroelastic controller.



a) Wing bending torsion

b) Stabilizer bending torsion

c) Short-period wing bending

Fig. 9 Structural forms of flutter modes.

where $\boldsymbol{y}^{ID}(t)$ is the identified output. At the same time, for the real physical system, it is possible to say that the measures $\boldsymbol{y}^{M}(t)$ are

$$\boldsymbol{y}^{M}(t) = \boldsymbol{F}(\boldsymbol{u}(t)) \tag{29}$$

The operator $\boldsymbol{P}$ is designed to keep the identification error low, defined as $\|\boldsymbol{y}^{ID}(t) - \boldsymbol{y}^{M}(t)\|_2$.

The identifier is realized by a RNN for which the inputs are the accelerometer measures and the movable surface deflections at the kth time step (plus the hidden and visible neuron feedback signals) and the outputs are the identified measures at the $(k+1)$th time step. So,

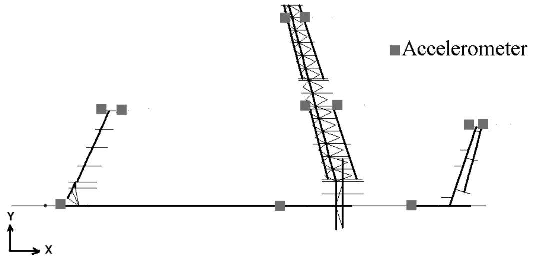


Fig. 10 Positioning of the accelerometers on the aircraft structure.

$$\boldsymbol{y}^{ID}(k+1) = \boldsymbol{P}(\boldsymbol{u}^{ID}(k)) \tag{30}$$

with

$$\boldsymbol{u}^{ID}(k) = \{\, \boldsymbol{y}^{ID}_{\text{visible}}(k), \quad \boldsymbol{y}^{ID}_{\text{hidden}}(k), \quad \boldsymbol{y}^{M}(k), \quad \boldsymbol{\delta}^{C}(k), \quad -1 \,\}^{T} \tag{31}$$

The identification network is trained to minimize the quadratic identification error E^{ID} defined as

$$E^{ID}(k) = \frac{1}{2}\sum_{j=1}^{N_M}(e^{ID}_j(k))^2 \tag{32}$$

where N_M is the number of measures and e^{ID}_j is the single-measure identification error, defined as the difference between the identified and measured acceleration:

$$e^{ID}_j(k) = y^{M}_j(k) - y^{ID}_j(k) \tag{33}$$

The synaptic weights are updated according to the *gradient technique*:

$$W^{ID}_{kl}(k+1) = W^{ID}_{kl}(k) - \eta \frac{\partial E^{ID}(k)}{\partial W^{ID}_{kl}(k)} \tag{34}$$

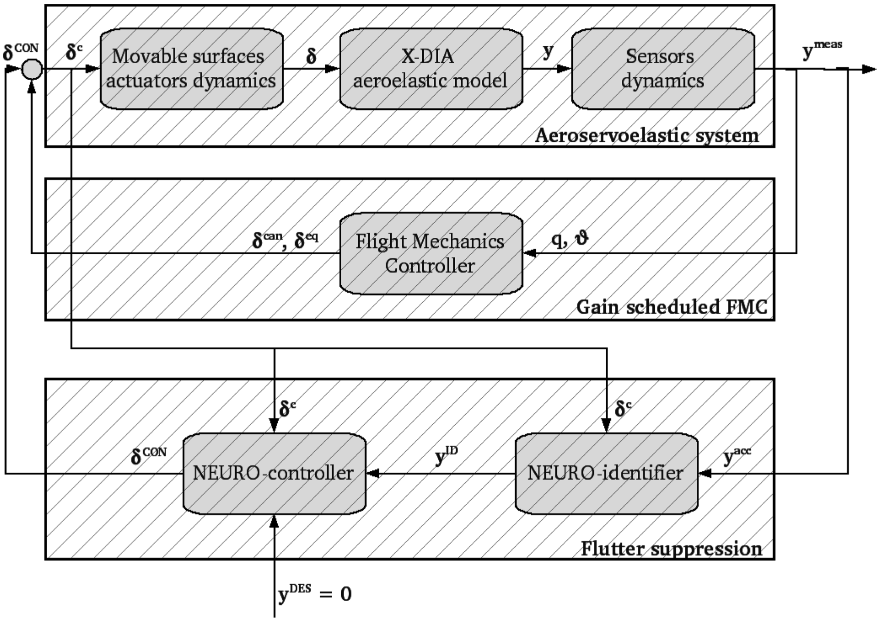


Fig. 11 Block diagram of the implemented control system.

Therefore, the variation of weights goes in the opposite direction of the gradient of the identification error. The parameter η is called the *learning rate* and it is the fundamental parameter governing the RNN training velocity. The learning rate value is limited by stability considerations: too-high values may destabilize the training algorithm, causing the divergence of synaptic weights. On the other hand, a low learning rate would limit the control system capability to quickly adapt itself to system variations. Unfortunately, there is no general theory to predict η limiting values for RNNs, and so the best value for it must be tuned during the design simulations.

