SIMULATION AND IDENTIFICATION OF A FLIGHT ENVELOPE USING NEURAL NETWORKS

Luciane de Fátima Rodrigues de Souza
lfrs@sc.usp.br

Carlos Eduardo Beluzo
beluzo@info99.com.br

Eduardo Morgado Belo
belo@sc.usp.br

Flávio Donizeti Marques
fmarques@sc.usp.br

Engineering School of São Carlos – University of São Paulo.
NPA - Nucleus of Research in Aeronautics
LADinC - Aeroelasticity, Flight Dynamics and Control Laboratory.
Av. Trabalhador Sancarlense, 400.
13560-590, São Carlos, SP, Brazil.

***Abstract**. The flight dynamics of an aircraft can be represented through a set of 12 non-linear ordinary differential equations, which can be solved using a dimensional or non-dimensional system. This representation of the movement of an aircraft involves coupling of the motion equations, making the solution of this system quite complex and slow by conventional methods. The objective of this work is to present an assessment for the identification of a flight envelope using recurrent artificial neural networks implemented in the Matlab® environment. A C++ program was also developed to simulate the aircraft. The C++ program implements and solves the motion equations, thus supplying the numerical values that describe the flight dynamics. In the simulation, the program uses as input parameters the geometric and aerodynamic data, the dimensional stability derivatives and aerodynamic coefficients. As outputs, it supplies the values of the motion variables such as accelerations, speeds, spatial position and orientation. Seven flight conditions with varied altitude and Mach number had been simulated. Three of these conditions had been used to train the neural network. This network used, as fixed inputs, altitude and Mach number, and, as dynamic input, variations of the elevator angle. To verify the performance of the neural network, responses for the other 4 conditions had been generated and compared with the answers generated by the C++ simulation program.*

***Keywords**: simulation, flight envelope, identification, non-linear systems, neural networks.*

1. Introduction

Flight dynamics is the area of the aeronautical engineering that studies the movement of vehicles flying through the atmosphere. The complex nature of the atmosphere, as well as the dynamic behavior of an aircraft, guarantees to this area challenging problems (Marques, 2003). Examples of this are the facts of the modern aircraft presenting every time bigger reduction of weight, increasing speed, etc, certainly suffering with the increase of non-linear effects. Simulations of non-linear effects using traditional methods can computationally be expensive and of difficult implementation. Therefore, the use of identification methods, in cases where the attainment of a set of equations capable of foreseeing the behavior of a system is not easy, has been sufficiently explored, mainly with the high level of development and increasing availability of faster and faster computers and with high capacity of processing.

Identification of systems is a technique for establishing dynamic systems mathematical models from input and output measurements data (Klein, 1989). Some identification methods in the area of flight dynamics have been considered and verified, such as the methods of the error equation (Morelli, 1999), the error equation in the frequency domain (Iliff and Wang, 1997), (Mendonça and Góes, 2003) and other methods.

The neural networks have been largely applied in this area due to their easiness of implementation and fast capacities of processing and learning. Linse and Stengel, (1993) had used multilayer recurrent neural networks associated to the method of extended Kalman filter to expand and to identify a flight envelope of a transport jet aircraft. The extended Kalman filter was used for the estimation of states and neural coefficients of normal forces and networks for the aerodynamic modeling of the aircraft. Satisfactory results had been shown. Raisinghani and Ghosh, (2000) had shown the application of artificial neural networks to the modeling problem and estimation of parameters for aeroelastic aircraft without needing measurements of elastic deflections or their derivatives. They had clearly shown the applicability of neural networs for modeling and estimation of parameters. Allen and Dibley, (2003) had used neural networks to identify bending moment loadings, torcion loads and hinge moments of control surfaces of the active wing of the aeroelastic airplane (AAW), using only as inputs the angle of attack and the rolling speed. This work supplies the used input data, the selected input parameters, the structure, the training, and the validation of the neural networks models.

