

MODELLING OF THE 3-DOF NONLINEAR AIRCRAFT BEHAVIOUR VIA SMALL STANDARD FORMS

Carsten DÖLL* , Caroline BERARD**,**

* ONERA The French Aerospace Lab, Department of System Control and Flight Dynamics
2, av Edouard Belin, BP 4025, F-31055 TOULOUSE, France

** ISAE (former SUPAERO) The French Institute for Aeronautics, Department of Control
10, av Edouard Belin, BP 4032, F-31055 TOULOUSE, France

Keywords: *LFT, modelling, nonlinear, aircraft*

Abstract

The objective of this paper is to illustrate that the standard form or Linear Fractional Transformation LFT framework is a unique and powerful framework to treat with nonlinear aircraft behaviour during flight control law synthesis and analysis. Within this paper, a precise and as small as possible LFT for a nonlinear 3-DOF aircraft with time varying parameters is derived. This LFT is validated both in a linear and nonlinear framework.¹

1 Introduction

A standard form $P(s) - \Delta$ or LFT representation of an uncertain or varying nonlinear system is a special system model where all well known and fixed linear system dynamics are put together in the so-called nominal transfer function $P(s)$ while the uncertain and varying parameters, as well as the nonlinearities, are stocked in the so-called perturbation matrix Δ . Both are interconnected via artificial inputs and outputs of the matrix $P(s)$ as shown in Figure 1.

The LFT form is needed for several purposes:

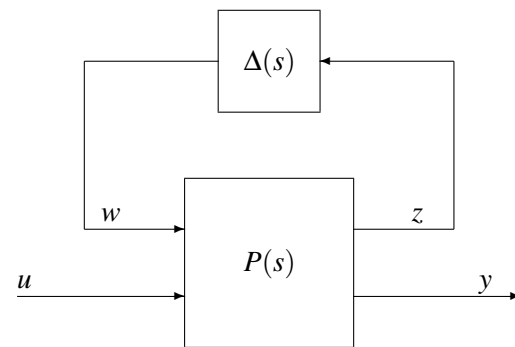


Fig. 1 The interconnection structure of an LFT

1. The most well known need is the use of LFTs during robustness analysis of designed control laws, especially using μ -analysis methods [7]. These LFTs are built to be able to account for parametric (or real) uncertainties as well as dynamic (or complex) uncertainties.
2. LFTs can also be used during robust control law designs, especially using \mathcal{H}_∞ approaches [5]. They are often built to take into account dynamic uncertainties or weighting functions.
3. LFTs are more and more used to model Linear Parameter Varying LPV systems, especially in the case of gain scheduling approaches [9]. These last LFTs are normally constructed such that the measurable scheduling variables appear in the Δ -

¹Corresponding author : carsten.doll@onera.fr, Tel.: +33 (0)5.62.25.29.20, Fax: +33 (0)5.62.25.25.64

matrix.

4. And finally, control law design and analysis in the presence of nonlinearities, especially in the presence of actuator saturations, gained a lot of interest in the last years [12]. Specific ways to model these saturations as LFTs were proposed. Anti-windup controller schemes are also proposed.

These four types of LFTs are different. The more physical parameters are used in an LFT for analysis purposes, the less conservative the analysis results will be. That is why analysis LFTs will make appear more and more uncertainties. On the other hand, the LFTs used for design purposes can use aggregated and/or artificial variables which can be computed from the physical parameters [8]. Hence, the size of an analysis LFT is often bigger than the one of a design LFT. However, the μ -analysis algorithms still have numerical problems when the size of the LFT is too big, where the need stems from to come up with precise LFTs of minimal size. It is usually very easy to develop an LFT like in Fig. 1, but to find its minimal realization is much more complicated [1]. For this purpose a MATLAB toolbox for LFT generation² was developed at ONERA [10]. It is usually not possible to obtain the minimal realization just by numerical post-realization order reduction. Already during the realization process, you have to pay attention to the way how you proceed [13, 6] which will be highlighted within this paper. For a straightforward implementation into Matlab/Simulink©, specific blocks have recently been added to the above toolbox. They allow the use of the linearization tools of Matlab and the simulation of LTI/LPV systems in standard form under Simulink³. Special blocks for actuator saturations and trigonometrical nonlinearities have also been added at

ONERA [2]. The use of these blocks will be demonstrated within this paper.

