From Generic Aircraft Models towards LFTs Based Parametric Uncertainties Descriptions: A Computer Aided Methodology

ANDECS Working Note: AWN-02

November, 1994

A. Varga

DLR - Oberpfaffenhofen
German Aerospace Research Establishment
Institute for Robotics and System Dynamics
P.O.B. 1116, D-82230 Wessling, Germany
E-mail: Andras.Varga@dlr.de

January 24, 1995

1 Introduction

The structured singular value (called also μ) was introduced to study linear models with structured parametric uncertainties [1] and later extended to cope with parametric uncertainties described by *linear fractional transformations* (LFTs) [2, 3]. The evolving μ-analysis and synthesis methodology represents a powerful tool for robust control design of systems with structured real parametric uncertainties described by LFTs (see for instance [4] for a complex synthesis example). In order to use such a methodoly, LFTs based uncertainty model are necessary to be developed.

In this working note we discuss the general aspects of the automatic generation of LFTs based parametric uncertainty models from a generic nonlinear aircraft-dynamics model. As it will be apparent, a computer assisted modelling methodology for the generation of linear models with LFTs based parametric uncertainty descriptions integrates many specialized software tools as for instance tools for object-oriented modeling to generate particular aircraft models, tools for simulation and numerical computations to determine equilibrium points and to reduce the order of LFT systems, and symbolic computations tools to perform linearization and to generate LFTs based uncertainty descriptions.

2 Development of Generic Models

The dynamical behaviour of many lumped-parameter processes, and in particular of a flying aircraft, can be described by non-linear dynamic system models of the form

$$\begin{aligned} E(x(t), p)\dot{x}(t) &= F(x(t), u(t), p) \\ y(t) &= G(x(t), u(t), p) \end{aligned} \tag{1}$$

where x, u, y are the state-, input- and output–vectors respectively, and p is the vector of model parameters. The matrix $E(x(t), p)$ is structurally non-singular and thus can be inverted if necessary. A generic aircraft model can be seen as the interconnection of several dynamical and static subsystems describing different parts of the aircraft dynamics and of the interactions of aircraft with its flying environment. Generic models are useful for obtaining dynamical models of particular aircrafts by appropriately choosing the various component subsystems and the corresponding parameter sets.

A generic model based approach for aircraft-dynamics modelling has been recently proposed in [5]. This approach is based on the object-oriented modelling environment Dymola [6, 7] and is a versatile methodology to generate particular aircraft models for simulation purposes. Dymola provides facilities for the generation of simulation codes for several simulation environments. For instance, the Fortran simulation code generated for the DSSIM simulation module of ANDECS can be used not only for simulation purposes, but also to compute equilibrium points for the nonlinear model (1). The MATLAB code generated for simulation with SIMULINK can be easily simplified and translated into a MAPLE-readable format to obtain symbolic description of the non-linear aircraft model.

3 Computation of Equilibrium Points

A prerequisite for linearization is the availability of equilibrium points corresponding to specific flight conditions. The generic nonlinear model implemented in Dymola can be used to automatically generate the Fortran subprograms to be used by DSSTAT module of ANDECS to compute equilibrium points for nonlinear systems. The computation of equilibrium points can be done either by specifying the constant inputs $u(t) = \bar{u}$ and computing $\bar{x}$ and $\bar{y}$, the corresponding equilibrium values of the state and output vectors respectively, or by specifying a desired stationary output $\bar{y}$ and looking for $\bar{x}$ and $\bar{u}$. In both cases the resulting equilibrium points set $\{\bar{x}, \bar{u}, \bar{y}\}$ satisfies, for the nominal values of parameters, the system of nonlinear of equations

$$\begin{aligned} 0 &= F(\bar{x}, \bar{u}, p) \\ \bar{y} &= G(\bar{x}, \bar{u}, p) \end{aligned} \tag{2}$$

Depending on user's option, it is possible to enlarge the parameter set p with the computed equilibrium points set values in order to account for the uncertainties produced by neglecting higher order terms during liniarizations.

