Proceedings of the EuroGNC 2013, 2nd CEAS Specialist Conference on
Guidance, Navigation & Control, Delft University of Technology, Delft,
The Netherlands, April 10-12, 2013

LFT model generation via ℓ_1-regularized least squares

Harald Pfifer and Simon Hecker

Abstract The paper presents a general approach to approximate a nonlinear system by a linear fractional representation (LFR), which is suitable for LFT-based robust stability analysis and control design. In a first step, the nonlinear system will be transformed into a quasi linear parameter varying (LPV) system. In the second step, the nonlinear dependencies in the quasi-LPV, which are not rational in the parameters, are approximated using polynomial fitting based on ℓ_1-regularized least squares. Using this approach an almost Pareto front between the accuracy and complexity of the resulting LFR can be efficiently obtained. The effectiveness of the proposed method is demonstrated by applying it to a nonlinear missile model of industrial complexity.

1 Introduction

Linear fractional transformations (LFTs) can be considered a standard form for many modern robust control methods. In literature, a plethora of algorithms based on LFTs exist for analysis or synthesis, see e.g. [1]. In general, control problems are dealing with nonlinear systems of the form

$$\begin{aligned} \dot{x} &= f(x,p,u) \\ y &= g(x,p,u), \end{aligned} \tag{1}$$

Harald Pfifer
Institute of System Dynamics and Control, German Aerospace Center - DLR, Muenchner Str. 20, 82234 Wessling, Germany, e-mail: harald.pfifer@dlr.de

Simon Hecker
University of Applied Sciences Munich, Lothstr. 64, 80335 Muenchen, Germany e-mail: hecker@ee.hm.edu

where $x \in X \subset \mathbb{R}^{n_s}$ is the state vector, $y \in \mathbb{R}^{n_y}$ the output vector and $u \in \mathbb{R}^{n_u}$ the input vector. In addition the system can depend on a parameter vector $p \in \Pi$. In order to apply modern robust control methods on them, an efficient approach to approximate (1) by a linear fractional representation (LFR) needs to be available.

As an intermediate step on the way of obtaining an LFR of (1), the system is first transformed into an LPV system of the form

$$\begin{aligned} \dot{x} &= A(\delta)x + B(\delta)u \\ y &= C(\delta)x + D(\delta)u. \end{aligned} \tag{2}$$

In (2), δ can not only consist of the parameter vector p but also includes state-dependent nonlinearities, i.e. $\delta \in (X \times \Pi) \subset \mathbb{R}^{n_\delta}$, see [2]. In the latter case, the system is called quasi-LPV. Various techniques have been proposed in literature to perform the transformation of (1) into an quasi-LPV system (see for example [3, 4]).

If the quasi-LPV system (2) depends only rationally on δ, the transformation into a linear fractional representation (LFR) of the form

$$\begin{aligned} \dot{x} &= Ax + B_1 w + B_2 u \\ z &= C_1 x + D_{11} w + D_{12} u \\ y &= C_2 x + D_{21} w + D_{22} u \\ w &= \Delta(\delta) z \\ \Delta &= \mathrm{diag}(\delta_1 I_{s_1}, \ldots, \delta_{n_\delta} I_{s_{n_\delta}}) \end{aligned} \tag{3}$$

is straightforward.

Many sophisticated methods have been proposed in literature to obtain low order LFRs of a given LPV system, see [5] and the references therein. Usually, three steps are applied in the transformation process. First, a symbolic preprocessing of the LPV model is performed. Second, the actual transformation is conducted via object oriented LFT realization. Finally, numerical order reduction can be utilized to further reduce the order of the resulting LFR.

In several cases one may directly derive an analytic quasi-LPV (2) suitable for transforming into an LFR from a nonlinear system (1) via symbolic calculations. However, especially in aeronautical applications the models usually include highly nonlinear functions (neural networks, tables) or may only be given for a discrete set of conditions (linear aeroelastic models). In such cases the quasi-LPV model obtained via function substitution cannot be directly transformed into an LFR. The highly nonlinear functions or the discrete set of conditions have to be approximated by rational functions first.

It is largely an open question, how a function approximation is obtained, which is suitable for transformation of (2) into an LFR. Note that the minimal achievable LFR order depends mainly on the complexity (order of rational or polynomial approximations) and the structure of (2). In [6], we proposed a method to generate optimal LFT models achieving a good accuracy while keeping the complexity low. It involves directly minimizing a weighted sum of the LFR order and some error metric via a complex non-convex optimization. While this procedure provides good results as shown in [7] and [8], it is rather cumbersome and time consuming. The major contribution of this paper is a convex relaxation of the optimization of [6], which allows to efficiently obtain an (almost) Pareto front representing the compromise between the LFR order and accuracy.