The paper is organized as follows. First, the proposed overall modelling procedure for nonlinear systems under LFT form is introduced. Then, the different steps are explained in more detail. The replacement of nonlinear tabulations as well as the substitution of exponential and trigonometrical functions by rational expressions is detailed. The advantage of using multivariate orthogonal functions [11] to come up with low order high fidelity expressions is illustrated. Symbolic processing, *i.e.* symbolic linearization and pre-processing, will then be highlighted before the LFT for the nonlinear aircraft with time varying parameters will be built. Finally, the LFT is validated by means of pole-charts, by frequency plots and by comparing time simulations of the LFT model with respect to simulations of the nonlinear 3-DOF aircraft.

2 The overall LFT modelling procedure

The LFR Toolbox for MATLAB has been developed at ONERA [10] in order to generate easily various LFTs of minimal or at least small size starting from a given model. Let us concentrate on LFT for analysis purposes. For conservativeness reasons, all physical dependencies have to be taken into account during the modeling phase, such that only completely independent parameters appear in the Δ -matrix.

The recommended analysis LFT generation procedure [6] is as follows:

1. Express tabulated data, irrational and trigonometrical functions by rational expressions
2. Write the nonlinear equations:

$$\dot{x} = f(x, u) \quad (1)$$

as symbolic expressions using all analytic dependencies using for example the Matlab

²Download is possible from <http://www.cert.fr/dcsd/idco/perso/Magni/index.html>

³Download is possible from <http://www.cert.fr/dcsd/biannic/idco/perso/Biannic/index.html>

Symbolic Toolbox© or the Maple© environment.

3. Derive them symbolically with respect to the state variables x and u around a chosen equilibrium point (x_0, u_0) . The system will be of the following form:

$$\begin{aligned} \delta\dot{x} &= \left. \frac{\partial f}{\partial x} \right|_{x=x_0, u=u_0} \delta x + \left. \frac{\partial f}{\partial u} \right|_{x=x_0, u=u_0} \delta u \\ \delta\dot{x} &= A \delta x + B \delta u \end{aligned} \quad (2)$$

Without any lack of generality, we assume $D = 0$:

$$y = Cx + 0u$$

The matrices A , B and C are function of the remaining symbolic variables, especially the equilibrium point and the uncertain parameters.

4. Improve conditioning by normalizing all remaining symbolic variables, *i.e.* all variables shall be of the same magnitude.
5. Generate the LFT properly speaking from a symbolic expression. This crucial step called *symbolic pre-processing* was already illustrated in [6].
6. Maybe apply numerical order reduction with for example `minlfr`
7. Normalize parameter variations such that $\delta_i \in [-1, 1]$ using the function `normalizelfr`

Here, the steps concerning more specifically the modelling of nonlinear systems will be illustrated.

3 The nonlinear 3-DOF aircraft model

The following well known flight mechanical equations describe the 3-DOF motion of an aircraft, see for example [3]:

$$\dot{\alpha} = q - \frac{1}{2m} \rho S V C_L - \frac{F_{mot}}{mV} \sin \alpha + \frac{g}{V} \cos \gamma$$

$$\dot{q} = \frac{M_{tot}}{I_{yy}} \quad (3)$$

$$\dot{\Theta} = q$$

with g the gravity, ρ the air density and α the angle of attack, q the pitch rate, Θ the pitch angle, $\gamma = \Theta - \alpha$ the flight path angle, m the mass of the aircraft, S its reference surface, I_{yy} its inertia around the y -axis, V the airspeed, C_L the lift coefficient, F_{mot} the thrust and M_{tot} the pitching moment w.r.t. the centre of gravity x_{cg} where:

$$\begin{aligned} M_{tot} &= M_{aero} + (z_{mot} - z_{ref}) F_{mot} \\ &\quad - (x_{ref} - x_{cg}) F_{aero,z} \end{aligned}$$

x_{ref} and z_{ref} define the reference point of the aircraft in the body axis and z_{mot} the vertical distance of the engines w.r.t. this reference point. The aerodynamic moment is given by

$$M_{aero} = \frac{1}{2} \rho S l V^2 C_M$$

with l the aerodynamic mean chord and C_M the moment coefficient. $F_{aero,z}$ is the resulting force in the z -axis:

$$F_{aero,z} = \frac{1}{2} \rho S V^2 (-C_L \cos \alpha - C_D \sin \alpha)$$

with C_D the drag coefficient.