So, the objective of this work is to train recurrent neural networks, to identify aircraft flight envelopes, using for that a given example of the A4-D aircraft. To obtain the speed, acceleration and displacement responses of the aircraft, a flight simulator will be developed and implemented using the C++ language. For that the non-linear equations of motion, presented in Etkin and Reid, (1996) will be used.

2. The development and implementation of the simulator

The flight dynamics model of an aircraft consists basically of its geometric and mass description, together with the equations of motion and external loads that act on it. The first part of this work, as said previously, consists of simulating the flight dynamics of an aircraft, in order to implement artificial neural networks capable to identify the responses inside a pre-established flight envelope of the aircraft. For the simulation of an aircraft flight dynamics, it is necessary to resolve its system of motion equations. Amongst some models that describe the flight dynamics of an aircraft, the non-linear mathematical model presented by Etkin and Reid, (1996) was adopted. This model, although the use of some simplifications, presents coherent results and near to the real ones.

The representation of this system is made through a set of 12 ordinary differential equations and its solution can be found using dimensional or non-dimensional parameters. In this system, the stability derivatives of the aircraft in the dimensional form are used for the calculation of the forces and aerodynamic moments, which in turn are used to find the solution of the equations. Before presenting the system of equations some considerations adopted in the modeling will be presented.

The aircraft will be considered as being a rigid body with six degrees of freedom, with a longitudinal plan of symmetry, and under the effect of the aerodynamic and gravity forces. The atmosphere is considered in rest and the effects of Earth rotation and propulsion forces are neglected. A set of orthogonal axes *Oxyz* fixed to tha airplane is considered, being *O* the origin located in the center of gravity of the aircraft; *Ox* the axis in the longitudinal direction (horizontal); *Oy* the axis in the starboard or lateral direction and *Oz* the axis in the gravity force direction or vertical line.

So, the motion of an aircraft can be described by the following set of equations:

$$\dot{u} = \frac{FX}{m} - g \cdot sen\,\theta - q \cdot w + r \cdot v \tag{1}$$

$$\dot{v} = \frac{FY}{m} + g \cdot cos\,\theta \cdot sen\,\phi - r \cdot u + p \cdot w \tag{2}$$

$$\dot{w} = \frac{FZ}{m} + g \cdot cos\,\theta \cdot cos\,\phi - p \cdot v + q \cdot u \tag{3}$$

$$\dot{p} = \frac{ML + I_{zx}(\dot{r} + p \cdot q) + (I_y - I_z)q \cdot r}{I_x} \tag{4}$$

$$\dot{q} = \frac{MM + I_{zx}(r^2 - p^2) + (I_z - I_x)r \cdot p}{I_y} \tag{5}$$

$$\dot{r} = \frac{MN + I_{zx}(\dot{p} - qr) + (I_x - I_y)p \cdot q}{I_z} \tag{6}$$

$$\dot{\phi} = p + (q \cdot sin\phi + r \cdot cos\,\phi) \cdot tan\,\theta \tag{7}$$

$$\dot{\theta} = q \cdot cos\,\phi - r \cdot sin\phi \tag{8}$$

$$\dot{\psi} = (q \cdot sin\phi + r \cdot cos\,\phi) \cdot sec\,\theta \tag{9}$$

$$\begin{aligned} \dot{x} = {} & u \cdot cos\,\theta \cdot cos\,\psi + v \cdot (sen\,\phi \cdot sen\,\theta \cdot cos\,\psi - cos\,\phi \cdot sen\,\psi) \\ & + w \cdot (cos\,\phi \cdot sen\,\theta \cdot cos\,\psi + sen\,\phi \cdot sen\,\psi) \end{aligned} \tag{10}$$

$$\dot{y} = u \cdot cos\,\theta \cdot sin\,\psi + v \cdot (sin\phi \cdot sin\theta \cdot sin\,\psi + cos\,\phi \cdot cos\,\psi) + w(cos\,\phi sin\,\theta sin\,\psi - sin\,\theta\, cos\,\psi) \tag{11}$$

