AIRCRAFT PARAMETER ESTIMATION USING ADAPTIVE FUNCTIONAL LINK NETWORK

Wilson Rios Neto
Instituto Tecnológico de Aeronáutica/ITA-CTA
12228-900 – São José dos Campos – SP, Brazil
wriosnet@ita.br

Marcelo Curvo
Empresa Brasileira de Aeronáutica/EMBRAER
12227-901 – São José dos Campos – SP, Brazil
marcelo.curvo@embraer.com.br

Luiz Carlos Sandoval Góes
Instituto Tecnológico de Aeronáutica/ITA-CTA
12228-900 – São José dos Campos – SP, Brazil
goes@ita.br

Abstract. *Certification, control law project, and the flight simulation are essential tasks in the aeronautical industries that require good mathematical modeling of aircraft dynamics. More realistic aircraft models, described by the aerodynamic and control coefficients, can be determined during flight tests programs. The objective of this work is the use of Artificial Neural Network that makes near real time aircraft coefficients estimation achievable. The network architeture employed is called Adaptive Functional Link Network (AFLN), this type of architeture allows the attachment of phsical meaning to its parameters (weights). Also this network is linear relative to its parameters, which implies that it can be trained in the least square sense. An adaptive scheme is used to avoid divergence, common in problems of least square estimators, when processing large data sets. Tests results of the AFLN application are shown through aerodynamic coefficients of the aircraft flight simulator, using simulated realistic data of an aircraft, whose aerodynaimic model and data is avaliable.*

Keywords*: Aerodynamic Coefficients Estimation, Adaptive Functional Link Network, Kalman Filter Algorithm, Optimal Linear Estimation*

1. Introduction

One of the important branches of modeling and simulation (M&S) is the system identification/parameter estimation applied to complex dynamic systems, such as an aircraft. System identification and parameter estimation are used to characterize the response of any dynamic systems, either considering an input/output black box approach or assuming the equation of motion known and estimating the unknown parameters.

Artificial Neural Network (ANN) are universal function modeler, able to approximate an unknown model to any desired accuracy, this characteristic have drawn the attention of scientists and engineers for the application of such tool to the problem of system identification and parameter estimation. In general, the parameters (weights) of an ANN have no physical meaning, also, the learning rules used to find such parameters are nonlinear, making it to costly from the computational point of view, and difficult to apply in near real time parameter estimation problem.

In this work, a network architecture called Adaptive Functional Link Network (AFLN) has been applied to the problem of aircraft parameter estimation. When compared with the classical networks (Multi Layer Perceptron), there are main benefits offered by the AFLN: (1) The nonlinear transformation of the network input is done through different types of nonlinear functions, such as polynomials functions, exponentials, and radial base functions. This characteristic gives the AFLN an enormous flexibility and makes the determination of the base functions an important phase during the network project (Curvo, 2000). (2) The network first process the nonlinear transformation and then combine the result in a linear manner, keeping it linear with respect to the parameters. This characteristic allow the use of linear training rules, such as Kalman filter (Rios Neto et al., 1985). (3) And to improve the problem of least squares type of estimators that tends to diverge when many data are processed, an adaptive procedure based on a criterion of statistical consistency is used to balance a priori information priority with that of a new learning information (Rios Neto, 1997).

The Functional Link Network (FLN) (Pao, 1992) presents the first two benefits, but it is not a robust estimator. If the initial parameter vector or its initial estimation error covariance matrix is far from the real parameter values, the FLN does not perform as well as an AFLN. The AFLN network was proposed to circumvent this weakness, applying a procedure to estimate the noise level adaptively in order to give the algorithm a good numerical behavior and the ability to distribute the extraction of learning information to all training data, improving the convergence and the capacity of learning by a substantial amount.