C_M , C_L are given as nonlinear look up tables in function of Ma and C_D is given as a nonlinear look up table in function of Ma and α . See section 4.1 for more details how these tables are replaced by polynomial expressions. The Mach number Ma is defined as:

$$Ma = \frac{V}{c} \quad (4)$$

with c the sound speed. c is given by:

$$c(T(h)) = \sqrt{1.4 \cdot 287.053 \cdot T(h)} \quad (5)$$

as a function of the atmosphere temperature T which is itself a function of the flight altitude h according to the International Standard Atmosphere ISA. The air density ρ is also depending on h via an exponential function:

$$\rho(h) = e^{-x(h)} \quad (6)$$

See more details on the replacement of irrational functions in section 4.2.

4 Treatment of nonlinearities

4.1 Replacement of nonlinear tabulations

Nonlinear tabulations are replaced by polynomial approximations of the functional relation

$$y = f(x_1, x_2, \dots, x_n) \quad (7)$$

between the variables x_i and the output y .

4.1.1 Classical Least Mean Square Error approach

The most common approach to find the functional relation of Eq. (7) is the Least Mean Square Error LMSE approach, see for example [11]. The user has to choose the polynomial structure of an estimated interpolation function:

$$\hat{f} = \sum_0^M a_m x_1^{k_1} \dots x_n^{k_n} \quad (8)$$

where a_m are the polynomial coefficients. The user has to choose the maximum order k_i for each variable x_i and to select the M monomials $x_1^{k_1} \dots x_n^{k_n}$. A first shot solution would consist in choosing the full combinatoric of $x_1^{k_1} \dots x_n^{k_n}$ with a high number M . The user then calculates the unknown polynomial coefficients a_m by minimizing the mean square error between y and \hat{y} using the following cost function:

$$J_{LMSE} = (\mathbf{y} - P \mathbf{a})^T (\mathbf{y} - P \mathbf{a}) \quad (9)$$

with the well known result:

$$\mathbf{a} = (P^T P)^{-1} P^T \hat{\mathbf{y}} \quad (10)$$

This approach is already available within the LFR Toolbox via the `data2lfr` or `data2sym` functions.

4.1.2 Predicted Mean Square Error approach

This method proposed by Morelli [11] uses a mixed criterion

$$J_{PMSE} = J_{LMSE} + \sigma M \quad (11)$$

penalising not only the mean square error but at the same time a term proportional to the total number of terms M included in the model where σ is a weighting parameter. This is a convex optimization problem. It is called predicted mean square error *PMSE*. The idea is hence to determine not only the polynomial coefficients \mathbf{a} but also automatically the polynomial structure of \hat{y} with minimal number of monomials. This can be achieved by applying orthogonal functions [11]. With this property, Eq. (10) can be decoupled and finally evaluated row wise as:

$$\mathbf{a}_j = \frac{p_j^T \hat{\mathbf{y}}}{p_j^T p_j} \quad (12)$$

The cost function J_{LMSE} of Eq. (9) can therefore be simplified to:

$$J_{LMSE_{orth}} = \mathbf{y}^T \mathbf{y} - \sum_{j=1}^M \mathbf{a}_j^2 p_j^T p_j \quad (13)$$

With the form of Eq. (13) the individual contribution of each monomial to the cost function can be quantified. The monomials can also be ranked in descending order in terms of their contribution to reduce the error. If their contribution is under a threshold ε , they are not used within the polynomial \hat{f} . The user has just to choose the maximum order k_i of the polynomial expression as well as the weighting parameter σ and the threshold ε .

4.1.3 PMSE-LFT based approach

Based on the method described in section 4.1.2 a new cost function will be proposed in the following which penalises not only the total number of terms, but additionally the sum of encountered exponents. In the framework of LFT modeling, this has the advantage that *e.g.* five monomials of order one contribute less to the LFT size than one single term of order six. It is thus worth penalising the contribution of this one term of higher order more severely than the sum of the number of the lower order terms. This fact is expressed

in the resulting cost function $J_{PMSE-LFT}$:

$$J_{PMSE-LFT} = J_{PMSE} + c_{exp} \sum_{l=1}^M \left\{ \sum_{j=1}^n \exp(x_j) \right\} \quad (14)$$

where $\exp(x_j)$ is the exponent of the variable x_j . The constant c_{exp} has to be adapted according to the desired ratio of weighting the different contributions to the cost function. Finally, the cost function $J_{PMSE-LFT}$ can be modified to a cost function not considering the total number of terms at all, but purely punishing the sum of exponents encountered in the final expression. This leads to the cost function J_{LFT} :

$$J_{LFT} = J_{LMSE} + c_{exp} \sum_{l=1}^M \left\{ \sum_{j=1}^n \exp(x_j) \right\} \quad (15)$$

