Robustness Analysis Applied to Autopilot Design Part 1: μ-Analysis of Design Entries to a Robust Flight Control Benchmark

G. Looye*, **A. Varga***, **S. Bennani**†, **D. Moormann*** and **G. Grübel***

* *DLR-Oberpfaffenhofen, Institute of Robotics and System Dynamics, D-82234 Wessling, Germany.*
E-mail: `Gertjan.Looye@dlr.de`
† *TU Delft, Aerospace Engineering, Kluyverweg 1, 2629 HS Delft, The Netherlands*

Abstract

μ-Analysis is a powerful tool for the assessment of the stability of uncertain parametric systems by means of the Structured Singular Value μ. The peak upper bound value of μ over a frequency range provides information on the stability margin of a system for given variations of uncertain parameters, while the computed lower bound on μ allows to obtain worst-case parameter combinations destabilizing the system. The applicability of μ-analysis is however conditioned by the availability of adequate uncertainty models based on Linear Fractional Transformations (LFTs). For complex systems, like aircraft models, LFT modeling is a very demanding and time-consuming task.
In this paper we present the generation of an LFT-based uncertainty model for a civil aircraft, starting from a nonlinear dynamic model with explicit parametric dependencies. This nonlinear model was the basis for the design of twelve different flight controllers according to identical specifications. We applied μ-analysis for stability robustness assessment of the twelve control configurations to uncertainties in the aircraft mass, the center of gravity location, and the on-line computational time delay. We determined the corresponding stability margins and the worst-case destabilizing parameter combinations. We used nonlinear simulations to validate our results.

1. Introduction

Flight control laws require extensive validation before they can be implemented in a flight control computer. An important step in this validation is the assessment of performance and stability robustness of the closed loop system to potential inaccuracies in the available model. These inaccuracies can principally be expressed as uncertainties in physical parameters, like for example the aircraft mass.

In such a robustness assessment, the designer wants to figure out if the required performance level and stability are maintained for all possible parameter combinations within their assumed bounds. Possibilities to do this are for example Monte-Carlo type simulations, computing eigenvalues over a grid of parameter values, or a worst-case parameter search via optimization.(13)

In this respect, μ-analysis is an interesting option. This analysis method basically consists of two steps. First, the set of linear models is captured as a function of the uncertain parameters, and transformed into a Linear Fractional Transformation (LFT) representation. In an LFT the uncertain parameters are normalized, pulled out of the system, and augmented in a so-called Δ-matrix: the unknowns are thus separated from the rest of the model.
Next, the μ-value is determined, indicating the magnitude of the worst-case Δ. Exact computation of μ is a hard problem, but algorithms for tight upper and lower bounds are available.(2) If μ is smaller than one, the worst-case is outside the parameter bounds. Hence, the required performance level or stability is guaranteed within the parameter bounds.

In the period 1995-1997 an action group of the Group for Aeronautical Research and Technology in EURope (GARTEUR) worked on a robust flight control project, assessing the applicability of modern robust control design concepts to flight control problems.(14) To this end, two benchmarks were defined, based on the so-called Research Civil Aircraft Model (RCAM(11)) and the High Incidence Research Model (HIRM(12)), representing a high-performance jet-fighter. Teams from member organizations of the action group made 12 designs for the RCAM (10 different methods), and 6 designs for the HIRM (5 methods) * .

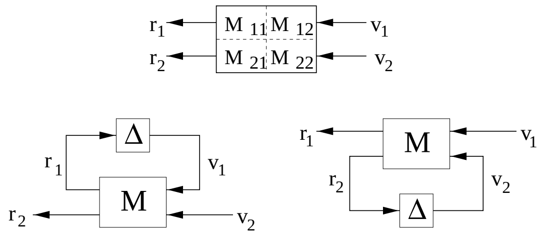
In the final phase of the project, the designs and the applied methods were evaluated by members from industry. In addition, we assessed stability robustness of the RCAM designs with μ-analysis. This paper describes how we performed this analysis, and the results that we obtained.

The paper is organized as follows. We briefly review LFT modeling and the structured singular value. Next, we describe the RCAM and show how we obtained an LFT description of the model. We perform μ-computations, and validate the found worst-cases in nonlinear simulations. Finally we will summarize our findings and indicate future plans.