In Section 2, the general problem formulation is stated. In Section 3, an overview over ℓ_1-regularized least squares and its application to polynomial fitting will be presented. Then the LFR generation problem is reformulated in the ℓ_1-regularized least squares framework in Section 4. Finally, in Section 5 the proposed algorithms are applied to a nonlinear missile model of industrial complexity with a nonlinear dynamic inversion based controller. It is shown that the approach allows to transform the highly nonlinear system into an LFR of sufficient accuracy, which still possesses a complexity suitable for performing LFT based stability analysis.

2 Problem Statement

The starting point of the LFT model generation is a quasi-LPV model (2), which can for instance be obtained via a function substitution technique as introduced by [3]. It is assumed that (2) does not depend rationally on the parameter vector δ. More precisely, there are $\{s_i(\delta)\}_1^p$ elements in (2) which need to be approximated by a rational/polynomial function, in order to transform (2) into an LFR. In a typical aerospace application the set $\{s_i(\delta)\}_1^p$ would contain for example the aerodynamic forces and moments coefficients. In the present work due to its simplicity only polynomial functions will be considered to approximate the original functions $s_i(\delta)$.

The first step is to generate a grid of values $s_{i,k}$ for each function s_i at a set of pre-specified parameter values. The value $s_{i,k}$ represents the i^{th} function evaluated at the k^{th} point and δ_k is the corresponding parameter vector. For each index value k an LTI system with transfer function $G_k = C_k(sI - A_k)^{-1}B_k + D_k$ can be built by evaluating the quasi-LPV model (2) at $\delta = \delta_k$.

The goal is now to calculate a polynomial approximation of the elements s_i, such that (2) can be transformed into an LFR of low complexity. It should, however, still represent the original nonlinear system (1) adequately. The problem can be conceptually described by

$$\min_{G_{lfr} \in \mathscr{S}_{lfr}} d(G_{lfr}, \{G_k\}_1^m) + w\, c(G_{lfr}), \tag{4}$$

where $d(.,.)$ is the notation of distance or model error between the approximate model G_{lfr} which is restricted to the class of LFT-based LPV models $\mathscr{S}_{lfr}$, see (3), and the grid point LPV model $\{G_k\}_1^m$. In addition, $c(.)$ describes the complexity of the resulting LFR and w is a weighting factor to balance complexity and accuracy.

As a measurement of the LFR's quality the ν-gap metric, as specified by [9], between the $\{G_k\}_1^m$ and the LFR evaluated at $\{\delta_k\}_1^m$ is applied. The ν-gap metric can take values between zero and one with zero meaning that two plants match closely and one that they are far apart. In general, any system norm can be used, e.g. $\mathscr{H}_2$-norm. The ν-gap metric has a decisive advantage over other system norms, though, as an error measurement for LPV model generation: It is also defined for unstable systems. Since in many practical cases the plant may be at least partially unstable in the admissible parameter set, special care has to be taken when choosing other system norms.

The complexity of the resulting LFR is estimated by the lower bound on its order as defined in [5], which is computationally faster than using the actual achievable order. For a given linear parametric model $S(\delta)$ with $\delta \in \mathbb{R}^{n_\delta}$ the lower bound can be calculated as follows: Substitute all but one parameter δ_i with random values and compute a minimal order, one parametric LFR with order m_i. Note, that for single parametric systems one can always calculate a minimal order LFR. Repeat this procedure for all parameters. Finally, the lower bound is given by $m_{LB} = \sum_{i=1}^{n_\delta} m_i$.

3 Polynomial Approximation by ℓ_1-Regularized Least Squares

The algorithm for finding polynomial approximations of the single matrix elements is based on a regularized least squares fitting. In the following $x_{j,k}$ denotes the numerical values of the j^{th} parameter at the k^{th} grid point, y is a vector including the m grid point values of an element s_i and b is a vector including the polynomial coefficients. In a first step, a matrix X will be built, which considers all possible bases for a multivariable polynomial of a given order evaluated at the m grid points.

In function approximation, it is often desired to choose from a set of potential bases one subset, which offers the best approximation of all subsets of the same cardinality, see [10]. The aim is to find a solution to the least squares problem with a sparse coefficient vector b, i.e with a small cardinality card(b), which is defined as the number of nonzero elements in the vector b. Considering a coefficient vector $b \in \mathbb{R}^n$ and $k < n$, it can be described as

$$\min_b \|Xb - y\|_2^2 \text{ , s.t. } \mathrm{card}(b) \leq k. \tag{5}$$

As shown in [10] this is a hard combinatorial problem. However, there exists a good heuristics to approximate (5), which is called ℓ_1-regularized least squares. In (6) $\|b\|_1 = \sum_{i=1}^{n} |b_i|$ is the ℓ_1-norm and λ the regularization parameter.

$$\min_b \|Xb - y\|_2^2 + \lambda \|b\|_1 \tag{6}$$

For a given λ this problem is actually convex and can be easily solved. By performing a sweep over λ a Pareto front is obtained, which presents the trade-off between card(b) and the residual $\|Xb - y\|_2$. Various techniques exist in literature to solve the convex problem (6), e.g. specialized interior-point methods as proposed in [11].