$$\dot{z} = -u \cdot sin\theta + v \cdot sin\phi \cdot cos\,\theta + w \cdot cos\,\phi \cdot cos\,\theta \tag{12}$$

where, respectively, u, v and w are the longitudinal, lateral and vertical linear velocities of the aircraft, p, q and r are the roll, pitch and yaw angular velocities, FX, FY and FZ are longitudinal, lateral and vertical aerodynamic loads, ML, MM and MN are the roll, pitch and yaw aerodinamic moments and ϕ, θ and ψ are the roll, pitch and yaw Euller angles. Also, respectivelly, x, y and z are the longitudinal, lateral and vertical spacial position of the aircraft, m is the total mass of the aircraft, I_x, I_y and I_z are the inertia moments in the Ox, Oy and Oz direction and I_{xy}, I_{yz} and I_{zx} are the inertia products relative to the planes Oxy, Oyz and Ozx.

The aerodynamic forces and moments acting on the aircraft are functions of the angle of attack and velocity components, and are represented by the following equations:

$$\Delta X = X_u \cdot \Delta u + X_w \cdot \Delta w + X_{\dot{w}} \cdot \Delta \dot{w} + X_q \cdot \Delta q + X_{\delta_e} \cdot \Delta \delta_e \tag{13}$$

$$\Delta Y = Y_v \cdot \Delta v + Y_p \cdot \Delta p + Y_r \cdot \Delta r + Y_{\delta_a} \cdot \Delta \delta_a + Y_{\delta_r} \cdot \Delta \delta_r \tag{14}$$

$$\Delta Z = Z_u \cdot \Delta u + Z_w \cdot \Delta w + Z_{\dot{w}} \cdot \Delta \dot{w} + Z_q \cdot \Delta q + Z_{\delta_e} \cdot \Delta \delta_e \tag{15}$$

$$\Delta L = L_v \cdot \Delta v + L_p \cdot \Delta p + L_r \cdot \Delta r + L_{\delta_a} \cdot \Delta \delta_a + L_{\delta_r} \cdot \Delta \delta_r \tag{16}$$

$$\Delta M = M_u \cdot \Delta u + M_w \cdot \Delta w + M_{\dot{w}} \cdot \Delta \dot{w} + M_q \cdot \Delta q + M_{\delta_e} \cdot \Delta \delta_e \tag{17}$$

$$\Delta N = N_v \cdot \Delta v + N_p \cdot \Delta p + N_r \cdot \Delta r + N_{\delta_a} \cdot \Delta \delta_a + N_{\delta_r} \cdot \Delta \delta_r \tag{18}$$

where X_u, X_w, $X_{\dot{w}}$, X_q, X_{δ_e}, Y_v, Y_p, Y_r, Y_{δ_a}, Y_{δ_r}, Z_u, Z_w, $Z_{\dot{w}}$, Z_q, Z_{δ_e} , L_v, L_p, L_r, L_{δ_a}, L_{δ_r}, M_u, M_w, $M_{\dot{w}}$, M_q, M_{δ_e} N_v, N_p, N_r, N_{δ_a} and N_{δ_r} represent the aerodynamic derivatives of stability, Δu, Δv and Δw represent the instantaneous values of the linear velocities perturbations, Δp, Δq and Δr represent the instantaneous values of the angular velocities perturbations and $\Delta \delta_e$, $\Delta \delta_a$ and $\Delta \delta_r$ represent variations of the control surfaces.

Considering that the aircraft is equilibrated, it is necessary to consider the initial values of the aerodynamic forces. These forces are represented by the following equations:

$$X_0 = m \cdot g sin\theta_0 \tag{19}$$

$$Y_0 = 0 \tag{20}$$

$$Z_0 = -m \cdot g\, cos\,\theta_0 \tag{21}$$

Then, the aerodynamic forces FX, FY and FZ and moments ML, MM and MN**,** which are substituted in equations (1) to (6) are defined by:

$$FX = \Delta X - X_0 \tag{22}$$

$$FY = \Delta Y - Y_0 \tag{23}$$

$$FZ = \Delta Z - Z_0 \tag{24}$$

Finally, in order to solve the system, equations (22), (23) and (24) are substituted in equations (1) to (12).