2. Review of LFTs and μ

A lot of material has been published about LFTs and μ, see Refs.[7, 8, 18, 23, 24] We will give a brief review here.

Let M be a complex matrix $M \in \mathcal{C}^{m \times n}$, relating two pairs of signals, r_1, v_1 and r_2, v_2 respectively. We may close the loop using either signal couple via matrix Δ:



so that $v_2 = \mathcal{F}_u(M,\Delta)r_2$, and $v_1 = \mathcal{F}_l(M,\Delta)r_1$, with:

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

$$\mathcal{F}_l(M,\Delta) = M_{11} + M_{12}(I - \Delta M_{22})^{-1}\Delta M_{21} \quad (2)$$

The expressions $\mathcal{F}_u(M,\Delta)$ and $\mathcal{F}_l(M,\Delta)$ are called *Linear Fractional Transformations* (LFTs), where the subscripts 'u' and 'l' mean *upper*-LFT and *lower*-LFT, respectively.

LFTs can be used to describe uncertainties in the elements of a matrix. For such a case, Δ is a structured matrix with uncertain nonzero elements. For example, Δ may belong to a set:

$$\mathbf{\Delta} := \{\Delta = \mathrm{diag}\,[\delta_1 I_2, \delta_2, \delta_3, \Delta_4] : \\ \delta_1, \delta_2 \in \mathcal{R}, \delta_3 \in \mathcal{C}, \Delta_4 \in \mathcal{C}^{2\times 2}\}$$

Usually, Δ can be normalized: $\bar{\sigma}(\Delta) \leq 1$, where $\bar{\sigma}$ is the maximum singular value. Note that an LFT can be used to capture a set of possible relations between the free input-output pair in a single representation.

An important reason for using LFTs for representing uncertainty is that standard interconnections (cascade, parallel, or feedback) of LFTs can be rewritten as one single LFT by 'pulling-out' the delta's, as illustrated in Fig. 1. This means that composing a model from LFT-submodels results in a new LFT. What is also apparent from the figure, is that uncertainties that appear locally, scattered all over the place, become structured at a system level in a newly defined structured Δ-matrix.

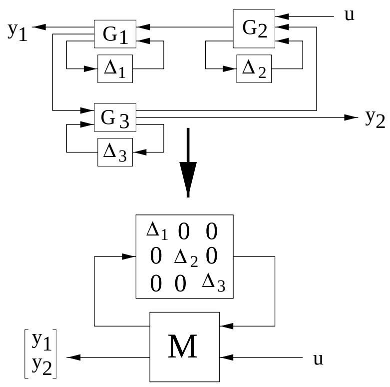


Fig. 1. Interconnection of LFTs

As can be seen from (1), the LFT corresponds to a feedback connection, with Δ in the feedback path. This results in the fractional term in the LFT. An important question is how large Δ may become without making the LFT singular. Since the structure of Δ is problem specific, an indicator is needed that accounts for this structure. This indicator is the Structured Singular Value (SSV), or μ:

$$\mu_{\mathbf{\Delta}}(M) := \frac{1}{\min\{\bar{\sigma}(\Delta) : \Delta \in \mathbf{\Delta}, \det(I - M\Delta) = 0\}}$$

In words: $\mu_{\mathbf{\Delta}}$ *is the reciprocal of the smallest* Δ *(with* $\bar{\sigma}$ *as the norm) that can be found in the set* $\mathbf{\Delta}$ *that makes the matrix* $I - M\Delta$ *singular. If no such* Δ *exists,* $\mu_{\mathbf{\Delta}}$ *is taken to be zero.*

Applying μ to M_{11} (upper LFT) results in the norm of the worst-case perturbation making the LFT singular. If $\mu_{\mathbf{\Delta}} > 1$, there exists a Δ, $\bar{\sigma}(\Delta) \leq 1$ for which the LFT is singular. On the other hand, if $\mu_{\mathbf{\Delta}} < 1$, the worst-case Δ is larger than 1. It is then *guaranteed* that no combination of δ's < 1 exists, that violates the well-definedness of the LFT.