2. Adaptive Functional Link Network (AFLN)

The AFLN architecture, as shown in Fig. 1, is arranged in three layers: an input layer, one hidden layer, and the output layer. The hidden layer is responsible for the mapping of the input space $\Re^n$ to another of dimension p, that is:

$$x \in R^n \to \left[h_1(x)...h_p(x)\right]^T \in \Re^p \tag{1}$$

where $h_i(x)$, $i = 1,2,..., p$ is the base function set that depends exclusively of x. The output layer has m nodes, each one a linear combination leading to the following input/output mapping of the network, with $1 \le j \le m$:

$$\hat{f}_j(x, w) = \sum_{k=1}^{p} w_{jk} h_k(x) \tag{2}$$

The AFLN can approximate any continuous function with desired precision (Narendra et al., 1990), if the set of base function $h(x)$ selected is representative or composed by a sufficient number of high order terms. One point that should be emphasized is that the network is linear in the parameters (weights), which mean that they can be determined using linear training rules.

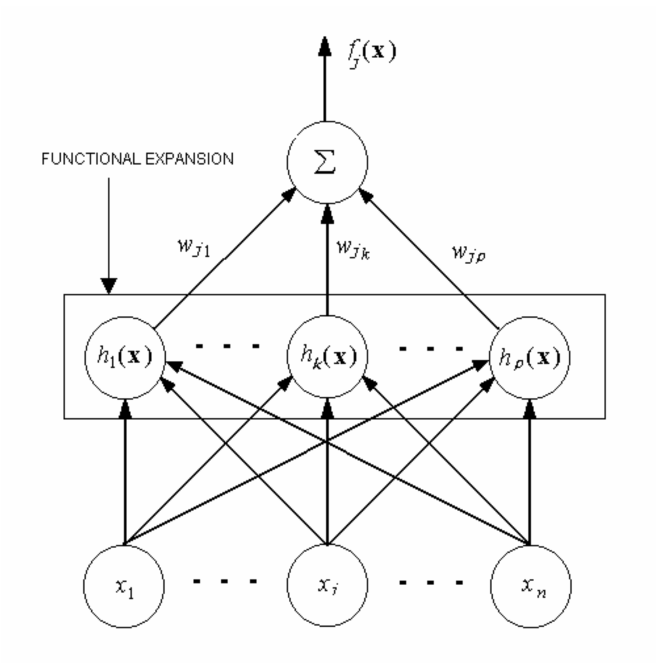

Figure 1. Adaptive Functional Link Network Architecture
Source: Curvo (2001)

2.1. Optimal Linear Estimation Procedure

As previously stated, an AFLN should achieve a continuous nonlinear mapping, and it is assumed that the network has the following mapping and definition:

$$f \in C : x \in D \subset \Re^n \to y \in \Re^m \tag{3}$$

$$\hat{y}(t) = \hat{f}\left[x(t), w\right] \tag{4}$$

where w is the weight vector (network free parameters) to be estimated from an input/output training set. During the training process, the minimization of the following functional is solved for each pair of input/output set j:

$$J_j(\mathbf{w}_j) \equiv \frac{1}{2}\left[\left(\mathbf{w}_j - \overline{\mathbf{w}}_j\right)^T \overline{\mathbf{P}}_j^{-1}\left(\mathbf{w}_j - \overline{\mathbf{w}}_j\right)\right] + \frac{1}{2}\sum_{t=1}^{T}\left[\left(\mathbf{y}_j(t) - \mathbf{H}(x(t))\mathbf{w}_j\right)^T \mathbf{R}_j^{-1}(t)\left(\mathbf{y}_j(t) - \mathbf{H}(x(t))\mathbf{w}_j\right)\right] \tag{5}$$

where $\overline{w}_j$ is the initial estimate vector of w_j, $\overline{P}_j$ is the statistic representing the quality of the initial estimation of the parameters, and R_j is the statistic representing the quality of the output pattern of the training set. This is equivalent to the following stochastic linear parameter estimation problem (Rios Neto, 1997), considering $t = 1,2,...,T$ and $j = 1,2,...,m$:

$$\begin{aligned} \overline{\mathbf{w}}_j &= \mathbf{w}_j + \overline{\mathbf{e}}_j \\ \mathbf{y}_j(t) &= \mathbf{H}(\mathbf{x}(t))\mathbf{w}_j + \mathbf{v}_j(t) \end{aligned} \tag{6}$$

$$\begin{aligned} &E\left[\mathbf{v}_j(t)\right] = 0 \qquad && E\left[\mathbf{v}_j^{\,2}(t)\right] = \mathbf{R}_j(t) \\ &E\left[\overline{\mathbf{e}}_j\right] = 0 && E\left[\overline{\mathbf{e}}_j \cdot \overline{\mathbf{e}}_j^{\,T}\right] = \overline{\mathbf{P}}_j \end{aligned} \tag{7}$$