4 Procedure for the Generation of LFT Models

A weighted form of (6) can be used to simultaneously approximate all $\{s_i(\delta)\}_1^p$ by polynomials.

$$\min_b \sum_{i=1}^{p} w_{1,i}\|X\beta_i - y_i\|_2^2 + \lambda w_2^T|b|, \tag{7}$$

where b is a vector containing the polynomial coefficients over all elements that should be approximated. The vectors $\{\beta_i\}_1^p$ consist of those coefficients which are used in the approximation of the i^{th} element. The weightings w_1 and w_2 will be used later on to bring (7) closer to the original problem of minimizing the LFR order and an error metric between the LFR and the quasi-LPV.

Since all elements and especially all polynomial coefficients over all elements are considered in the cost function, scaling them is important. In the presented approach, the data is normalized so that each column of X and y has unit length and zero mean [12].

$$\begin{aligned} &\sum_{i=1}^{m} y_i = 0 \qquad \sum_{i=1}^{m} X_{i,j} = 0 \\ &\sum_{i=1}^{m} y_i^2 = 1 \qquad \sum_{i=1}^{m} X_{i,j}^2 = 1 \ \text{ for } \ j = 1,\dots,n \end{aligned} \tag{8}$$

Note that making use of orthogonal polynomials as e.g. Chebyshev polynomials and the standardization (8) results in X being orthonormal, i.e. $X^TX = I$. This simplifies solving the ℓ_1-regularized problem (7).

The regularization parameter λ in (7) is employed as a weighting between the accuracy and the complexity of the polynomial approximation as described in Section 3. The Pareto front between card(b) and the polynomial fit serves as an approximation of the trade-off between the LFR order and the accuracy of the LFR model.

The starting value of λ is is chosen in accordance with the following proposition.

Proposition 1 *Setting λ in (7) to*

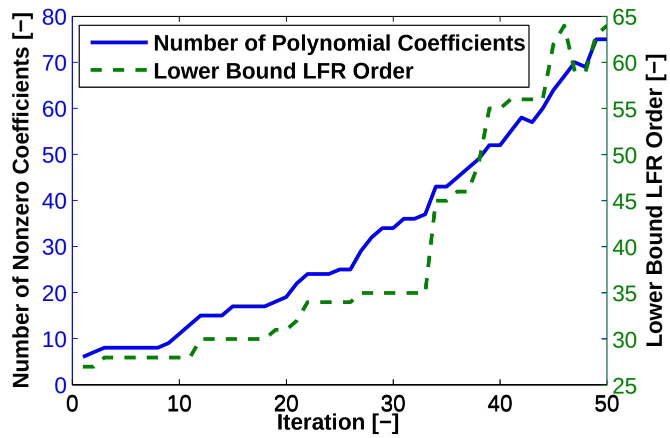
$$\lambda_0 = 2\max_i\left(w_{1,i}\max_j\left(\frac{|(X^Ty_i)_j|}{w_{2,j}}\right)\right) \tag{9}$$

yields a constant approximation, i.e. $b = 0$.

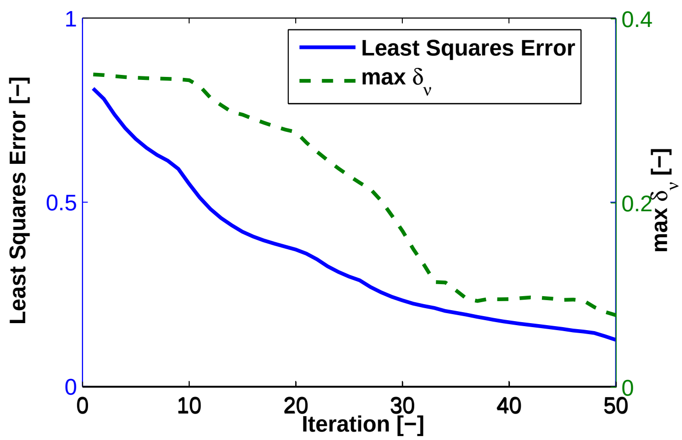
For the proof the reader is referred to [11]. Incrementally decreasing λ will result in steadily better fits with higher card(b).