The function `strident` applying the PMSE, PMSE-LFT and the pure LFT approach will be added to the next release of the LFR-Toolbox.

4.1.4 Example

For the drag coefficient $C_D(Ma, \alpha)$ of the considered aircraft the polynomial expression is determined using the 4 preceding methods. Tab. 1 gives an overview over the obtained results. It can be seen that the complexity and the total number of terms considerably decrease from a LMSE approach via PMSE and PMSE-LFT to a purely LFT size optimized approach. At the same time, the accuracy decreases only slightly. A trade off has to be made between an acceptable loss in accuracy on the one hand side and an acceptable LFT size on the other hand.

Method	Number of terms	Sum exponents	Maximum error
LMSE (all terms)	33	156	5.4026e-003
PMSE	32	154	5.5578e-003
PMSE-LFT	27	120	6.1668e-003
LFT	25	108	6.2141e-003

Table 1 Comparison of the different interpolation results

4.2 Substitution of irrational functions

All irrational functions have also to be replaced by rational approximations.

The square root function in Eq. (5) for the sound speed is replaced by the following 6th order polynomial expression:

$$c = k_{c,0} + k_{c,1}h + k_{c,4}h^4 + k_{c,5}h^5 + k_{c,6}h^6 \quad (16)$$

found by the function `strident`. The exponential function in Eq. (6) for the air density is replaced by a 2nd order polynomial expression:

$$\rho = k_{\rho,0} + k_{\rho,1}h + k_{\rho,2}h^2 \quad (17)$$

4.3 Substitution of trigonometrical functions

Concerning the trigonometrical functions, the following Taylor series can be applied:

$$\begin{aligned} \sin \alpha &= \alpha - \frac{\alpha^3}{3} + \frac{\alpha^5}{5} \\ \cos \alpha &= 1 - \frac{\alpha^2}{2} + \frac{\alpha^4}{4} - \frac{\alpha^6}{6} \end{aligned} \quad (18)$$

in order to limit the errors to 10⁻³% at $\alpha = 40^\circ$. This means that in the worst case, α will appear 9 times for a sinus term and 12 times for a cosine term. For small α the following approximations could be used :

$$\begin{aligned} \sin \alpha &= \alpha \\ \cos \alpha &= 1 \end{aligned} \quad (19)$$

The special LFT blocks for use with Simulink `sineLFR`,...use the approach proposed by Cumer *et al*[4] for trigonometrical dependencies.

5 The symbolic linearization and the construction of the LFT

In order to obtain the linearized system around a trim condition, Eq. (3) are symbolically derived with respect to the two states α and q and the elevator deflection δm using the following Matlab code:

```
A11 = diff(alphadot,'alpha');
A12 = diff(alphadot,'q');
A13 = diff(alphadot,'Theta');
A21 = diff(qdot,'alpha');
A22 = diff(qdot,'q');
A23 = diff(qdot,'Theta');
A31 = diff(thetadot,'alpha');
A32 = diff(thetadot,'q');
A33 = diff(thetadot,'Theta');
B1 = diff(alphadot,'dm');
B2 = diff(qdot,'dm');
B3 = diff(thetadot,'dm');
```

Name	Dims
V	31x31
alpha	84x84
centxcg	3x3
dm	19x19
fmotd	1x1
fmotg	1x1
iyy	3x3
mach	325x325
masse	6x6
q	19x19
zp	18x18

The 3-DOF system is then symbolically built with the following code:

```
Asys = [A11 A12 A13;
        A21 A22 A23;
        A31 A32 A33];
Bsys = [B1;
        B2;
        B3];
Csys = [eye(3,3);
        -1 0 1];
Dsys = zeros(4,1);
sys_lon = [Asys Bsys;
           Csys Dsys];
```