This system can be solved by a linear estimation algorithm, i.e., Kalman filter algorithm (Gelb, 1996).

$$\begin{aligned} \mathbf{K}_j(t) &= \overline{\mathbf{P}}_j \cdot \mathbf{H}_j^{\,T}(x(t))\left[\mathbf{R}_j(t) + \mathbf{H}_j(\mathbf{x}(t))\overline{\mathbf{P}}_j\mathbf{H}_j^T(\mathbf{x}(t))\right]^{-1} \\ \mathbf{P}_j(t) &= \left[\mathbf{I}_j - \mathbf{K}_j(t)\mathbf{H}_j(x(t))\right]\overline{\mathbf{P}}_j \\ \hat{\mathbf{w}}_j &= \overline{\mathbf{w}}_j + \mathbf{K}_j(t)\left(\mathbf{y}_j(t) - \mathbf{H}_j(x(t))\overline{\mathbf{w}}_j\right) \end{aligned} \tag{8}$$

2.2. Adaptive Optimal Linear Estimation Procedure

One problem of the optimal linear estimation, in particular Kalman filter, is that it has bad numerical behavior and observation model errors, so divergence usually occurs as many data sets are processed. These problems cause an excessive and unrealistic decrease of the estimated dispersions of the covariance in the calculated estimates; i.e., a matrix of estimated covariance of the errors, in the estimates, with very small eigenvalues. In order to avoid this ill behavior and try to keep a distributed and uniform capacity of learning, an adaptive procedure, based on a criterion of statistical consistency, to balance a priori information priority with that of new learning information, the following procedure is applied (Rios Neto, 1997). Considering that in typical jth iteration for $t = 1,2,...,T-1$.

$$w_j = w_j + \eta(t) \tag{9}$$

$$E[\eta(t)] = 0, E[\eta(t)\eta^T(\tau)] = Q(T)\delta_{t\tau} \tag{10}$$

$$Q(t) = diag[q_j(t) : j = 1,2,...,n_w] \tag{11}$$

where $\delta_{t\tau}$ is the Kronecker delta, and n_w is the number of weight parameters, the approximation of the neural weight learning, using the jth input/output training set is transformed in the following estimation problem:

$$\begin{aligned} \overline{\mathbf{w}}_j &= \mathbf{w}_j + \overline{\mathbf{e}}_j(t) \\ z_j(t) &= \mathbf{H}_j\left(x(t)\right)\mathbf{w}_j(t) + \mathbf{v}_j(t) \end{aligned} \tag{12}$$

considering $\overline{e}_j(1) = \overline{e}_j$ and $t = 1,2,...,T$.

To propagate the estimation of the Kalman filter predictor from t to $t+1$, Eq. (9) is used:

$$\begin{aligned} \overline{w}_j &= \hat{w}_j \\ \overline{P}_j(t+1) &= P_j(t) + Q(t) \end{aligned} \tag{13}$$

where $\overline{P}_j(t+1) = E\left[\overline{e}_j(t+1)\overline{e}_j^{\,T}(t+1)\right]$ and $P_j(t)$ is given by the filter algorithm, Eq. (8). $\overline{P}_j(t)$ starts with $\overline{P}_j(1) = E\left[\overline{e}_j \overline{e}_j^{\,T}\right] = \overline{P}$. The adaptation is done by adjusting the noise $\eta(t)$ dispersion such as to keep the consistency to achieve distributed learning:

$$\beta E\left[\nu_j^{\,2}(t+1)\right] = H_j(t+1)\left[P_j(t) + Q(t)\right]H_j^T(t+1) \tag{14}$$