The gradient of the quadratic identification error, required by Eq. (34), is computed according to the real-time recurrent learning (RTRL) algorithm [31]. Using Eqs. (32) and (33), the derivative in Eq. (34) is equal to

$$\frac{\partial E^{ID}(k)}{\partial W_{kl}^{ID}(k)} = \sum_{j=1}^{N_M} e_j^{ID}(k)\frac{\partial e_j^{ID}(k)}{\partial W_{kl}^{ID}(k)} = -\sum_{j=1}^{N_M} e_j^{ID}(k)\frac{\partial y_j^{ID}(k)}{\partial W_{kl}^{ID}(k)} \tag{35}$$

The term $\partial y_j^{ID}(k)/\partial W_{kl}^{ID}(k)$ is initially equal to zero. Considering that the jth neuron output is

$$y_j^{ID}(k+1) = f(v_j^{ID}(k)) \tag{36}$$

it follows that

$$\frac{\partial y_j^{ID}(k+1)}{\partial W_{kl}^{ID}(k)} = f'(v_j^{ID}(k))\frac{\partial v_j^{ID}(k)}{\partial W_{kl}^{ID}(k)} \tag{37}$$

and from Eqs. (26–31),

$$\frac{\partial v_j^{ID}(k)}{\partial W_{kl}^{ID}(k)} = \sum_{i=1}^{N_{\text{in}}^{ID}} W_{ji}^{ID}(k)\frac{\partial u_i^{ID}(k)}{\partial W_{kl}^{ID}(k)} + \delta_{jk}u_l^{ID}(k)$$
$$= \sum_{i=1}^{N_{\text{neuron}}^{ID}} W_{ji}^{ID}(k)\frac{\partial y_i^{ID}(k)}{\partial W_{kl}^{ID}(k)} + \delta_{jk}u_l^{ID}(k) \tag{38}$$

In this way, the term $\partial y_j^{ID}(k)/\partial W_{kl}^{ID}(k)$ can be updated for the next time step. The summation in Eq. (38) can be limited to the first N_{neuron}^{ID} inputs by considering that only the first N_{neuron}^{ID} inputs by means of the feedback depend on synaptic weights. To effectively compute the error gradient, the RTRL algorithm introduces the following approximation,

$$\frac{\partial y_i^{ID}(k)}{\partial W_{kl}^{ID}(k)} = \frac{\partial y_i^{ID}(k)}{\partial W_{kl}^{ID}(k-1)} \tag{39}$$

which means that the weights variation during the training phase is neglected. Through simple simulations, it has been verified that this approximation does not invalidate the identifier.

The performance of the RNN identifier has been evaluated on the aeroservoelastic system using random deflections of the movable surfaces as background excitation on the system. Figure 13 shows the comparison between the acceleration signal measured at the wing tip leading edge with the signal identified by the RNN. In particular, Fig. 13a clearly shows how the synaptic weights initialized at zero evolve rapidly toward the correct value to generate a fairly good signal after only 4 s. In 50 s, the identification process reached a perfect convergence, as shown in Fig. 13b.

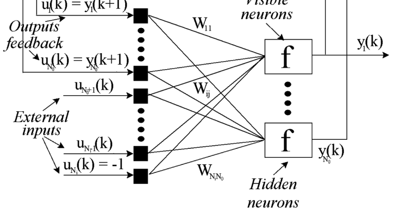


Fig. 12 Recurrent neural network architecture.

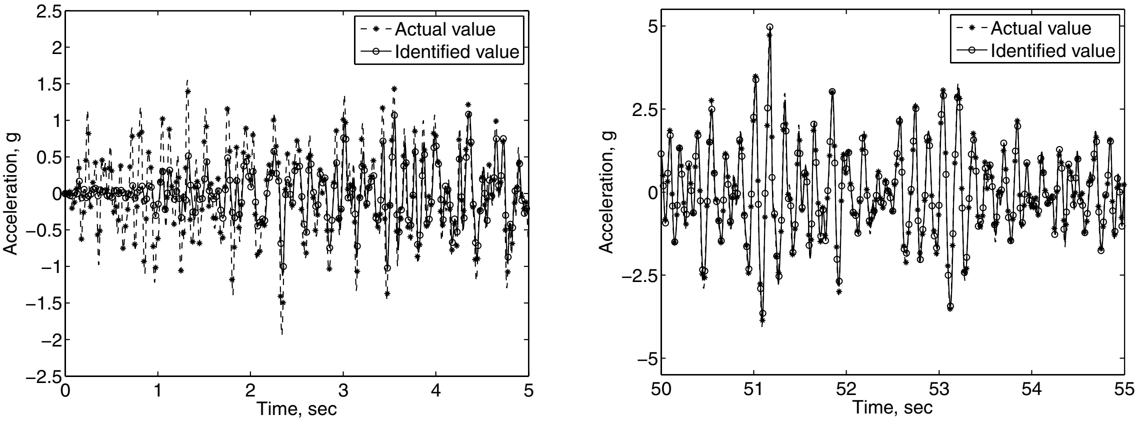
B. Neurocontroller

The control strategy implemented is based on an indirect inversion of the identified system to obtain the desired system output $\boldsymbol{y}^D$. The control effort $\boldsymbol{u}^C$ is obtained by computing the inverse of the operator P defined in Eq. (28); that is,