An advantage of this approach is that a lot of computations can be done upfront and reused at each iteration. Additionally, the solution of the last iteration is used as the starting point of the new iteration. Hence, performing a sweep over λ is computationally comparably cheap.

To demonstrate the validity of the convex approximation some brief results of the polynomial approximation used in the missile model with nonlinear dynamic inversion controller are presented (see Section 5 for more details). The elements that require approximation in the missile model are the aerodynamic force and moment coefficients.



(a) Number of Polynomial Coefficients vs Lower Bound LFR Order



(b) Least Squares Residual vs Maximum ν-Gap Metric

Fig. 1 Results of ℓ_1-Regularized Least Squares Problem for LFR Model Generation

In Fig. 1, the difference between the cost function of the convex problem (7) and the lower bound of the LFR order as well as the maximum ν-gap metric $\delta_{\nu,max}$ is presented. As shown in the upper figure the number of polynomial coefficients and the lower bound LFR order m_{LB} follow a similar trend. However, it can be seen that it is possible to increase the cardinality of the coefficient vector b without increasing m_{LB}.

A likewise statement can be made about the relation between the sum of the residuals $\sum_i \|Xb_i - y_i\|_2^2$ and the maximum ν-gap metric between the LFR and the grid-point LPV model. Minimizing the residual in general seems to also lower $\delta_{\nu,max}$. Still, it has to be kept in mind that this is only a heuristics and no direct relation between the residual and some system metric can be established.

4.1 Weighting of the Elements

The quality of the LPV approximation is only accounted for in the weighted sum of the polynomial approximation errors in a least squares sense. Since not all elements s_i have the same significance for the LPV model, equally weighting them would not reflect the major aim to find an LFR of good accuracy and low order. Hence, for each element s_i a so-called influence coefficient IC_i is determined.

An element s_i has a low influence coefficient if its variation among the set of grid point models does not significantly influence the transfer matrix of the frozen models in terms of a specified error metric. For each s_i a set of transfer matrices $\{G_{k_i}\}_1^m$ is generated, which is equal to the set $\{G_k\}_1^m$ except that for s_i the mean value of the m grid point values y_i is chosen. Finally, the influence coefficient IC_i of s_i is defined as

$$IC_i = \max_k(\delta_\nu(G_k, G_{k_i})),\ k = 1, \ldots, m, \tag{10}$$

where δ_ν denotes the ν-gap metric between G_k and G_{k_i}.

The IC can be directly used as weighting w_1 in the convex approach. Hence, the algorithm is biased towards minimizing the approximation errors of elements with a high IC.

In order to show the advantage of using the influence coefficient as weighting w_1, a comparison between a weighted sum and weighting each elements equally is made. The example uses again the data from the missile model. In Fig. 2 the maximum error in terms of the ν-gap metric $\delta_{\nu,max}$ is shown over the lower bound on the LFR order for a sweep over the regularization parameter λ. The case, where each approximation error is weighted equally, is depicted by the circles. The crosses represent the results of the approximation with using the influence coefficients of the elements as weightings w_1 in (7). As can be seen in the figure, the weighted ℓ_1-regularized problem yields much better results for most λ in comparison to the unweighted problem. Note, that in both cases not each point is actually Pareto optimal in terms of ν-gap and LFR order.

In addition to its usage to weight the elements, the influence coefficients are also used to determine which elements can be considered constant in the proposed approach. Is the influence coefficient of an element below a specified threshold, the respective element will be approximated by its constant mean value over all grid points.

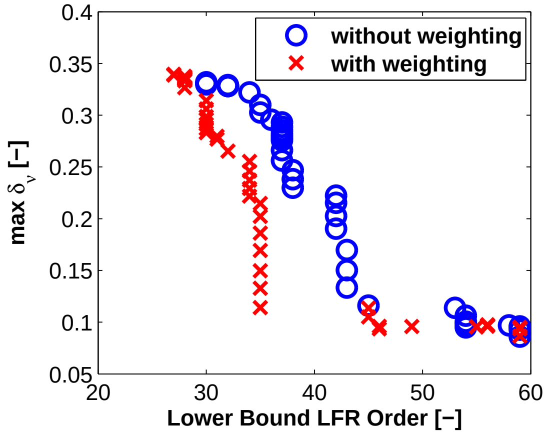


Fig. 2 Effect of Using Influence Coefficients as Weightings in the Optimization

4.2 Weighting of the Polynomial Coefficients

Instead of minimizing the lower bound LFR order a weighted ℓ_1-norm over all possible polynomial coefficients is used. The weighting w_2 shall be chosen such as to penalize polynomial bases which would lead to higher order LFRs. A good heuristics for choosing w_2 is to set $w_{2,j}$ to the degree of the respective polynomial basis $g_j(x)$. For instance considering the monomial basis $g_j(x) = x_1^2 x_2^2$, the corresponding weighting factor $w_{2,j}$ would be four. This also coincides with the minimum achievable LFR order of g_j.