In General:

$$\begin{gathered} \mu_{\mathbf{\Delta}}(M_{11}) < 1 : \\ \Longleftrightarrow \\ \text{LFT is well-defined for all } \Delta \in \mathbf{\Delta},\ \bar{\sigma}(\Delta) \leq 1 \end{gathered} \quad (3)$$

* All GARTEUR reports can be downloaded from `http://www.nlr.nl/public/hosted-sites/garteur/tplst.html`

Exact computation of μ is a hard problem. The best way is to find tight upper and lower bounds. By squeezing the gap between these bounds, a good approximation is obtained. For a good description of the underlying algorithms we refer to Ref,[2] the manual of the μ-Analysis and Synthesis Toolbox for Matlab, which we used for the computations.
Extensive experience has shown that the gap between the upper and the lower bound for mixed real/complex perturbations may be up to $10-20\%$, but is usually smaller.

Robust stability assessment in face of the perturbations in Δ is based on the following theorem:

$$\textit{Robust stability} \iff \mu_{\boldsymbol{\Delta}}(M_{11}(j\omega)) < 1 \quad \forall\omega$$

where M is the transfer matrix of a system. This theorem has an easy interpretation: in fact we walk along the imaginary axis and at each frequency ω we find the smallest Δ required to move a pole over the axis at that frequency. Thus, if $\mu_{\boldsymbol{\Delta}}(M_{11}(j\omega)) < 1\ \forall\omega$, we are *guaranteed* that no pole will travel from one half plane into the other for any $\Delta \in \boldsymbol{\Delta}$, $\bar{\sigma}(\Delta) \leq 1$.

A robust stability test with μ consists of the following steps:

(1) obtain M, define the set $\boldsymbol{\Delta}$,
(2) calculate frequency response of M,
(3) calculate bounds on $\mu_{\boldsymbol{\Delta}}(M(j\omega))$,
(4) find peak value (upper bound),
(5) peak < 1: *system is robustly stable.*

The first step involves the generation of the LFT-based uncertainty description of the plant. Closing the loop with the controller to be analyzed, K, we obtain the system M. In step 3 we calculate the bounds over a chosen grid of frequency points. We have to be careful here: the grid must be dense enough to avoid that a thin peak is missed.

3. The aircraft model

A detailed description of the nonlinear rigid-body equations of motion for an aircraft can be found in.[4,20] More details about the aircraft under consideration are given in.[11,14] We will give a brief description here, introducing some notations simultaneously.

The axis systems are earth-fixed inertial axes F_E, vehicle-carried vertical axes F_V (same orientation as F_E, attached to the center of gravity), body-fixed axes F_B, and wind axes, F_W. The dynamics are described using twelve states:

- $P = [x\ y\ z]_E^T$ position of the vehicle center of gravity in F_E. z is positive downward; the altitude is $h = -z$.
- $\Lambda = [\phi\ \theta\ \psi]^T$ the attitude of the body-fixed axes relative to F_V.
- $\Omega = [p\ q\ r]_B^T$ the angular rates around the x, y, and z axes respectively in F_B.
- $V = [u\ v\ w]_B^T$ the velocity components along the F_B axes.

The vehicle is depicted in fig 2. The model is based on

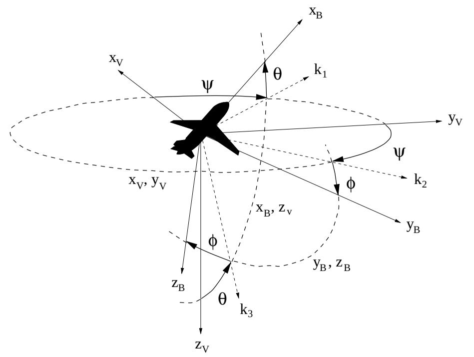


Fig. 2. Attitude of aircraft

the Newton-Euler equations of motion:

$$\dot{\Omega}_B = \mathrm{I}^{-1}\left[\mathrm{M}_A + \mathrm{M}_T - \Omega_B \times \mathrm{I}\,\Omega_B\right] \tag{4}$$

$$\dot{\mathrm{V}}_B = \frac{1}{m}\mathrm{F}_A + \frac{1}{m}\mathrm{F}_T + \mathrm{R}_{BE}(\Lambda)\ \mathrm{F}_G - \Omega_B \times \mathrm{V}_B \tag{5}$$

The following kinematic relations hold:

$$\dot{\mathrm{P}}_E = \mathrm{R}_{EB}(\Lambda)\ \mathrm{V}_B \qquad \dot{\Lambda} = \mathrm{R}_{\phi B}(\Lambda)\ \Omega_B$$