$$\boldsymbol{u}^C = \boldsymbol{P}^{-1}(\boldsymbol{y}^D) \tag{40}$$

The inputs to the RNN controller are the output of the identification RNN $\boldsymbol{y}^{ID}(k+1)$ and the control-surface deflections $\boldsymbol{\delta}^C(k)$, and so

$$\boldsymbol{u}^{\text{CON}}(k) = \{\boldsymbol{y}_{\text{visible}}^{\text{CON}}(k),\quad \boldsymbol{y}_{\text{hidden}}^{\text{CON}}(k),\quad \boldsymbol{y}^{ID}(k+1),\quad \boldsymbol{\delta}^C(k),\quad -1\}^T \tag{41}$$



a)

b)

Fig. 13 Comparison of the acceleration measured at wing tip with the signal identified by the RNN with a learning rate of 0.2. On the left are shown the first few seconds after the beginning of identification; on the right the two signals after 50 s.

whereas the outputs are the desired control efforts [i.e., the $\boldsymbol{\delta}^C(k+1)$ to be applied]. Here, the network is trained to minimize the quadratic error combined with an additional term, called the *control penalization term*, used to limit the control effort in a way similar to that done in standard quadratic optimal control:

$$E^{\text{CON}}(k) = \frac{1}{2}\sum_{j=1}^{N_M}(e_j^{\text{CON}}(k))^2 + \frac{1}{2}\rho\sum_{j=1}^{N_C}(\delta_j^C(k))^2 \tag{42}$$

with

$$e_j^{\text{CON}}(k) = y_j^D(k+1) - y_j^{ID}(k+1) \tag{43}$$

In a nonminimum-phase system, such as the one under investigation, all zeros in the right half of the complex plane may be transformed into unstable poles of the controller by a straight system inversion. In this case, the penalty term is necessary to avoid the divergence of the control [32].

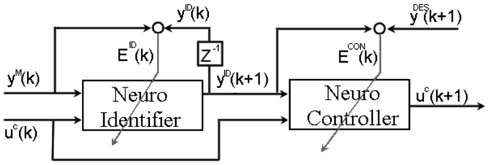


Fig. 14 Adaptive neural controller framework.

The control network is trained, like the identification network, following the gradient technique and using the RTRL algorithm to calculate the error gradient. In this case,

$$\frac{\partial E^{\text{CON}}(k)}{\partial W_{kl}^{\text{CON}}(k)} = -\sum_{j=1}^{N_M} e_j^{\text{CON}}(k)\frac{\partial y_j^{ID}(k+1)}{\partial W_{kl}^{\text{CON}}(k)} + \rho\sum_{j=1}^{N_C}\delta_j^C(k)\frac{\partial\delta_j^C(k)}{\partial W_{kl}^{\text{CON}}(k)} \tag{44}$$

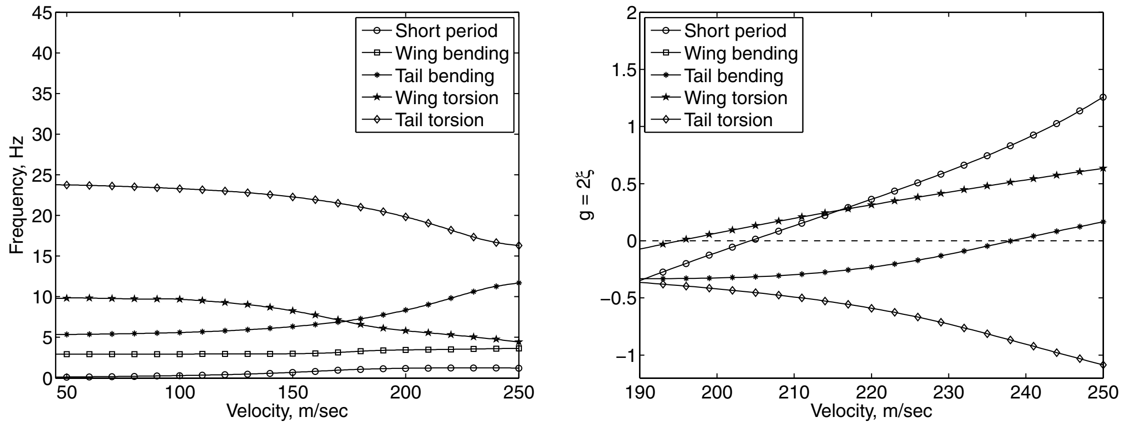
Using Eq. (36),

$$\frac{\partial y_j^{ID}(k+1)}{\partial W_{kl}^{\text{CON}}(k)} = f'(v_j^{ID}(k))\frac{\partial v_j^{ID}(k)}{\partial W_{kl}^{\text{CON}}(k)} \tag{45}$$