where β is to be adjusted closer to one, for a distributed learning. In order to use the optimal linear estimation procedure, i.e., Kalman filter algorithm, the following associated estimation problem can be established by considering Eq. (14) as a pseudo observation, and imposing an a priori information such that $q(t)$:

$$0 = q(t) + \overline{e}^{\,q} \tag{15}$$

$$z_j^{\,q}(t+1,\beta) = H_j^{\,q}(t+1)q(t) + \nu^q(t+1) \tag{16}$$

$$E\left[\overline{e}^{\,q}\right] = 0 \qquad E\left[\overline{e}^{\,q}\overline{e}^{\,q^T}\right] = I_{n_w} \tag{17}$$

$$E\left[\nu^q(t+1)\right] = 0 \qquad E\left[\nu^q(t+1)\nu^{q^T}(t+1)\right] = R^q(t+1) = 0 \tag{18}$$

The problem above is, in fact, one of exact observation and is possible to be solved by a Kalman filter algorithm (Freitas Pinto et al., 1990). It should be noticed that the solution gives a $\hat{q}(t)$ close to zero in magnitude. Whenever a $q_k(t)$ component is less than zero it is disregarded and taken to be zero, since a positive condition must be observed.

3. Aircraft Dynamic System

To illustrate the proposed technique, a case study of a longitudinal dynamic model using the model of an F-16 aircraft will be shown. The aerodynamic model used is a classical one for simulation, performance, stability, and control analysis (Steven et al., 1992).

Lift Force:

$$L = \overline{q} S_w CL \tag{19}$$

$$CL = CL_0 + CL_\alpha \alpha + CL_{\dot\alpha} \frac{1}{2}\left(\frac{\dot\alpha\,\overline{c}_W}{V}\right) + CL_{\delta_e}\delta_e \tag{20}$$

Drag Force:

$$D = \overline{q} S_w CD \tag{21}$$

$$CD = CD_0 + CD_\alpha \alpha + CD_{\dot\alpha} \frac{1}{2}\left(\frac{\dot\alpha\,\overline{c}_W}{V}\right) + CD_{\delta_e}\delta_e \tag{22}$$

Pitching Moment:

$$M = \overline{q} S_w \overline{c}_w Cm \tag{23}$$

$$Cm = Cm_0 + Cm_\alpha \alpha + Cm_{\dot\alpha} \frac{1}{2}\left(\frac{\dot\alpha\,\overline{c}_w}{V}\right) + Cm_q \frac{1}{2}\left(\frac{q\,\overline{c}_w}{V}\right) + Cm_{\delta_e}\delta_e \tag{24}$$

The dynamic model is a nonlinear longitudinal model, described by the following set of equations and state vector. For the geometric, mass and aerodynamic data of the airplane used in the calculations see Steven et al. (1992).

$$\begin{aligned}\dot{V} &= T\cos\alpha - D - g\sin(\theta-\alpha)\\ \dot{\alpha} &= -(T/V)sen\alpha - (L/V) - (g/V)\cos(\theta-\alpha)\\ \dot{q} &= (M/I_{yy})\\ \dot{\theta} &= q\end{aligned} \tag{25}$$

$$\mathbf{x}^T = [V \quad \alpha \quad q \quad \theta] \tag{26}$$

where:

T - Engine thrust;
g - Acceleration of gravity;
I_{yy} - Moment of inertia about lateral axis y;
$\bar{q}$ - Dynamic pressure;
S_w - Wing area;
$\bar{c}_w$ - Mean aerodynamic chord of the wing;
V – Airspeed;
α - Angle of attack;
q - Pitch velocity;
θ - Pitch angle;
δ_e - Elevator deflection.

4. Estimation of Aerodynamic Derivatives

Instead of flight data, simulated data is used to estimate the following aerodynamic derivatives: CL_0, CL_α, $CL_{\dot{\alpha}}$, CL_q, CL_{δ_e}, Cm_0, Cm_α, $Cm_{\dot{\alpha}}$, Cm_q, and Cm_{δ_e}. The use of simulated data does not invalidate the results, because the objective is to demonstrate the use of AFLN in coefficients estimation.