5 Example: Missile Model

The example is based on a nonlinear model of a modern air defense missile. The missile is in a cruciform configuration with four fins at the tail. It is axis symmetric with a slender body. A controller based on nonlinear dynamic inversion has been designed for the missile model within [13]. The aim of this work is to obtain an LFR of the closed loop missile, which is suitable for modern LFT based robust stability analysis.

The mathematical model has six states, namely the velocities in y- and z-direction v and w respectively, the roll rate p, pitch rate q, yaw rate r and the bank angle around the velocity vector Φ_V. The inputs are the deflections angles in aileron ξ, elevator η and rudder ξ. As outputs the accelerations in y- and z-direction a_y and a_z, Φ_V and the angular velocities $\Omega = [p,q,r]^T$ are available. The parameter vector δ consists of the Mach number Ma, the angle of attack α and the side slip angle β.

In the following, the general differential equations of momentum and angular momentum are given in the body-fixed coordinate system. F and M represent the external forces and moments respectively acting on the missile and I_T the inertial tensor.

$$\begin{bmatrix}\dot{u}\\ \dot{v}\\ \dot{w}\end{bmatrix} = \frac{1}{m}\sum F - \begin{bmatrix}p\\ q\\ r\end{bmatrix} \times \begin{bmatrix}u\\ v\\ w\end{bmatrix} \tag{11}$$

$$\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = I_T^{-1}\left(\sum M - \begin{bmatrix}p\\ q\\ r\end{bmatrix} \times \left(I_T \begin{bmatrix}p\\ q\\ r\end{bmatrix}\right)\right) \tag{12}$$

Note that the acceleration in x-direction $\dot{u}$ has been neglected in the quasi-LPV model, as the control has no influence on them.

In the following equations the C_i are the aerodynamic coefficients which are nonlinear functions of δ. In addition, ρ is the air density, $V = Ma\,a$ is the absolute velocity with a being the speed of sound. The other parameters are the reference area S_{ref}, reference length l_{ref}, mass m and moments of inertia I_{xx}, I_{yy} and I_{zz}. For the sake of brevity the auxiliary variables $K_1 = \rho V S_{ref}/(2m)$, $K_2 = \rho V S_{ref} l_{ref}/2$ and $K_3 = \rho V S_{ref} l_{ref}^2/4$ are introduced.

In the quasi-LPV model the forces and moments will be described by their respective dimensionless coefficients, which is common practice in aerospace. These aerodynamic coefficients are nonlinear functions of the parameter vector δ and are only available as discrete table data.

$$\begin{aligned}\begin{bmatrix}F_Y\\ F_Z\end{bmatrix} &= mK_1V\begin{bmatrix}C_{Y0}(\delta)+C_{Y\zeta}(\delta)\zeta\\ C_{Z0}(\delta)+C_{Z\eta}(\delta)\eta\end{bmatrix}\\ \begin{bmatrix}L\\ M\\ N\end{bmatrix} &= K_2V\begin{bmatrix}C_{L0}(\delta)+C_{L\xi}(\delta)\xi\\ C_{M0}(\delta)+C_{M\eta}(\delta)\eta\\ C_{N0}(\delta)+C_{N\zeta}(\delta)\zeta\end{bmatrix} + K_3\begin{bmatrix}C_{Lp}(\delta)p\\ C_{Mq}(\delta)q\\ C_{Nr}(\delta)r\end{bmatrix}\end{aligned} \tag{13}$$

The state equations are given in (16), where $A_1(\delta)$ is obtained via a function substitution of the aerodynamic coefficients. The following substitutions have been used:

$$\bar{C}_{i0} = \begin{cases}0, & \text{if } \beta = 0\\ C_{i0}/\sin\beta, & \text{otherwise}\end{cases} \quad \text{for } i = Y, L, N \tag{14}$$

$$\bar{C}_{i0} = \begin{cases}0, & \text{if } \alpha = 0\\ C_{i0}/(\sin\alpha\cos\beta), & \text{otherwise}\end{cases} \quad \text{for } i = Z, M \tag{15}$$

$$\begin{bmatrix} \dot{v} \\ \dot{w} \\ \dot{\Phi}_V \\ \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = A(\delta) \begin{bmatrix} v \\ w \\ \Phi_V \\ p \\ q \\ r \end{bmatrix} + B(\delta) \begin{bmatrix} \xi \\ \eta \\ \zeta \end{bmatrix} \tag{16}$$