Only $\boldsymbol{\delta}^C$ in the identification network input vector (31) depends on the weights of the control network, and the weights of the identification network are not function of the weights of the control network. So it results that

$$\begin{aligned}\frac{\partial v_j^{ID}(k)}{\partial W_{kl}^{\text{CON}}(k)} &= \sum_{i=1}^{N_{\text{in}}^{ID}} W_{ji}^{ID}(k)\frac{\partial u_i^{ID}(k)}{\partial W_{kl}^{\text{CON}}(k)} \\ &= \sum_{i=1}^{N_C} W_{j(i+N_{\text{neuron}}^{ID}+N_M)}^{ID}\frac{\partial y_i^{\text{CON}}(k)}{\partial W_{kl}^{\text{CON}}(k)}\end{aligned} \tag{46}$$

$$\frac{\partial y_j^{\text{CON}}(k+1)}{\partial W_{kl}^{\text{CON}}(k)} = f'(v_j^{\text{CON}}(k))\frac{\partial v_j^{\text{CON}}(k)}{\partial W_{kl}^{\text{CON}}(k)} \tag{47}$$



a) V-f

b) V-g

Fig. 15 Flutter analysis without aeroelastic control system of the modified aircraft.

Table 1 Parameters used for the RNNs

	Neuroidentifier	Neurocontroller
Frequency	200 Hz	200 Hz
Neuron number	18	18
Learning rate	$\eta = 0.2$	$\eta^A = 0.1$; $\eta^B = 0.4$
Penalty term	——	$\rho^A = 0.3$; $\rho^B = 0.02$

Table 2 Computational time required by the RNN controller

		Controller		
	t^{ID}	8 neurons	12 neurons	16 neurons
ID 8 neurons	0.9 ms	1.2 ms	1.5 ms	2.3 ms
ID 12 neurons	1.2 ms	1.2 ms	1.6 ms	2.4 ms
ID 16 neurons	1.8 ms	1.3 ms	1.7 ms	2.6 ms

$$\frac{\partial v_j^{\mathrm{CON}}(k)}{\partial W_{kl}^{\mathrm{CON}}(k)} = \sum_{i=1}^{N_{\mathrm{neuron}}^{\mathrm{CON}}} W_{ji}^{\mathrm{CON}}(k)\frac{\partial y_i^{\mathrm{CON}}(k)}{\partial W_{kl}^{\mathrm{CON}}(k)} + \delta_{kj} u_l^{\mathrm{CON}}(k) \qquad (48)$$

Instead, the term in Eq. (44) related to the control penalization is simply equal to

$$\sum_{j=1}^{N_C} \delta_j^C(k)\frac{\partial \delta_j^C(k)}{\partial W_{kl}^{\mathrm{CON}}(k)} = \sum_{j=1}^{N_C} y_j^{\mathrm{CON}}(k)\frac{\partial y_j^{\mathrm{CON}}(k)}{\partial W_{kl}^{\mathrm{CON}}(k)} \qquad (49)$$

The variation of weights during the training phase is neglected here.

The resulting structure of the neural adaptive control system is summarized in Fig. 14. As a matter of fact, the adaptivity of the control system is achieved by keeping the networks' training always active, so that the control adapts itself to system variations by modifying the networks' weights appropriately.

IV. Results

The straight application of the described control strategy to the whole aircraft model failed in achieving flutter suppression. In fact, as previously pointed out, the presence of three flutter modes in a small velocity range and the excessive tail compliance made the control system task too difficult. Therefore, it has been decided to adopt a partly passive strategy by stiffening the tail to increase the first torsional frequency of the stabilizer by about 40%. It should be noted that such a stiffening also appeared to be required to fix an inadequate static-strength behavior. Figure 15 presents the results of a flutter analysis on the stiffened model, showing a clear increase of the flutter velocity of the stabilizer bending-torsion flutter, moving it well outside the limits of the flight envelope, while keeping the related weight penalty acceptable.

As a consequence, the number of accelerometers has been reduced to six: four on the wing and two on the fuselage (one on the nose and one on the rear) to observe the aircraft pitch. The resulting configuration of the control system can be used to stabilize the two remaining flutters and to extend the flight envelope by 15%, as shown in the following paragraphs.