The neurons of the input layer of the network are formed by: airspeed (V), pitch velocity (q), angle of attack (α), elevator deflection (δ_e), and sideslip angle (β) and each of the network neurons of the output layer, $\hat{f}_j(\mathbf{x},\mathbf{w}) \quad j = 1,2,...,m$ is formed by ordinary polynomials:

$$\mathbf{CA}_j = \sum_{k=1}^{p} w_{jk} h_k(\mathbf{x}) \qquad j = 1, 2, ..., m \tag{27}$$

For example, the lift force coefficient (CL) may be represented as a function of angle of attack, airspeed, pitch velocity, elevator deflection, etc.

$$CL = CL(\alpha, q\bar{c}_w/2V, \delta_e) = w_{10} + w_{12}\Delta\alpha + w_{13}\Delta\delta_e + w_{14}\, q\bar{c}_w/2V + ... + O^S \tag{28}$$

where the network parameters are represented by the aerodynamic coefficients, giving physical meaning to the network parameters (weights). This same model is used for the others coefficients.

An important aspect of this estimation technique is the modeling of the noises inherent to the processes. For the parameters to be estimated the following criterion, suggested by Bauer et al. (1990), was considered $\bar{P}_{jk} \approx \left(\frac{1}{4}\, w_{jk}\right)^2$. These values may be subject to some adjustment after the first estimates, and as guidelines the residuals are used.

For parameters with initial values equal to or close to zero, the values are established based on engineering judgment. For the observation noise the criterion adopted was the use of an expected standard deviation of the estimation.

5. Results and Discussion

The results obtained for the application were very satisfactory. Compared to the FLN procedure, the AFLN showed to be a good estimator even when the initial guesses of the weights vector or/and estimation error covariance are far from the real parameter values, i.e., the estimation response was better. Two case studies are presented: (1) initial guess closer to the real values; and (2) initial guess considerably far to the real values, to demonstrate the strength of AFLN.

Considering the initial weight vector corrupted by ±15% of the true values, and the error covariance matrix given by $\overline{P}_{jk} \approx \left(\frac{1}{4} w_{jk}\right)^2$ (see Table 1 and 2, respectively), the answer was equivalent for FLN and AFLN network. The results are showed in Fig. (2) and Fig. (3) for true, estimated, and optimal estimation (after a pruning heuristic is applied) of the total lift coefficient, and Fig. (4) and Fig. (5) for true, estimated, and optimal estimation of the total moment coefficient.