$$A(\delta) = \begin{bmatrix} K_1\bar{C}_{Y0} & 0 & 0 & V\sin\alpha\cos\beta & 0 & -V\cos\alpha\cos\beta \\ 0 & K_1\bar{C}_{Z0} & 0 & -V\sin\beta & V\cos\alpha\cos\beta & 0 \\ 0 & 0 & 0 & \cos\alpha\cos\beta & \sin\beta & \sin\alpha\cos\beta \\ K_2/I_{xx}\bar{C}_{L0} & 0 & 0 & K_3/I_{xx}C_{Lp} & 0 & 0 \\ 0 & K_2/I_{yy}\bar{C}_{M0} & 0 & 0 & K_3/I_{yy}C_{Mq} & 0 \\ K_2/I_{zz}\bar{C}_{N0} & 0 & 0 & 0 & 0 & K_3/I_{zz}C_{Nr} \end{bmatrix} \tag{17}$$

$$B(\delta) = \begin{bmatrix} 0 & 0 & K_1VC_{Y\zeta} \\ 0 & K_1VC_{Z\eta} & 0 \\ 0 & 0 & 0 \\ K_2V/I_{xx}C_{L\xi} & 0 & 0 \\ 0 & K_2V/I_{yy}C_{M\eta} & 0 \\ 0 & 0 & K_2V/I_{zz}C_{N\zeta} \end{bmatrix} \tag{18}$$

Note that as a requirement for the application of the inversion based control method, the system has to be minimum phase. This is not the case when considering the accelerations at the center of gravity. As a remedy, in [13], it has been proposed to use accelerations a_y and a_z at a virtual point P instead. If the point P is sufficiently far in front of the center of gravity the system is minimum phase. The output equations for the accelerations at P can be written as

$$a_y = \frac{\bar{q}S_{ref}}{m}C_Y + \dot{r}x_{gp} \qquad a_z = \frac{\bar{q}S_{ref}}{m}C_Z - \dot{q}x_{gp} \tag{19}$$

with x_{gp} being the distance between P and the center of gravity. Using the same function substitution (19) can be written in a suitable form to fit into the quasi-LPV framework. The equations for the remaining outputs, namely Φ_V and Ω, are easily obtained, as both Φ_V and Ω are states of the system.

In addition to the quasi-LPV parameters defined above also uncertainties in the aerodynamic data are considered in the model. For the moment and control surface coefficients (δC_L, δC_M, δC_N and δC_{ctrl}) these are ± 20 percent and for the force coefficients (δC_Y and δC_Z) ± 5 percent.

The controller is a standard nonlinear dynamic inversion based one. It is separated into three parts: the inversion of the rotational dynamics Ω, the inversion of the outer dynamics a_z, a_y and Φ_V and a reference model, which is only used in the feed forward path. A classical PI-controller has been developed for the inverted plant. A detailed description of the controller is found in [13].

5.1 Generation of the LFR Model

In order to transform (16) into an LFR the aerodynamic coefficients need to be approximated by polynomials. The results of this ℓ_1-regularized least squares fitting are shown in Fig. 3. An approximate Pareto front between the lower bound of the LFR order and the approximation error in terms of the ν-gap metric for the plant can be seen. Both the maximum error (dashed red line) and the mean error (solid blue line) over all grid points are provided. The black vertical line designates the chosen iteration for the LFR generation. This iteration has a maximum error $\delta_{\nu,max} = 0.11$, a mean error $\delta_{\nu,mean} = 0.054$ and a lower bound of LFR order of $m_{LB} = 35$.

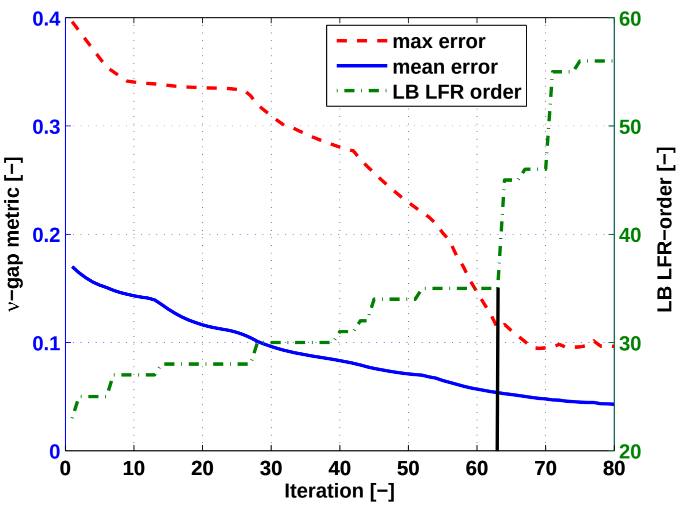


Fig. 3 Results of the ℓ_1-regularized Least Squares Fitting

The nonlinear dynamic inversion controller is already in LPV form. The same polynomial approximations for the aerodynamic coefficients can be used for it. The trigonometric functions in (17) can simply be approximated by a Taylor series expansion and truncation after a sufficient high order.