The LFT is finally obtained with the following functions:

```
ABCD_lon = sym2lfr(sys_lon);
lft_lon = abcd2lfr(ABCD_lon,3);
```

where `sym2lfr` transforms the symbolic system into an `lfr` object. `abcd2lfr` puts this object into the typical transfer function $P - \Delta$ form of Fig. 1 knowing that there are 3 states. Using the LMSE expression of Table 1 for C_D , the LFT size is given by:

```
>> size(lft_lon)
```

```
Uncertainty blocks
globally(510 x 510):
```

When using the LFT expression of Tab. 1, the LFT size can be considerably reduced to:

```
Uncertainty blocks
globally(137 x 137):
```

Name	Dims
V	9x9
alpha	33x33
centxcg	3x3
dm	1x1
fmotd	1x1
fmotg	1x1
iyy	3x3
mach	61x61
masse	6x6
q	1x1
zp	18x18

without almost no loss in precision. Ongoing studies will illustrate how this LFT size can still be reduced by physical means or a more sophisticated symbolical pre-processing.

6 Validation of the LFT

6.1 Linear validations

Multiple flight cases are studied within the flight domain of Fig. 2 depending on calibrated airspeed V_c and altitude on the borders of the flight domain. Additionally, flight cases within the

MODELLING OF THE 3-DOF NONLINEAR AIRCRAFT BEHAVIOUR VIA SMALL STANDARD FORMS

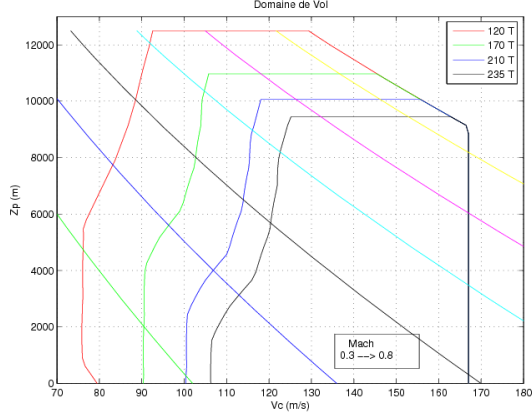


Fig. 2 The flight domain of the considered aircraft

flight domain with $Ma = 0.3$ and $Ma = 0.8$ have been considered.

In the worst case, the Δ matrix is defined as follows:

```

zp      = 23000;
V       = 215.4;
Vcas   = 300;
masse  = 155000;
centxcg = 0.40;
alpha  = 1.4*degrad;
dm     = -0.94*degrad;
fmotd  = 1.73*degrad;
fmotg  = 1.73*degrad;
mach   = 0.69;
theta  = 1.4*degrad;
    
```

The matrices A and B of the numerically linearized nonlinear system Eq. (2) are:

$$A(\Delta_0) = \begin{bmatrix} -1.1106 & 0.9814 & 0 \\ -0.8495 & -0.6440 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$B(\Delta_0) = \begin{bmatrix} -0.1063 \\ -4.3250 \\ 0 \end{bmatrix}$$

while the following Matlab sequence:

```
Delta_nom = [V, alpha, theta,
```

```

centxcg, dm, fmotd,
fmotg, iyy, mach,
masse, q, zp];
name = {'V', 'alpha', 'theta',
'centxcg', 'dm', 'fmotd',
'fmotg', 'iyy', 'mach',
'masse', 'q', 'zp'};
sys_lft = uplft(lft_lon, name,
Delta_nom)
    
```

closes the LFT via Δ . `sys_lft.a` and `sys_lft.b` are the resulting state space matrices:

$$A'(\Delta_0) = \begin{bmatrix} -1.1013 & 0.9815 & 0 \\ -0.8454 & -0.6400 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$B'(\Delta_0) = \begin{bmatrix} -0.1056 \\ -4.2997 \\ 0 \end{bmatrix}$$

The error is less than 2% in the coefficients of A and B . For the eigenvalues holds:

$$\lambda_A = -0.8773 \pm 0.8828i$$

$$\lambda_{A'} = -0.8706 \pm 0.8813i$$

with an error of 2%. In all other cases, the errors are smaller. In order to confirm these results, the Bode diagrams for all cases are shown in Fig. 3. The frequency plots of 7 cases based on the LFT systems (black lines) and the corresponding linearized systems are almost superposed (red lines).