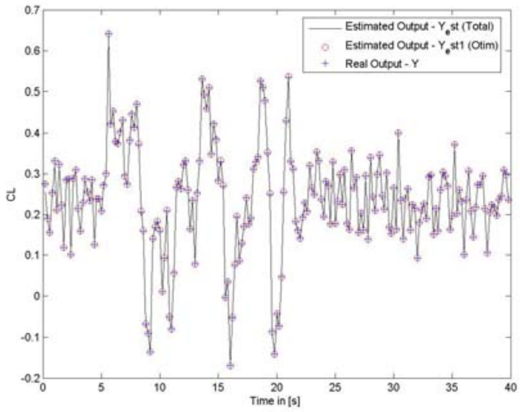

Figure 2. Time History of Total Lift Coefficient FLN

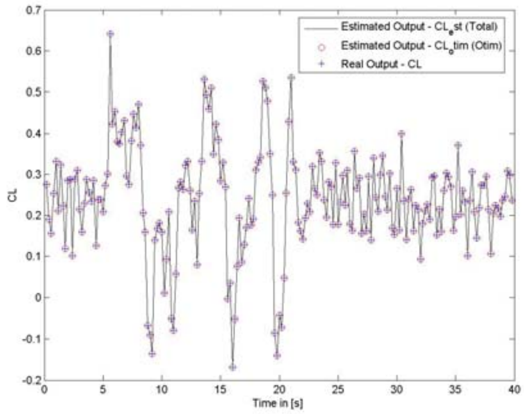

Figure 3. Time History of Total Lift Coefficient AFLN

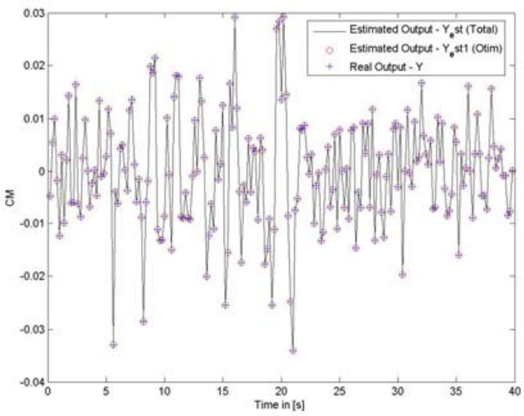

Figure 4. Time History of Total Moment Coefficient FLN

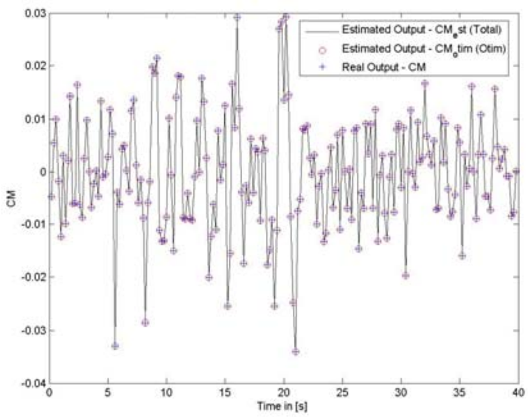

Figure 5. Time History of Total Moment Coefficient AFLN

Table 1 – Aerodynamic Coefficients: True, Initial, Optimized FLN, and Optimized AFLN

	CL_0	CL_α	$CL_{\dot\alpha}$	CL_q	CL_{δ_e}	Cm_0	Cm_α	$Cm_{\dot\alpha}$	Cm_q	Cm_{δ_e}
True Values	***0.30***	***3.45***	***1.12***	***0.00***	***0.40***	***0.00***	***-0.41***	***-1.65***	***-4.30***	***-0.60***
Initial Values	***0.255***	***3.968***	***0.952***	***0.000***	***0.460***	***0.000***	***-0.349***	***-1.898***	***-3.666***	***-0.690***
Optimized FLN	***0.302 ± 1.86e-4***	***3.561 ± 4.58e-3***	***0.994 ± 5.60e-2***	—	***0.379 ± 9.96e-4***	—	***-0.422 ± 1.00e-8***	***-1.933 ± 5.81e-3***	***-3.182 ± 6.21e-3***	***-0.606 ± 5.76e-5***
Optimized AFLN	***0.300 ± 5.20e-3***	***3.538 ± 1.12e-2***	***0.953 ± 5.66e-2***	—	***0.456 ± 5.57e-3***	—	***-0.422 ± 2.24e-3***	***-1.933 ± 7.13e-3***	***-3.182 ± 1.03e-2***	***-0.606 ± 2.79e-3***

Table 2 – Initial Parameter Variance and Measurement Noise Data of the Estimators

P0									
CL_0	CL_α	$CL_{\dot{\alpha}}$	CL_q	CL_{δ_e}	Cm_0	Cm_α	$Cm_{\dot{\alpha}}$	Cm_q	Cm_{δ_e}
3.164e-3	***9.841e-1***	***5.664e-2***	***1.000e-3***	***1.323e-2***	***1.000e-8***	***7.613e-3***	***2.252e-1***	***9.165e-1***	***2.976e-2***
R0									
CL					*Cm*				
0.735e-2					***0.540e-4***				

For initial guesses far from reality (very common in real flight test), the optimal linear estimation algorithm (FLN) tends to diverge. Especially if important coefficients, such as $CL_{\dot{\alpha}}$, CL_q, Cm_0, Cm_α, and Cm_{δ_e} are not accounted for. This makes the application of FLN impracticable to near real time identification, since the control system of an aircraft needs an accurate set of coefficients to keep its flying quality level.

As shown below, the pruning heuristic (same for both algorithms) applied to FLN algorithm eliminated essentials coefficients, and consequently the coefficients estimation was erroneous. The adaptive optimal linear estimation algorithm (AFLN) showed better behavior and robustness than the original FLN. The estimated coefficients led to a good network response, after the pruning process called OBS was applied (Hassibi et al., 1993).