At this point, the closed loop of the missile benchmark is available as a symbolic description of an LPV model in its general form (2). It only depends rationally on the parameter vector δ. This is a requirement for transforming the system into an LFR. By employing the sophisticated techniques of [14] the resulting closed loop LFR has a dimension of 65, with the Δ-Block having the following structure:

$$\Delta = \text{diag}(MaI_{25\times 25}, \alpha I_{19\times 19}, \beta I_{11\times 11}, \delta C_y, \delta C_z, \delta C_l, \delta C_m, \delta C_n, \delta C_{ctrl} I_{5\times 5}) \quad (20)$$

5.2 Model Assessment

In order to show that the closed loop LFT system still closely matches the original nonlinear system, a Monte-Carlo simulation is conducted. To estimate the error, which has been introduced due to the various approximation steps, the LFT system is run in a parallel setup with the fully nonlinear model. The error is then measured in terms of the maximum of a relative $\mathscr{L}_2$-norm over a finite time horizon, which is defined as

$$error = \max_i \frac{\left(\int_{t0}^{t1} (y_{nls,i} - y_{lfr,i})^2 dt\right)^{0.5}}{\left(\int_{t0}^{t1} y_{nls,i}^2 dt\right)^{0.5}} \tag{21}$$

with $y_{nls} = [\Phi_{V,nls}, a_{z,nls}, a_{y,nls}]^T$ and $y_{lfr} = [\Phi_{V,lfr}, a_{z,lfr}, a_{y,lfr}]^T$.

Simultaneous sinusoidal sweeps in all three command channels are applied as input signals for the nonlinear simulation. The amplitude of the signals are 10°, 20m/s^2 and 10m/s^2 for the Φ_V-, a_z- and a_y-channel respectively. The frequencies from the sinusoidal sweeps range from 0.1 Hz to 10 Hz.

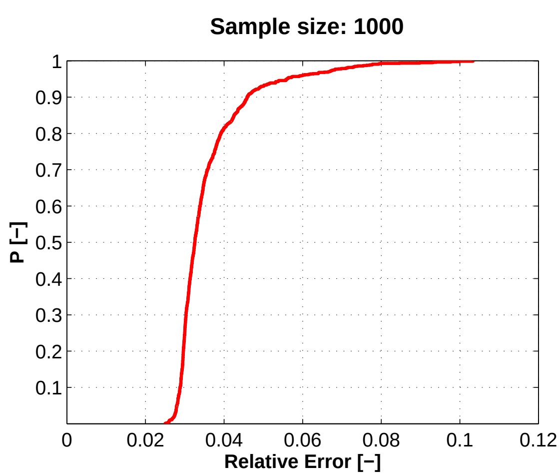


Fig. 4 Statistical Results of the Monte-Carlo Simulation

The only parameters considered in the Monte-Carlo simulation are Ma, α and β. All parameters corresponding to model uncertainty are set to their respective maximum values. The results of the Monte-Carlo run with 1000 samples are shown in Fig. 4 in form of the cumulative distribution function (CDF). The CDF gives the percentage of simulation runs which are less than a specified error. The samples are uniformly distributed over the considered flight envelope spanned by $Ma = [0.9, 4.4]$,

$\alpha = [0,25]^\circ$ and $\beta = [0,10]^\circ$. In the CDF, it is seen that in 90 percent of the cases the error is less than 5 percent. The worst case found in the Monte-Carlo run is 10.2 percent.

The time history of the worst case found in the Monte-Carlo simulation is shown in Fig. 5, i.e. the error is 10.2 percent. It can be seen in the figure that the LFR model still matches the original nonlinear model well.