From this model, a software was developed that carries out the simulation of the flight dynamics of an aircraft. The software consists of the implementation and the resolution of this system of equations and is developed in the programming language C++.

The program solves this system through the interactive method of numerical integration *Runge-Kutta* of 4th order, as described in literature. The simulation is carried out using the given geometric and aerodynamic data, non-dimensional stability derivatives and aerodynamic coefficients, which are passed as input parameters for the software. These data are obtained from literature.

As output parameters, the program returns the values of the following variables: linear velocities, angular velocities and orientation of the aircraft. These results are presented in numerical values and plotted in cartesian graphics. The results obtained in simulation describe the behavior of the aircraft. The analysis of the aircraft behavior is done comparing different simulation results obtained for different flight conditions where the aileron, elevator and rudder inputs of the aircraft were varied.

The software works as described next. The program is executed and is in "waiting mode", or either, it waits until some event is carried out by the user. First the user defines the flight condition and enters with the respective values of the initial conditions. It is also necessary to define the integration step and the simulation time. The values of the dimensional aerodynamic derivatives are read of a text file previously defined. The input values for the drive of the controls of the aircraft (elevator, aileron and rudder), are also defined by the user with an input of the type "step", i.e., the input value is a stipulated fixed value for a time also determined. All these values are entered by the user, throught the input fields that are specified and available in the software interface, except the aerodynamic derivatives that are in a file, as described above. After that, the simulation is initiated by pressing an available button in the software interface.

For the simulation of the flight dynamics, i.e., for the solution of the system of equations, the altitude, the density in function of the altitude and the total velocity of the aircraft are defined. After the aerodynamic derivatives being supplied, the values of the forces and aerodynamic moments in equations (13) to (18) are calculated, using the initial values for the velocities (u, v, w, p, q and r). After that, these values for each equation (1) to (12) are calculated.

Done this, the numerical integration using the *Runge-Kutta* method is initiated, repeating the two last steps for each iteration of the method, in order to find the values for the linear velocities, angular velocities and orientation of the aircraft in the current instant. This is carried out every instant, in accordance with the defined integration step, and during the stipulated time of simulation. In each step the values found in the previous iteration are used as input values for the next iteration.

In this work, the longitudinal behavior of the A4-D aircraft will be simulated with data given in McRuer, (1973) and Schimidt (1998). These data will be used to train the artificial neural networks and to identify a flight envelope.

3. Artificial Neural Networks

Artificial neural networks are information processing systems with the capability of learning through examples (Haykin, 1994). Based on concepts derived from neuro-biology, neural networks are composed by a set of interconnected processing units, called *neurons*. The neurons process the signals presented to the neural network by accumulating each stimulus and by transforming the total value using a function; that is, the *activation function*. The stimulus to and from a neuron are modified by the real value called *synaptic weight*, which characterises the respective connection between neurons.

Figure 1 shows a typical representation for a generic neuron j, where $x_1, x_2, \ldots, x_p$ are the stimulus signals, $w_{j1}, w_{j2}, \ldots, w_{jp}$, are the synaptic weights, θ_j is a bias value, v_j is the activation potential, o_j is the neuron output signal, and $\varphi(.)$ is the activation function (generally adopted as a non-linear sigmoid function).

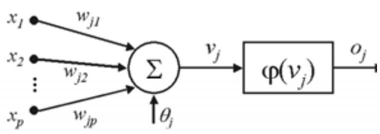


Figure 1. Typical neuron representation.

Then, from Figure 1, one can observe that the neuron output is given by:

$$o_j = \varphi\left(\theta_j + \sum_{i=1}^{p} w_{ji} x_i\right) \tag{25}$$

Network *architecture* is the name given to the arrangements of neurons into layers and how they are connected. Typical neural networks have the following architecture: (1) *input layer* – where the input stimulus is presented to the network; (2) *hidden layers* – internal layers of a network, and (3) *output layer* – the last layer of the network, where the outputs are given. Such typical network architecture is commonly referred to as a *multi-layer neural network*.