The AFLN algorithm demonstrated good identification properties (good coefficient estimation) and good network response. The algorithm demonstrated a better learning capacity, no matter if the number of processed data is increasing or not. Results for the case where AFLN demonstrate an improvement over the FLN algorithm, for a divergent problem, are shown.

Table 3 – Aerodynamic Coefficients: True, Initial, Optimized FLN, Optimized AFLN

	CL_0	CL_α	$CL_{\dot{\alpha}}$	CL_q	CL_{δ_e}	Cm_0	Cm_α	$Cm_{\dot{\alpha}}$	Cm_q	Cm_{δ_e}
True Values	***0.30***	***3.45***	***1.12***	***0.00***	***0.40***	***0.00***	***-0.41***	***-1.65***	***-4.30***	***-0.60***
Initial Values	***0.051***	***0.794***	***0.190***	***0.00***	***0.092***	***0.00***	***-0.070***	***-0.380***	***-0.733***	***-0.138***
Optimized FLN	***0.303 ± 1.79e-4***	***3.407 ± 4.53e-3***	—	—	—	—	—	***-1.814 ± 2.00e-9***	***-2.168 ± 1.32e-3***	—
Optimized AFLN	***0.305 ± 5.21e-3***	***3.536 ± 1.21e-2***	***0.190 ± 1.13e-2***	—	***0.094 ± 5.48e-3***	—	***-0.421 ± 2.24e-3***	***-1.378 ± 1.84e-3***	***-4.086 ± 9.49e-3***	***-0.606 ± 2.80e-3***

Table 4 – Initial Parameter Variance and Measurement Noise Data of the Estimators

P0									
CL_0	CL_α	$CL_{\dot{\alpha}}$	CL_q	CL_{δ_e}	Cm_0	Cm_α	$Cm_{\dot{\alpha}}$	Cm_q	Cm_{δ_e}
6.328e-4	***1.968e-1***	***1.133e-2***	***2.000e-4***	***2.646e-3***	***2.000e-9***	***1.523e-3***	***4.504e-2***	***1.833e-1***	***5.952e-3***