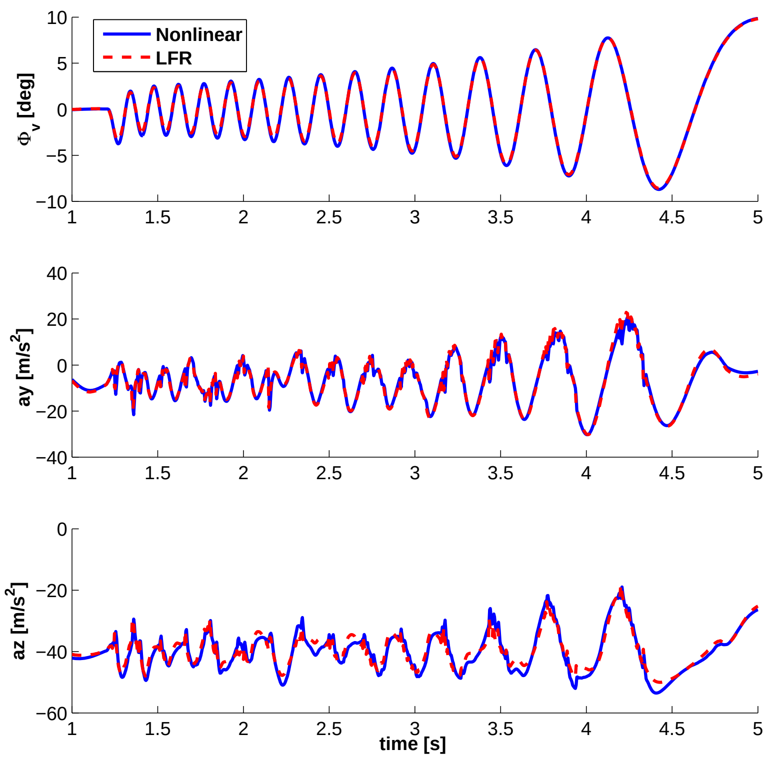


Fig. 5 Comparison Simulation between Nonlinear Model and LFR

6 Conclusion

A very general algorithm for generating LFT models has been developed, which can be applied to arbitrary nonlinear systems, as long as the system behavior can be accurately described/approximated with polynomial or rational parametric statespace systems. In order to efficiently generate LFT models, a convex relaxation has been proposed. It is very time efficient and can compute an almost Pareto front between the LFR order and accuracy.

In the present work, this algorithm has been successfully applied to an industrial benchmark problem. LFRs of high accuracy and reasonable order could be generated for a highly nonlinear missile model. The quality of the LFRs has been assessed using Monte-Carlo simulations.

In the future, it is contemplated incorporating the approximation error made during the polynomial fitting as a dynamic unstructured uncertainty. Methods as the ones described in [15] can likely be adopted for this purpose. Such mixed parametric dynamic uncertainty models might be better suited for controller synthesis purpose.

References

1. K. Zhou and J. C. Doyle. *Essentials of Robust Control*. Prentice Hall, 1998.
2. D.J. Leith and W.E. Leithead. Survey of gain-scheduling analysis and design. *International Journal of Control*, 73(11):1001–1025, 2000.
3. W. Tan. Applications of linear parameter varying control theory. Master's thesis, University of California at Berkeley, 1997.
4. A. Marcos and G. Balas. Development of linear-parameter-varying models for aircraft. *Journal of Guidance, Control and Dynamics*, 27, 2004.
5. S. Hecker. *Generation of low order LFT Representations for Robust Control Applications*. VDI Verlag, 2007.
6. H. Pfifer and S. Hecker. Generation of optimal linear parametric models for LFT-based robust stability analysis and control design. In *Proceedings of IEEE Conference on Decision and Control*, 2008.
7. H. Pfifer, S. Hecker, and G. Michalka. Lft-based stability analysis of generic guided missile. In *Proceedings of the AIAA Guidance, Navigation and Control Conference*, 2009.
8. S. Hecker and H. Pfifer. *Optimization Based Clearance of Flight Control Laws*, chapter Generation of LPV Models and LFRs for a Nonlinear Aircraft Model, pages 39–58. Springer, 2011.
9. G. Vinnicombe. Frequency domain uncertainty and the graph topology. *IEEE Transactions on Automatic Control*, 38, 1993.
10. S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
11. S. Kim, K. Koh, M. Lustig, S. Boyd, and G. Gorinevsky. An interior-point method for large scale ℓ_1-regularized least squares. *IEEE Journal on Selected Topics in Signal Processing*, 2007.
12. B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *The Annals of Statistics*, 32, 2004.
13. F. Peter, M. Leitao, and F. Holzapfel. Adaptive augmentation of a new baseline control architecture for tail-controlled missiles using a nonlinear reference model. In *Proceedings of the AIAA Guidance, Navigation and Control Conference*, 2012.
14. S. Hecker, A. Varga, and J. Magni. Enhanced LFR-toolbox for Matlab. *Aerospace Science and Technology*, 9, 2005.
15. H. Hindi, C.-Y. Seong, and S. Boyd. Computing optimal uncertainty models from frequency domain data. In *Proceedings of Conference on Decision and Control*, 2002.