Once trained, one can assume that the network stored the knowledge supplied to it. However, the knowledge in a neural network is not stored in a particular localization. It depends on its topology and the magnitude of the weights in the input layer.

The generalization of an artificial neural network is the capacity to reproduce desired signals for different input signals that have not been used during the network training, or either, that it is able to catch the dynamics of the system being emulated (Saravanan and Duyear, 1994).

3.1 Recurrent Neural Networks (RNN)

During the last years the use of neural networks in dynamic systems modeling has increased significantly. This is justified by its parallel processing capacity, its ability to approach functional relationship, specifically the non-linear ones, the learning capability and its implementation easiness. Typical neural networks can only deal with input-to-output mappings that are static and a solution to this case has been given by using the idea of regressive models, in other words, models based on past values of the system input and output.

Recurrent networks (RNN) are neural networks with one or more feedback connections that can be of local or global nature. Feedback allows the recurrent networks to acquire state representations, making them appropriate devices for different dynamic applications such as: forecasting or modeling non-linear systems, adaptive equalization of communication channels, control of industrial installations, diagnostic of automotive engines and processing of temporal signals as the voice signal (Haykin, 1994).

In RNN's both feedforward and feedback (recurrent) connections between neurons are allowed (Kling, 2003). As with ordinary multilayer perceptrons, recurrent multilayer perceptrons can perform any nonlinear mapping, but the difference is that the response to an input from a recurrent network is now based on all previous inputs, as these are used in feedback connections. Nonetheless, the recurrent network is a dynamic system, with the activations of the neurons with feedback connections being the state of the system.

The output of a RNN network is a function of the current external input together with its previous inputs and outputs as given by:

$$y(k) = f(u(k), u(k-1), \dots, u(k-M), y(k-1), y(k-2), \dots, y(k-N)) \tag{26}$$

3.2 Neural Network Training

To achieve a desirable set of synaptic weights to a pre-defined network architecture, a training process is needed. A training process is generally based on an optimisation scheme to adjust the network parameters (mainly, the weights) in relation to a set of input-to-output to be matched by the neural network model (supervised learning scheme). The backpropagation algorithm based on a gradient descent technique (Haykin, 1994) has been widely applied for general neural network training. More efficient training scheme can be achieved by using the Levenberg-Marquardt Algorithm (LMA).

3.2.1 Levenberg-Marquardt Algorithm (LMA)

This algorithm is a variation of the Newton's method for minimizing functions that are sums of squares of other non-linear functions (Hagan *et al.*, 1996). The LMA provides better performance when compared with typical backpropagation algorithms.

From Newton's method the network update rule is:

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \mathbf{H}_n^{-1}\mathbf{g}_n \, , \tag{27}$$

where, $\mathbf{w}$ is the network weight matrix, n is a step of iteration, $\mathbf{H}$ is the Hessian matrix and $\mathbf{g}$ is the gradient matrix.

For the performance index as a sum of squares functions, the Hessian matrix can be approximated in terms of the Jacobian matrix, $\mathbf{J}$, which contains first derivatives of the network errors with respect to the weights and biases. Thus,

$$\mathbf{H} \cong \mathbf{J}^{\mathrm{T}}\mathbf{J} \, . \tag{28}$$

When the approximation in Eq. (28) is substituted into Eq. (27), the Gauss-Newton method is obtained, that is:

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \left[\mathbf{J}_n^{\mathrm{T}}\mathbf{J}_n\right]^{-1}\mathbf{g}_n \, . \tag{29}$$

A problem that may arise in the Gauss-Newton method is that the matrix [$\mathbf{J}^{\mathbf{T}}\mathbf{J}$] may not have an inverse. This can be overcome by assuming a modification to the matrix [$\mathbf{J}^{\mathbf{T}}\mathbf{J}$] that leads to the LMA:

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \left[\mathbf{J}_n^{\mathrm{T}}\mathbf{J}_n + \mu_n\mathbf{I}\right]^{-1}\mathbf{g}_n\,, \tag{30}$$

where, I is the identity matrix and μ is a scalar.