6.2 Nonlinear validation

In order to simulate the family of LTI systems given by the LFT, the Sim LFR LTI/LTV block for Simulink developed by [2] is used. See Fig. 4 for the interconnection structure. Just α and q are varying significantly during the simulation period due to a deflection $\Delta\delta m$ around trim condition. The α entering the Δ matrix is hence the sum of the initial trim value α_0 and $\Delta\alpha$ stemming from the integration of the LFT. The trimming point evolves with the time. The linearized

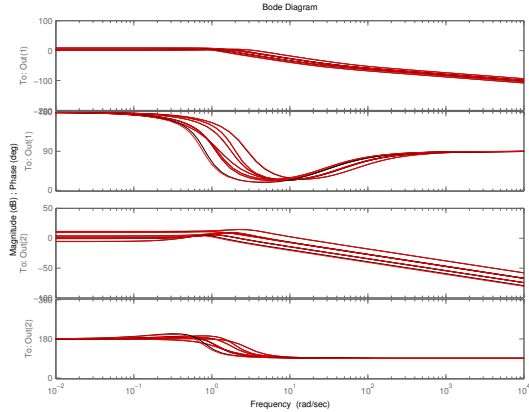


Fig. 3 The comparison of the Bode plots using the at Δ_0 fixed LFT (black) and those ones using the linearized nonlinear time varying system (red)

system is continuously adapted to a new flight condition. For q holds the same.

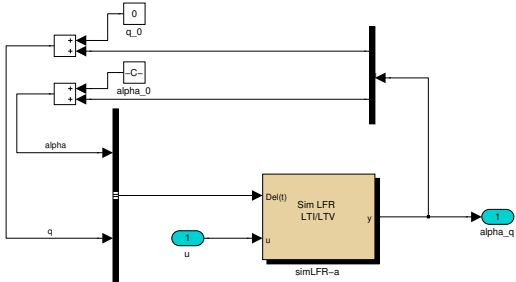


Fig. 4 The block-diagram for the simulation of the LFT

The simulation of the LFT system is compared to the simulation of the nonlinear system. The time responses of 7 cases are given in Fig. 5. They superpose perfectly. The LFT framework is hence an adequate framework to simulate or to analyze a nonlinear system as an LFT represents the continuum of its linearized systems.

7 Conclusions

First, the proposed overall modelling procedure for nonlinear systems under LFT form was intro-

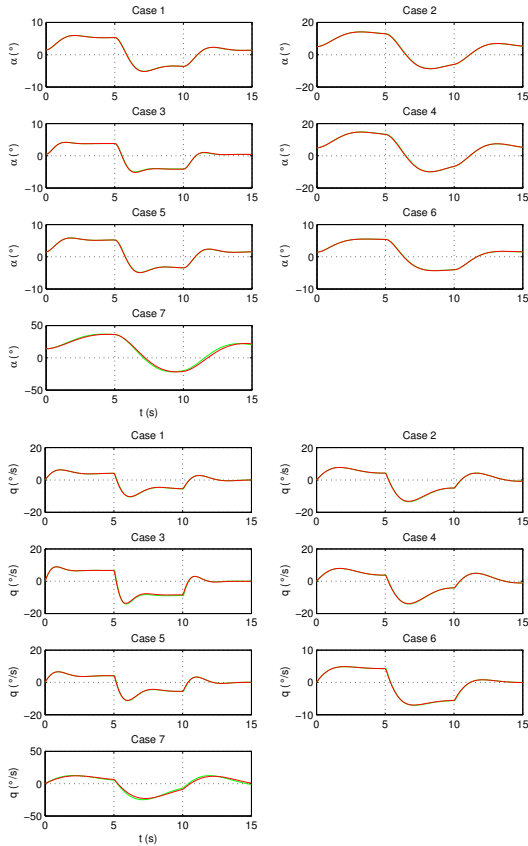


Fig. 5 The comparison of the time plots using the at Δ_0 fixed LFT (green) and those ones using the complete nonlinear time varying system (red)

duced. The replacement of nonlinear tabulations as well as the substitution of exponential and trigonometrical functions by rational expressions were detailed. In this context, the advantage of multivariate orthogonal functions was illustrated. Symbolic linearization and pre-processing have then be highlighted before having built the LFT for the nonlinear aircraft with time varying parameters. Finally, the LFT was validated by means of pole-charts, by frequency plots and nonlinear time plots.