4 Development of Generic Symbolic Models

With the computed equilibrium points it is possible to generate from a generic aircraft model implemented in Dymola a generic symbolic model suitable for linearization. The approach which we propose relies on the MATLAB code which can be generated for SIMULINK with the help of Dymola [7]. The resulting *.m* file is an ordinary text file devised to evaluate "the right hand sides "

$$\begin{array}{rcl} \bar{F}(\delta x,\, \delta u,\, p) & := & [E(\bar{x} + \delta x,\, p)]^{-1} F(x,\, \bar{u} + \delta u,\, p) \\ \bar{G}(\delta x,\, \delta u,\, p) & := & G(\bar{x} + \delta x,\, \bar{u} + \delta u,\, p) \end{array} \tag{3}$$

corresponding to the ordinary differential equations arising from (1). This text file can be easily modified to conform with the syntax of MAPLE, a powerful symbolic computations tool. Because of the well structured layout of the MATLAB interface, most of necessary modifications can be done automatically by using standard stream editing tools (for instance the UNIX programs `sed` or `awk`). The only difficulty is caused by the presence of *if-then-else* constructs in the generated MATLAB code. These constructs can be eliminated by implementing a specialized conversion program which can use the MATLAB file itself to check the various *if*-conditions in the computed equilibrium points and to select from the alternative expression that one which corresponds to the true or false value of the *if*-condition. This approach relies on the **eval** function provided in MATLAB to interpret and evaluate strings containing MATLAB expressions. In this way, the conversion program produces explicit symbolic expressions, readable from MAPLE, for $\bar{F}(\delta x,\, \delta u,\, p)$ and $\bar{G}(\delta x,\, \delta u,\, p)$ to serve further for symbolic evaluation of the matrices of the linearized model.

5 Symbolic Linearization

By linearization of the non-linear model (1) in the neighborhood of an equilibrium points set $\{\bar{x}, \bar{u}, \bar{y}\}$ we obtain a linear-time invariant model of the form

$$\begin{array}{rcl} \delta\dot{x} & = & A(p)\delta x + B(p)\delta u \\ \delta y & = & C(p)\delta x + D(p)\delta u \end{array} \tag{4}$$

where the system matrices can be computed symbolically from the expressions of $\bar{F}(\delta x,\, \delta u,\, p)$ and $\bar{G}(\delta x,\, \delta u,\, p)$ in (3) as follows

$$\begin{array}{ll} A(p) = \left.\dfrac{\partial \bar{F}(\delta x, \delta u, p)}{\partial(\delta x)}\right|_{\substack{\delta x=0\\ \delta u=0}}, & B(p) = \left.\dfrac{\partial \bar{F}(\delta x, \delta u, p)}{\partial(\delta u)}\right|_{\substack{\delta x=0\\ \delta u=0}}, \\ C(p) = \left.\dfrac{\partial \bar{G}(\delta x, \delta u, p)}{\partial(\delta x)}\right|_{\substack{\delta x=0\\ \delta u=0}}, & D(p) = \left.\dfrac{\partial \bar{G}(\delta x, \delta u, p)}{\partial(\delta u)}\right|_{\substack{\delta x=0\\ \delta u=0}}. \end{array} \tag{5}$$

The computation of these matrices can be easily performed by using the differentiation and partial evaluation functions provided in MAPLE.

6 Generation of LFTs Descriptions

The elements of the matrices of the linearized system (4) depend only on the parameters p_i, $i = 1, \ldots, q$. (As mentioned before, the vector p optionally can include the components of the equilibrium state-, input- and output-vectors $\bar{x}$, $\bar{u}$, and $\bar{y}$.) Any uncertainty in a parameter p_i expressed as $p_i \in [\underline{p_i},\ \bar{p}_i]$ can transcribed in a normalized form $p_i = p_{i0} + s_{i0}\delta p_i$ with $|\delta p_i| \leq 1$, $p_{i0} = (\underline{p_i} + \bar{p}_i)/2$ and $s_{i0} = (\bar{p}_i - \underline{p_i})/2$. This local parameter uncertainty can be also expressed as an elementary upper LFT

$$p_i = \mathcal{F}_u\left(\begin{bmatrix} 0 & s_{i0} \\ 1 & p_{i0} \end{bmatrix}, \delta p_i\right).$$