The scalar μ presents an important role to the LMA. When μ_n is zero, the weight update is basically the Gauss-Newton method. When μ_n is sufficiently large, Eq. (30) becomes gradient descent with small step size. By choosing the proper value of μ the LMA provides an efficient compromise between the great performance of the Newton's method and the guaranteed convergence of the gradient descent approach.

4. Identification of flight envelope using recurrent neural networks

After the non-linear mathematical model of the aircraft having been developed, it was implemented in C++. 7 flight conditions had been considered for simulation of the longitudinal dynamics with 3 simulations for each were done considering different values of steps as inputs in order to obtain significant sets of examples for the training of the networks. The representative scheme of the flight envelope of the aircraft under consideration is presented in Figure 2 as follows.

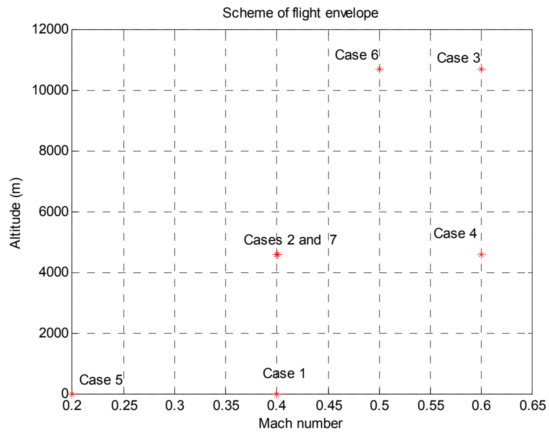


Figure 2. Scheme of flight envelope to be identified.

The red points in Figure 2 refer to the altitudes and Mach number for each one of the known conditions. The neural networks will be trained using results of the aircraf simulation with corresponding data to three of these points. Next, the network will be tested in order to verify if it identifies the other 4 flight conditions that it does not know. The results obtained for the *u* velocity of the aircraft will be shown.

During the simulation of the aircraft, elevator input steps of 0.5°, 1.5° and 2° had been entered for each one of the conditions. Multilayer neural networks with feedback of the output layer were trained with data corresponding to three conditions of the flight envelope and the other 4 data sets were kept to be used in generalization tests.

For the identification of the responses of horizontal velocities a network with topology 6-8-1 was trained using pairs of input-output data being the input signal an elevator deflection δ_E associated to static inputs of Mach number and air density. A delay for each dynamic input will be used. To train the networks, the corresponding simulation data of Cases 3, 5 and 7 of the flight envelope had been input simultaneously to them. Points 3, 5 and 7 were chosen because they represent distinct cases of the envelope. Each input was associated to the output response *u.* The error decline reached 10^{-5} order and is shown in Figure 3. A microcomputer with 1,4 MHz Pentium processor was used and the training time has been around 5 minutes.

After the network having been trained various generalization tests have been done.

The results found in a first test are shown in Figure 4 to follow. For the accomplishment of this test, the network was simulated, using the data of the cases with which it was trained, but having a step input of 1.8° for the three conditions after 10 seconds of the beginning of the simulation. It could be observed that it captured the dynamics of the cases in question since the results were shown to be satisfactory.

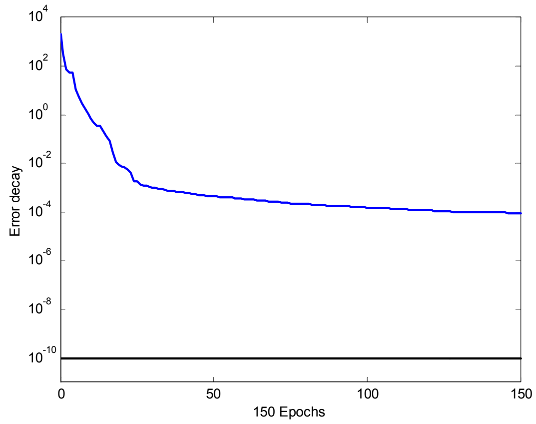


Figure 3. Error decay after training of the recurrent neural network.