A. Neural Network Parameters

Extensive numerical simulations using the design model were carried out to tune all network parameters. The controller sampling rate was set to 200 Hz, considering that the maximum flutter frequency of interest is around 10 Hz. Both the identification and the control networks are composed of a single layer of 18 neurons, 6 of which are *visible*: they correspond to the 6 identified accelerations for the identification network and to the 6 movable surfaces rotations for the control network. It has been verified that an additional increase in the number of neurons does not improve the control behavior significantly. The learning rates are a result of a tradeoff: high rates lead to control algorithm instability, whereas too-low rates reduce the control capability to quickly adapt itself to system variations. All parameters chosen for the different RNNs are presented in Table 1.

To improve the behavior of the controller in suppressing the pitch-bending flutter, the control network has been split in two separate parts, each one using a different learning rate and control penalty term. Part A is specialized in stabilizing the torsion-bending wing flutter using the four wing accelerometers and the four wing control surfaces, and the part B is dedicated to the pitch-bending flutter using only the elevator and canard deflections and fuselage accelerometer signals. The parameters of the resulting networks are summarized in

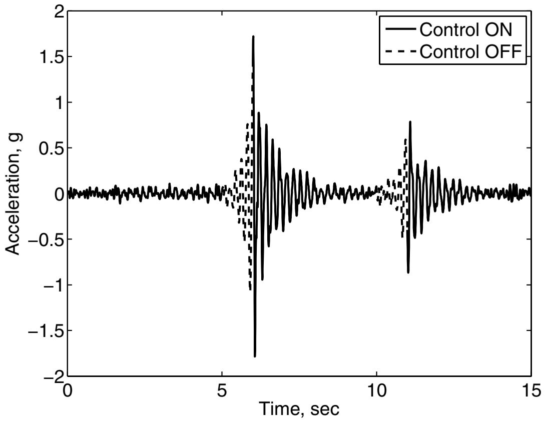
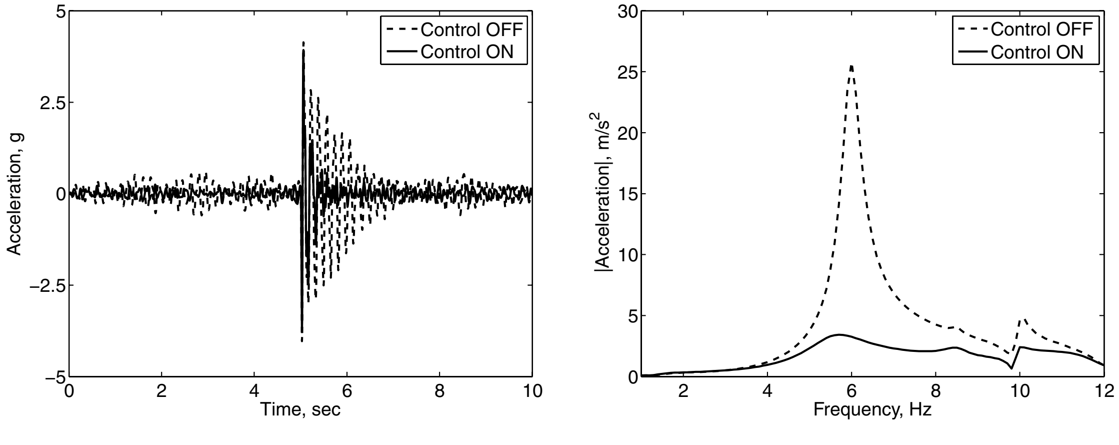


Fig. 17 Wing tip leading-edge acceleration at $V_\infty = 220\ \mathrm{m/s}$.