8 In memory of Jean-François Magni

This article pays homage to Jean-François Magni who died suddenly on January 4th, 2008 at the age of 53 years. His research interests covered the pole and eigenstructure assignment techniques where he was the first to study necessary and sufficient conditions and the tractability for the challenging problem of pole assignment by static and fixed-order output feedback. More recently, his research has focused on multi-objective and gain scheduling controller design for linear parameter varying systems using modal techniques. He always desired to offer practical tools to design engineers. The LFR toolbox used in this article was his latest achievement. We endeavour to update and to extend it in his memory.

References

- [1] C.M. Belcastro, T.H. Khong, J.Y. Shin, H. Kwatny, B.C. Chang, and G.J. Balas. Uncertainty modeling for robustness analysis of aircraft control upset prevention and recovery systems. In *Proc. AIAA Conf. on Guidance, Navigation and Control*, San Francisco, California, USA, August 2005. paper no.6427.
- [2] J.M. Biannic, C. Döll, and J.F. Magni. Simulink-based tools for creating and simulating interconnected. In *IEEE Symposium on Computer-Aided Control System Design*, Munich, Germany, October 2006.
- [3] J.L. Boiffier. *The dynamics of flight : The equations*. John Wiley & Sons, Hoboken, Chichester, Weinheim, Singapore, Mitton, 1st edition, August 1998.
- [4] C. Cumer and J.P. Chretien. Minimal LFT form of a spacecraft built up from two bodies. In *Proc. AIAA Conf. on Guidance, Navigation and Control*, Montreal, Quebec, Canada, August 2001.
- [5] R.L. Dailey, J.C. Doyle, G. Stein, S.S. Banda, and H.H. Yeh. Lecture notes for the workshop on \mathcal{H}_∞ and μ methods for robust control. In *Proc. American Control Conference*, San Diego, California, USA, May 1990.
- [6] C. Döll and A. Knauf. Comparison in terms of size of several LFT models for the lateral motion of a generic fighter aircraft. In *Proc. AIAA Conf. on Guidance, Navigation and Control*, San Francisco, California, USA, August 2005. paper no. 6252.
- [7] G. Ferreres and J.M. Biannic. Reliable computation of the robustness margin for flexible transport aircraft. In *Proc. AIAA Conf. on Guidance, Navigation and Control*, volume Paper no. AIAA 2000-4365, Denver, Colorado, USA, June 2000.
- [8] A. Knauf, C. Döll, and J.F. Magni. A LFT model for the longitudinal motion of a generic fighter aircraft. In *Proc. 1st European Conference of Aerospace Sciences EUCASS*, Moscow, Russia, July 2005.
- [9] A. Knauf, S. Gaulocher, and A.P. Feuersänger. Self-scheduled controller design for relative-motion control of a satellite formation. In *IMAACA07/IMSM07*, Buenos Aires, Argentina, February 2007. IEEE.
- [10] J.F. Magni. Presentation of the Linear Fractional Representation Toolbox (LFRT). In *Proc. IEEE Symposium on Computer-Aided Control System Design*, Glasgow, Scotland, UK, September 2002.
- [11] E.A. Morelli and R. De Loach. Wind tunnel database development using modern experiment design and multivariate orthogonal functions. In *Proc. 41st AIAA Aerospace Sciences Meeting and Exhibit*, pages 2003-0653, Reno, Nevada, USA, January 2003.
- [12] C. Roos and J.M. Biannic. Aircraft-on-ground lateral control by an adaptive LFT-based anti-windup approach. In *Proc. IEEE Symposium on*

Computer–Aided Control System Design, Munich, Germany, October 2006.

- [13] A. Varga and G. Looye. Symbolic and numerical software tools for LFT–based low order uncertainty modeling. In *Proc. IEEE International Symposium on Computed Aided Control System Design*, pages 1–6, Kohala Coast, Hawaii Island, Hawaii, USA, August 1999.

9 Copyright Statement

The authors confirm that they, and/or their company or institution, hold copyright on all of the original material included in their paper. They also confirm they have obtained permission, from the copyright holder of any third party material included in their paper, to publish it as part of their paper. The authors grant full permission for the publication and distribution of their paper as part of the ICAS2008 proceedings or as individual off-prints from the proceedings.