Generalization tests had been carried out for other flight conditions and the responses data is presented to verify if the net identified the aircraft behavior. To diminish the computational cost, a network for identifying the horizontal velocities will be trained, another one for the vertical velocities and another for *theta*. In this work the result of the identification of the horizontal velocities for the cases presented in the flight envelope will be presented only.

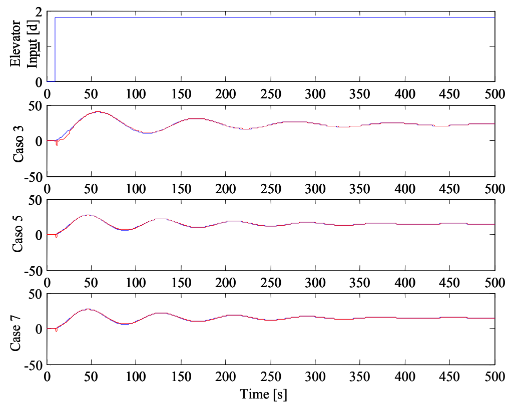
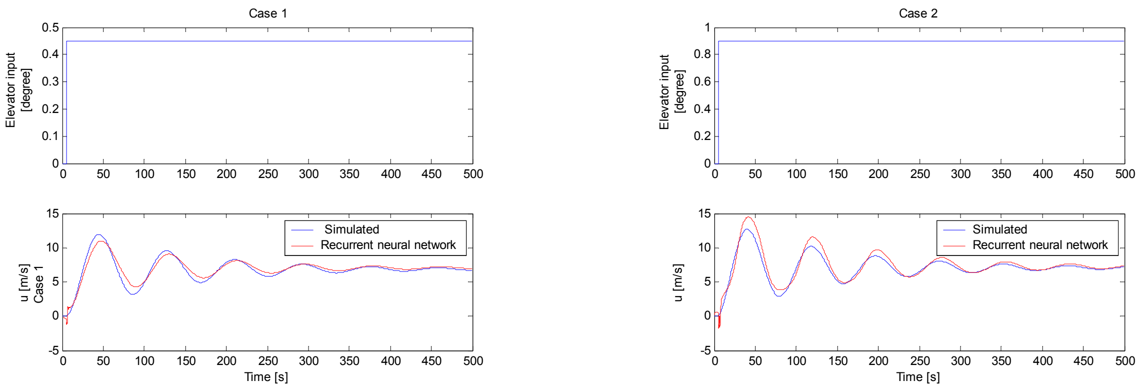
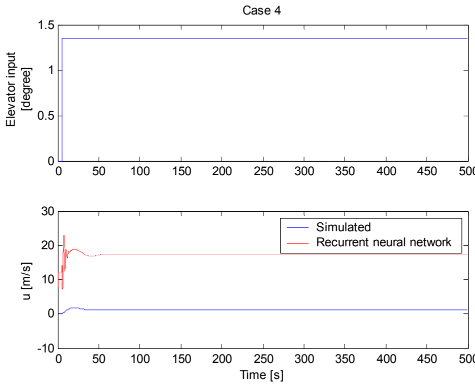


Figure 4. Results of a first test of the neural network generalization.

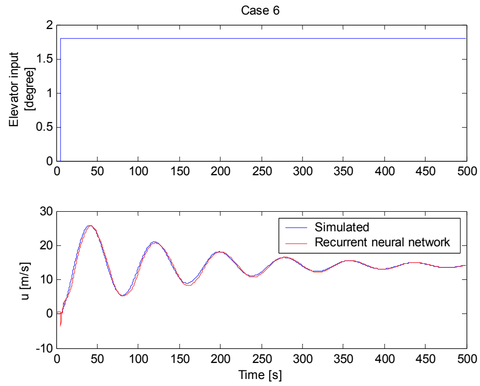
Several other generalization tests had been carried out, however, in this work only some will be shown. In the generalization test, presented next, the objective is to verify if the network identifies u for the other 4 flight conditions of the A4-D aircraft, which the net does not know. The results are shown in Figure 5.