Recall that for a partitioned matrix M

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \in \mathbb{R}^{(p_1+p_2)\times(q_1+q_2)}$$

and for $\Delta \in \mathbb{R}^{q_1\times p_1}$, the *upper* LFT $\mathcal{F}_u(M, \Delta)$ is defined as

$$\mathcal{F}_u(M, \Delta) = M_{22} + M_{21}(I - \Delta M_{11})^{-1}\Delta M_{12}.$$

If all elements of matrices A, B, C and D are rational functions in parameters p_i, $i = 1, \ldots, q$, then the parametric uncertainties at the components level can be transformed to structured uncertainties at the level of system matrices by using the properties of LFTs [2]. Thus for the system matrices, LFT uncertainty models can be generated in the forms

$$A(p) = \mathcal{F}_u\left(\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_0 \end{bmatrix}, \Delta_A\right), \quad B(p) = \mathcal{F}_u\left(\begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_0 \end{bmatrix}, \Delta_B\right),$$

$$C(p) = \mathcal{F}_u\left(\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_0 \end{bmatrix}, \Delta_C\right), \quad D(p) = \mathcal{F}_u\left(\begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_0 \end{bmatrix}, \Delta_D\right),$$

where Δ_A, Δ_B, Δ_C and Δ_D are diagonal matrices having on the diagonal the normalized uncertainty parameters $\delta p_1, \delta p_2, \ldots, \delta p_q$. Notice that A_0, B_0, C_0 and D_0 are the *nominal* values of the respective matrices (for all δp_i set to zero).

Procedures to generate LFTs based uncertainty descriptions have been proposed in [8] and [9]. For the approach proposed in [8] MATLAB implementations are also available. These implementations rely on the assumption that all elements of system matrices are rational functions in parameters.

For our purposes, an alternative approach based on using the symbolic package MAPLE seems to be more attractive for the following reasons:

- The matrices of the linearized model are already generated in MAPLE format from the previous step.

- Non-rational matrix elements can be easily handled within MAPLE, by replacing any non-rational expression with an appropriate order Taylor series based polynomial approximation.
- An optimization feature present in MAPLE allows to minimize the number of multiplications in evaluating polynomials in several variables. This leads implicitly to the minimization of the ordera of all LFT description constructed for individual elements of the system matrices. It is in our opinion very likely that the global LFT descriptions obtained in this way will have smaller order than those resulted with the procedure proposed in [8]. Moreover, the intrinsic built-in minimization feature is also beneficial in case of using the alternative procedure proposed in [9].

Preliminary MAPLE programs are already available for several basic operations involving LFTs.

7 Further Processing

In order to obtain LFTs based parametric descriptions suitable for use in μ-analysis and synthesis several further processing are usually necessary. A reordering of diagonal elements of the uncertainty matrices Δ_A, Δ_B, Δ_C and Δ_D is frequently necessary, in order to obtain repeated blocks. Such grouping is also necessary if we intend to further reduce the order of resulted LFTs descriptions. This processing step can be performed simbolically with MAPLE.

Further computations are essentially numerical and thus it is necessary to import the resulting LFTs description of system matrices from MAPLE into ANDECS (or MATLAB). This can be done by using a dedicated program to perform the conversion of MAPLE matrix data format into the ANDECS data-base format. Such a conversion is presently problematic because of the lack in the current version of ANDECS of appropriate *control data objects* (CDOs) to handle LFT based parametric uncertainty models. It is intended however to include such CDOs in the new version (Version 2.0) of ANDECS [10].