a) Time domain

b) Frequency domain

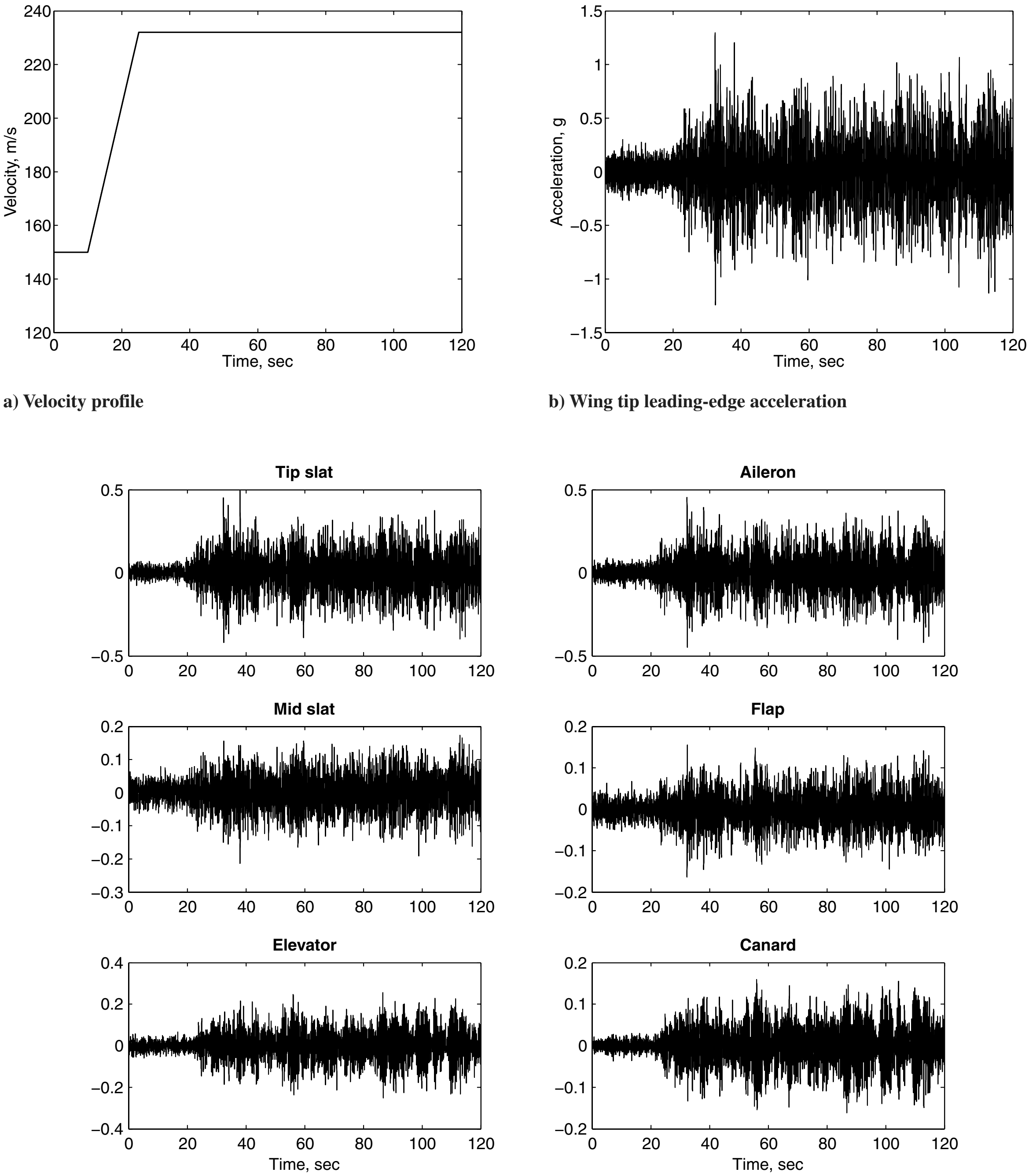
Fig. 16 Wing tip trailing-edge acceleration at $V_\infty = 190\ \mathrm{m/s}$.

Table 1. All numerical simulations have been performed in MATLAB and without any code optimization. Nevertheless, the computational time required (shown in Table 2) is small and will not jeopardize a real-time implementation of such a controller.

To start from a set of realistic preliminary synaptic weights, both networks are trained offline, with the very same training algorithm described in the previous section. In this way, excessive accelerations during the initial online training phase are avoided and the training time is reduced. During each simulation, the network training is kept online to achieve the required adaptiveness.

B. Simulation Results

The designed flutter-suppression system has been tested by running several simulations to quantify the achieved performances. During all simulations, the system has been perturbed by random deflections of the control surfaces to keep the system identification always active, with a maximum frequency of 20 Hz and an amplitude of 0.2 deg for the wing surfaces and 0.02 deg for the elevator and the canard surface. A zero-mean-measurement white noise of 0.1 g intensity is added to the accelerometer signals to test neural control system robustness. Figure 16a shows the signal of the accelerometer placed at the wing tip trailing edge, due to an impulsive deflection of 5 deg of the wing aileron when the aircraft is flying at a velocity of 5.5 m/s below the first flutter onset with control turned on and off. The significant increase of damping can also be seen from Fig. 16b, which shows the frequency content of the signal. Figure 17 presents the results of a simulation at a flight velocity of 220 m/s, well beyond the first and second flutter velocities. The control system is able to keep the aircraft stable, and it is also able to suitably recover stability



a) Velocity profile

b) Wing tip leading-edge acceleration

c) Control surfaces deflections (in deg)

Fig. 18 Variable flight velocity simulation.

Table 3 Response to atmospheric turbulence: accelerometer signals variance in $\mathrm{m}^2/\mathrm{s}^4$. σ_W^{off} is the variance of the wing trailing-edge acceleration computed with control off, and σ_W^{on} is the same with the control on. σ_F is the variance of the acceleration of the center of mass of the aircraft