(a) (b)



(c)



(d)

Figure 5. Generalization tests results of the neural network, in terms of u, in the points relative to cases 1, 2, 4 and 6, where the network does not know the responses.

5. Conclusions

In this work, a modeling and an implementation of an aircraft simulator has been shown to get aircraft responses relative to some flight conditions. It was also shown the implementation of recurrent neural networks for identification of a specific flight envelope of the A4-D aircraft. Some simulation responses of the A4-D aircraft in some points of the flight envelope were used for training the neural networks and others used to verify if the neural networks were capable to identify the aerodynamic behavior of the aircraft in points of the envelope that they did not know. For the development of the mathematical model, the non-linear equations of motion presented in Etkin and Reid, (1996) were used, and after the mathematical model was built, it was implemented in C++ language. The simulation results shown that the simulator was sufficiently representative of the real aircraft once comparing the simulated responses with results observed in literature and from flight tests. It was also shown that the recurrent artificial neural networks using constant and dynamic inputs, after correctly trained, identified points of a flight envelope built varying values of altitudes and velocities. The neural network did not identify u for case 4, since it is the response with bigger damping. New tests are being carried out and the results will be presented in a next work.

6. References

Etkin, B. and Reid, D.R., 1996, “Dynamics of flight: stability and control”, – 3rd ed, John Wiley and Sons.

Hagan, M.T.; Demuth, H.B.; Beale, M., 1996 ,“Neural Network Design”. PWS Publishing Co..

Haykin, S., 1994, “Neural Network: a Comprehensive Foundation”, Macmillan College Publishing Company, New York.

Iliff, K.W. and Wang, K.C, 1997, “Extration of Lateral-Directional Stability and Control Derivatives for the Basic F-18 Aircraft at High Angles of Attack”. Nasa Technical Memorandum 4786.

Klein, V., 1989, ”Estimation of aircraft aerodynamic parameters from flight data”. Prog. Aerospace Sci., Vol. 26, pp. 1-77.

Kling, R., 2003, “An Implementation of Recurrent Neural Networks for Prediction and Control of Nonlinear Dynamic Systems”. MSc Thesis. Monash University in Melbourne in Australia.

Linse, D.J. and Stengel, R., 1993, “Identification of Aerodynamic Coefficients Using Computational Neural Networks”. Journal of Guidance, Control and Dynamics. Vol. 16, No. 6, pp. 1018-1025.

Marques, F.D, 2003, “Dynamic Flight Introduction, Formulation and Simulation: Non-linear Problems and Applications”. Anais of DINCON, Vol. 2, pp. 38-61.

McRuer, D., Ashkenas, I. and Graham, D., 1973, “Aircraft Dynamics and Automatic Control”. Princenton University Press. Princeton, New Jersey.

Mendonça, C.B. and Góes, L. C.S, 2003, “Airplane Parameter Identification Using Frequency Response Error Method”. 17 th International Congress of Mechanical Eingineering. Proceedings of Cobem, Rio de Janeiro, Brazil.

Morelli, E.A., 1999, “Real – Time Parameter Estimation in the Frequency Domain”. AIAA-99-4043.

Raisinghani, S.C.and Ghosh, A. K., 2000, “Parameter estimation of an aeroelastic aircraft using neural networks”. Sãdhanã, Vol. 25, pp. 181-191.

Saravanan, N.and Duyar, A., 1994, “Modeling Space Shuttle Main Engine Using Feed-Forward Neural Networks”. Journal of Guidance, Control and Dynamics, Vol. 17, No. 4, pp.641-648.

Schimidt, L.V., 1998, “Introduction to Aircraft Flight Dynamics”. American Institute of Aeronautics, Inc., Reston, Virginia.