The resulting LFTs based parametric descriptions are generally non-minimal. The construction minimal order descriptions is essentially a multidimensional minimal realization problem, which even for the 2-D case is a very difficult problem to solve. Although the intrinsic minimization built in the procedures to generate the LFTs models helps to keep the resulting orders lower, these orders are usually still to large because of non-minimality of the overal LFT descriptions. An ad-hoc procedure suggested in [8] can be used for reducing the order of individual repeated blocks. The procedure essentially solves 1-D minimal realizations problems for each repeated block. Although there is no guaranty for minimality, this procedure is apparently effective on many practical examples. A more involved approach based on model reduction techniques for LFT systems can be also used [11]. The use of the latter procedure involves the solution of Lyapunov-type *linear matrix inequalities* (LMIs). Efficient computational procedures for this purpose are still under development. [12]

8 Conclusions

For the automatic generation of LFTs based parametric uncertaiuty descriptions from a generic aircraft-dynamics model, the following problems are to be solved:

1. Elaboration of a generic aircraft-dynamics model which explicitly accounts for parametric uncertainties
2. Implementation of a conversion module (in MATLAB, Fortran or C) to translate a standard MATLAB-SIMULINK simulation *.m* file into a MAPLE readable program
3. Implementation of MAPLE procedures for the following symbolic manipulations:
   - linearization of nonlinear systems
   - elimination of non-rational parametric expressions
   - generation of multidimensional LFT system description for a rational expression
   - generation of multidimensional LFT system description for the system matrices
   - postprocessing of resulting LFTs to form repeated blocks
   - outputing the resulting LFTs in a format appropriate for conversion to ANDECS CDOs format
4. Implementation of ANDECS programs for handling CDOs with LFTs
5. Implementation of a conversion program from MAPLE to ANDECS for LFTs
6. Implementation of numerical procedures for order reduction of LFTs systems

References

[1] J. C. Doyle. Analysis of feedback systems with structured uncertainties. *IEE Proceedings*, 129, Part D:242–250, 1982.

[2] J. C. Doyle, A. Packard, and K. Zhou. Review of LFTs, LMIs, and μ. In *Proc. of 30th CDC, Brighton, England*, pages 1227–1232, 1991.

[3] P. M. Young, M. P. Newlin, and J. C. Doyle. μ analysis with real parametric uncertainty. In *Proc. 30th CDC, Brighton, England*, pages 1251–1256, 1991.

[4] J.C. Doyle, K. Lenz, and A. Packard. Design examples using μ-synthesis: space shuttle lateral axis FCS during reentry. In R. F. Curtain, editor, *Modelling, Robustness, and Sensitivity Reduction in Control Systems*, volume F-34 of *NATO ASI Series*, pages 127–163. Springer Verlag Berlin, 1987.

[5] D. Moormann, H. D. Joos, and G. Grübel. First steps in developing an automatically computer-processable generic aircraft-dynamics model. Technical Report, TR R159-94, DLR-Oberpfaffenhofen, Institute of Robotics and System Dynamics, May 1994.

[6] H. Elmquist. Object-oriented modeling and automatic formula manipulation in Dymola. Scandinavian Simulation Society SIMS'93, Konsberg, Norway, June 1993.

[7] H. Elmquist. *Dymola – Dynamic Modeling Language. User's Manual.* Dynasim AB, 1994.

[8] J. Terlouw, P. Lambrechts, S. Bennani, and M. Steinbuch. Parametric uncertainty modeling using LFTs. In *Proc. of AIAA GNC Conf., Hilton, South Carolina*, 1992.

[9] Y. Cheng and B. De Moor. A multidimensional realization algorithm for parametric uncertainty modeling problems. In *Proc. 32nd CDC, San Antonio, Texas*, pages 3022–3023, 1993.

[10] A. Varga. Proposals for Control Data Objects in ANDECS 2.0. ANDECS Working Notes AWN-01, May 1994.

[11] W. Wang, J. Doyle, and C. Beck. Model reduction of LFT systems. In *Proc. 30th CDC, Brighton, England*, pages 1233–1238, 1991.

[12] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory.* SIAM, 1993.