Flight velocity	σ_W^{off}	σ_W^{on}	$\sigma_W^{\mathrm{on}}/\sigma_W^{\mathrm{off}}$	σ_F^{off}	σ_F^{on}	$\sigma_F^{\mathrm{on}}/\sigma_F^{\mathrm{off}}$
160 m/s	28.9	17.3	0.59	0.29	0.16	0.55
190 m/s	51.9	15.4	0.29	0.43	0.22	0.51
220 m/s	——	28.4	——	——	0.57	——

after turning off the controller for a short period of time, during which the instability develops. Finally, Fig. 18b shows the result of a simulation with a changing velocity, using a profile shown in Fig. 18a. Such a simulation was carried out to test the capability of the controller to widen the flutter-free flight envelope of about 30 m/s and to adapt itself automatically to flight velocity variations. Furthermore, Fig. 18c shows that these results have been achieved with a limited control effort, because the largest surface deflection is below 1 deg. Table 3 shows the variances of the accelerations at the wing tip and at the aircraft center of mass due to atmospheric turbulence, modeled as a random noise with a limited frequency content. Also in this case, the neural control system has a beneficial effect in terms of gust alleviation, due to its capability to increase the system damping.

Simulations performed with the verification model showed essentially the same behavior as the neural control system, proving the absence of any spillover effect.

V. Conclusions

In this work, the feasibility of a RNN adaptive control scheme for flutter suppression of an aircraft has been demonstrated. To achieve these results, it has been necessary to implement a comprehensive aeroservoelastic model in the time domain, correctly taking into account the flight dynamics of the aircraft.

It has been shown that the neural control is able to increment the system damping to 1) increase the flutter onset velocity and 2) improve the aircraft gust response. The controller shows a good robustness, ensuring the stabilization in the presence of a significant measurement noise. A passive tail stiffening has been required to stabilize the aircraft. In future activities, it will be interesting to analyze in more detail the possibility to control the tail compliance by an active system by increasing the number of control surfaces on it. Because the proposed neural control system is naturally nonlinear, an interesting development of this study would be to test the controller with a nonlinear model of the aircraft, including movable surface backlashes and friction, to evaluate the capability of suppressing possible limit cycle oscillations. An experimental test of such a control methodology is already scheduled on the 1/10-scaled model of the X-DIA in the DIAPM wind tunnel.

Acknowledgments

The authors wish to acknowledge the help of Sergio Ricci and Alessandro Scotti in accessing the X-DIA aircraft numerical and experimental data.

References

[1] Kendall, E., "The Aerodynamics of Three-Surface Airplanes," AHS, ASEE, Aircraft Design Systems and Operations Meeting, San Diego, CA, AIAA Paper 1984-2508, 1984.

[2] Owens, D., and Coe, P., Jr, "Exploratory Wind Tunnel Investigation of the Stability and Control Characteristics of a Three-Surface, Forward-Swept Wing Advanced Turboprop Model," 8th AIAA Applied Aerodynamics Conference, Portland, OR, AIAA Paper 1990-3074, Aug. 1990.

[3] Strohmeyer, D., Seubert, R., Heinze, W., Osterheld, C., and Fornasier, L., "Three Surface Aircraft—A Concept for Future Transport Aircraft," 38th AIAA Aerospace Science Meeting and Exhibit, Reno, NV, AIAA Paper 2000-0566, 2000.

[4] Agnew, J., and Hess, J., "Benefits of Aerodynamic Interaction to the Three-Surface Configuration," *Journal of Aircraft*, Vol. 17, No. 11, 1980, pp. 823–827.
doi:10.2514/3.57971

[5] Ricci, S., Scotti, A., Malecek, J., and Cecrdle, J., "Experimental Investigations of a Vibration Suppression System for a Three Surface Aeroelastic Model," Proceedings of the 46th Structures, Structural Dynamics and Materials Conference, Austin, TX, AIAA Paper 2005-2232, Apr. 2005.

[6] Ricci, S., Scotti, A., and Zanotti, D., "Control of an All-Movable Foreplane for a Three Surface Aircraft Wind Tunnel Model," *Mechanical Systems and Signal Processing*, Vol. 20, July 2006, pp. 1044–1066.
doi:10.1016/j.ymssp.2005.08.020

[7] Scotti, A., Quaranta, G., and Ricci, S., "Active Control of three Surface Wind Tunnel Aeroelastic Demonstrator: Modelling and Correlation," CEAS/AIAA/DGLR International Forum on Aeroelasticity and Structural Dynamics (IFASD-2005), Munich, Paper 029, June 28–July 1 2005.

[8] Ricci, S., Scotti, A., De Gaspari, A., and Riccobene, L., "Active Aeroelastic Control over a Multi-Surface Wing: Modelling and Wind Tunnel Testing," 48th AIAA, ASME, ASCE, AHS, ASC, Structures, Structural Dynamics and Material Conference, Honolulu, HI, AIAA Paper 2007-2138, Apr. 2007.