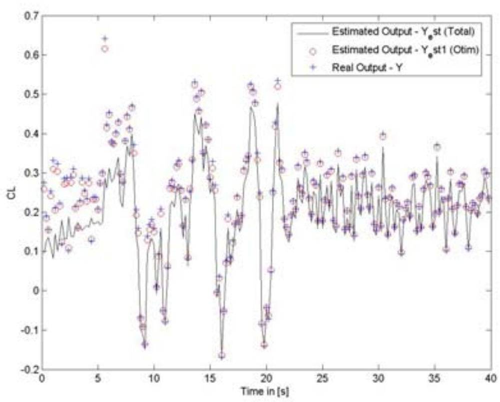

Figure 6. Time History of Total Lift Coefficient FLN

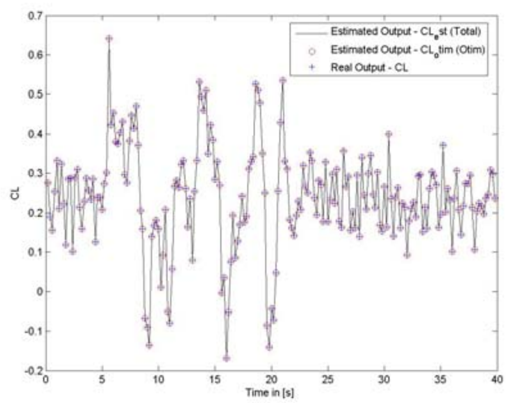

Figure 7. Time History of Total Lift Coefficient AFLN

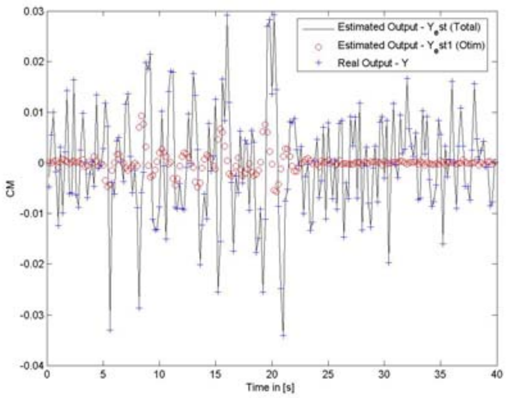

Figure 8. Time History of Total Moment Coefficient FLN

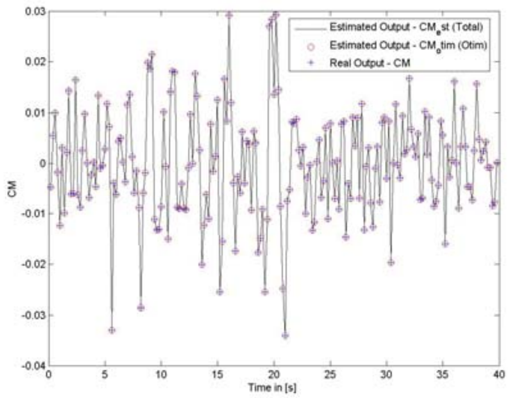

Figure 9. Time History of Total Moment Coefficient AFLN

6. Conclusions

This work presented an adaptive optimal linear estimation algorithm for the architecture known as Functional Link Network. This architecture has the advantage of attaching physical meaning at the free parameters of the network and allows the use of a linear training algorithm. Improvement of the divergence problem of the Least Squares type, i.e. Kalman filter, when many data are processed was the main result of this research and the main contribution was a substantial improvement over the performance of the FLN architecture, as far as parameter estimation is concerned. This algorithm can be applied to identification and control problems. It will provide the parameters estimation in near real time, even with initial guesses far from reality. The algorithm proposed is simple enough to be packed as a function and included in the flight control computer, so the estimation process could be conducted in near real time during flight test for certification, sensors/actuators calibration and stability and control coefficients calculation.

7. Acknowledgements

This work was only possible due to the support given by CNPq to the first author.

8. References

Bauer, J.E. and Andrisani, D., 1990, "Estimating Short-Period Dynamics using an Extended Kalman Filter", 5th Biannual Flight Test Conference, Ontario, Canada, May 21-24.

Curvo, M., 2000, "Estimation of Aircraft Aerodynamic Derivatives using Extended Kalman Filter", Journal of the Brazilian Society of Mechanical Sciences, Vol XXII (2), pp. 133-148.

Curvo, M., 2001, "Use of Functional Link Network in Aerodynamic Modeling of Aircraft", 16th Brazilian Congress of Mechanical Engineering, Uberlândia, Brazil, November 26-30.

Freitas Pinto, R.L.U. and Rios Neto, A., 1990, "An Optimal Linear Estimation Approach to Solve Systems of Linear Algebraic Equations", J. Computational and Applied Mathematics, 33, pp. 261-268.

Gelb, A., 1996, "Applied Optimal Estimation", The MIT Press, 14th edition.

Hassibi, B. et al., 1993, "Optimal Brain Surgeon and General Network Pruning", IEEE Int. Conference on Neural Networks, San Francisco, pp. 293-299.

Narendra, K.S. and Parthasarathy, K., 1990, "Identification and Control of Dynamical Systems using Neural Networks", IEEE Transactions on Neural Networks, Vol. 1 (1).

Pao, Y.H., Phillips, S.M. and Sobajic, D.J., 1992, "Neural-Net Computing and Intelligent Control Systems", Int. J. Contr., Vol. 56 (2), pp. 263-289.

Rios Neto, A. and Kuga, H.K., 1985, "Kalman Filtering State Noise Adaptive Estimation", Proceedings of Second IASTED Int. Conference in Telecom and Control, TELECON'85, Rio de Janeiro, Brazil, pp. 210-213.

Rios Neto, A., 1997, "Stochastic Optimal Linear Parameter Estimation and Neural Nets Training in System Modeling", RBCM-Journal of the Brazilian Soc. of Mechanical Sciences, Vol. 19 (2), pp. 138-145.

Stevens, B.L. and Lewis, F.L., 1992, "Aircraft Control and Simulation", New York, John Wiley&Sons, 2nd edition.