[9] Levin, A., and Narendra, K., "Control of Nonlinear Dynamical Systems Using Neural Networks: Controllability and Stabilization," *IEEE Transactions on Neural Networks*, Vol. 4, No. 2, 1993, pp. 192–206.
doi:10.1109/72.207608

[10] Levin, A., and Narendra, K., "Control of Nonlinear Dynamical Systems Using Neural Networks, Part 2: Observability, Identification, and Control," *IEEE Transactions on Neural Networks*, Vol. 7, No. 1, 1996, pp. 30–42.
doi:10.1109/72.478390

[11] Haley, P., and Soloway, D., "Generalized Predictive Control for Active Flutter Suppression," *IEEE Control Systems Magazine*, Vol. 17, No. 4, 1997, pp. 64–70.
doi:10.1109/37.608553

[12] Scott, R., and Pado, L., "Active Control of Wind-Tunnel Model Aeroelastic Response Using Neural Networks," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 6, 2000, pp. 1100–1108.
doi:10.2514/2.4661

[13] Jin, L., Nikiforuk, P., and Gupta, M., "Adaptive Control of Discrete-Time Nonlinear Systems Using Recurrent Neural Networks," *Control Theory and Applications*, Vol. 141, No. 3, 1994, pp. 169–176.

[14] Chow, T., and Fang, Y., "A Recurrent Neural-Network-Based Real-Time Learning Control Strategy Applying to Nonlinear Systems with Unknown Dynamics," *IEEE Transactions on Industrial Electronics*, Vol. 45, No. 1, 1998, pp. 151–161.
doi:10.1109/41.661316

[15] Bernelli-Zazzera, F., and Lo-Rizzo, V., "Adaptive Control of Space Structures via Recurrent Neural Networks," *Dynamics and Control*, Vol. 9, No. 1, 1999, pp. 5–20.
doi:10.1023/A:1008323023512

[16] Bernelli-Zazzera, F., Mantegazza, P., Mazzoni, G., and Rendina, M., "Active Flutter Suppression Using Recurrent Neural Networks," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 6, 2000, pp. 1030–1036.
doi:10.2514/2.4671

[17] Albano, E., and Rodden, W. P., "A Doublet–Lattice Method for Calculating the Lift Distributions on Oscillating Surfaces in Subsonic Flow," *AIAA Journal*, Vol. 7, No. 2, 1969, pp. 279–285.
doi:10.2514/3.5086

[18] Davies, D., "Generalized Aerodynamic Forces on a *T*-Tail Oscillating Harmonically in Subsonic Flow," Royal Aircraft Establishment, Rept. 295, London, May 1964.

[19] Rodden, W. P., and Love, J. R., "Equation of Motion of a Quasisteady Flight Vehicle Utilizing Restrained Static Aeroelastic Characteristics," *Journal of Aircraft*, Vol. 22, No. 9, 1985, pp. 802–809.
doi:10.2514/3.45205

[20] Morino, L., Mastroddi, F., De Troia, F., Ghiringhelli, G. L., and Mantegazza, P., "Matrix Fraction Approach for Finite-State Aerodynamic Modeling," *AIAA Journal*, Vol. 33, No. 4, 1995, pp. 703–711.
doi:10.2514/3.12381

[21] Roger, K. L., "Airplane Math Modeling Methods for Active Control Design," AGARD, TR CP-228, Neuilly-sur-Seine, France, Aug. 1977.

[22] Antoulas, A., and Sorensen, D., "Approximation of Large-Scale Dynamical Systems: An Overview," *International Journal of Applied*

Mathematics and Computer Science, Vol. 11, No. 5, 2001, pp. 1093–1121.

[23] Pasinetti, G., and Mantegazza, P., "Single Finite States Modeling of Aerodynamic Forces Related to Structural Motions and Gusts," *AIAA Journal*, Vol. 37, No. 5, May 1999, pp. 604–612. doi:10.2514/2.760

[24] Merritt, H. E., *Hydraulic Control Systems*, Wiley, New York, 1967.

[25] *General Specification for Flight Control Systems—Design, Installation and Test of Piloted Aircraft*, U.S. Air Force, MIL-SPEC MIL-F-9490D, 5 Oct. 1992.

[26] Friedland, B., *Control System Design. An Introduction to State Space Methods*, McGraw–Hill, New York, 1987.

[27] Lind, R., and Brenner, M., *Robust Aeroservoleastic Stability Analysis, Advances in Industrial Control*, Springer, New York, 1999.

[28] Etkin, B., *Dynamics of Atmospheric Flight*, Wiley, New York, 1972.

[29] Hancock, G. J., *An Introduction to the Flight Dynamics of Rigid Aeroplanes*, Prentice–Hall, Upper Saddle River, NJ, 1995.

[30] Haykin, S., *Neural Networks: A Comprehensive Foundation*, Prentice–Hall, Upper Saddle River, NJ, 1994.

[31] Williams, R., and Zipser, D., "A Learning Algorithm for Continually Running Fully Recurrent Networks," *Neural Computation*, Vol. 1, No. 2, 1989, pp. 270–280. doi:10.1162/neco.1989.1.2.270

[32] Isermann, R., Matko, D., and Lachmann, K., *Adaptive Control Systems*, Prentice–Hall, Upper Saddle River, NJ, 1992.