The matrix R_{YX} denotes the transformation from axis system X into axis system Y, I is the inertia tensor, m is the mass, the subscripts A, T, and G denote aerodynamic, thrust and gravity contributions respectively, to the forces and moments. The aerodynamic forces and moments depend on the angular rates Ω, the airspeed V_A, the angle of attack α, and the sideslip angle β. The wind speed vector in F_B is given as: $W_B = [u_w\, v_w\, w_w]_B^T$. The airspeed vector in F_B is $V_a = V_B - W_B = [u_a\, v_a\, w_a]_B^T$. V_A, α, and β are obtained from:

$$V_A = \sqrt{V_a^T\ V_a}, \quad \alpha = \tan^{-1}(\frac{w_a}{u_a}), \quad \beta = \sin^{-1}(\frac{v_a}{V_A})$$

The aircraft has two engines. The control inputs are throttles (δ_{TH1}, δ_{TH2}), ailerons δ_A, elevator δ_E, and rudder δ_R. The engine dynamics and the actuator dynamics of aerodynamic control surfaces are represented by simple first order models, with rate and position limits.

The measured outputs are:

$$\begin{aligned} y_1 &= [q\, n_x\, n_z\, w_V\, z\, V_A\, V]^T \text{ (longitudinal)} \\ y_2 &= [\beta\, p\, r\, \phi\, u_V\, w_V\, y\, \chi]^T \qquad \text{(lateral)} \\ y_3 &= [\psi\, \theta\, \alpha\, \gamma\, x\, n_y]^T \end{aligned}$$

where only y_1 and y_2 may be used for feedback. No sensor models are used. n_x, n_y, and n_z are load factors.

The designers had to cope with uncertain parameters in the model, see table 1. In this table, m is the mass,

parameter	unit	min	max	nominal
m	kg	100 000	150 000	120 000
X_{cg}	m	$0.15\bar{c}$	$0.31\bar{c}$	$0.23\bar{c}$
Z_{cg}	m	$0.0\bar{c}$	$0.21\bar{c}$	$0.0\bar{c}$
τ	s	0.05	0.10	0.075
(V_A)	m/s	1.23 V_{stall}	90	80

Table 1. Parameter ranges RCAM

X_{cg} and Z_{cg} are the horizontal and vertical center of gravity shifts respectively, τ is a computational time delay in the flight control computer, and $\bar{c}$ is the mean aerodynamic chord. Although the designers had to consider varying airspeed as well, they were allowed to use it as a scheduling parameter. We performed the analysis at a constant speed of 80 m/s; the reason is discussed in the following section.

The RCAM was implemented in Dymola,[9,10] an object-oriented modeling package. This is described in more detail in.[17]

4. LFT-modeling of RCAM

μ-Analysis applies to parametrized linear systems. Therefore we need to linearize the aircraft dynamics first, and obtain a state space model that explicitly depends on the uncertain parameters, collected in $p = [m,\, X_{cg},\, Z_{cg}]^T$.

Beginning with the nonlinear equations of motion:

$$\begin{aligned} \dot{x} &= f(x,u,p) \\ y &= h(x,u,p) \end{aligned} \tag{6}$$

we use the following procedure:

(1) Compute an equilibrium point $\{\bar{x}, \bar{u}\}$ of the system, for nominal values of the model parameters p_{nom}:

$$0 = f(\bar{x},\, \bar{u},\, p_{nom})\,. \tag{7}$$

(2) Define

$$\begin{bmatrix} A(p) & B(p) \\ C(p) & D(p) \end{bmatrix} = \begin{bmatrix} \dfrac{\partial F(\delta x,\, \delta u,\, p)}{\partial(\delta x)}, & \dfrac{\partial F(\delta x,\, \delta u,\, p)}{\partial(\delta u)} \\[2ex] \dfrac{\partial H(\delta x,\, \delta u,\, p)}{\partial(\delta x)}, & \dfrac{\partial H(\delta x,\, \delta u,\, p)}{\partial(\delta u)} \end{bmatrix}_{\substack{\delta x=0 \\ \delta u=0}} \tag{8}$$

where

$$\begin{aligned} F(\delta x,\, \delta u,\, p) &:= f(\bar{x} + \delta x,\, \bar{u} + \delta u,\, p) \\ H(\delta x,\, \delta u,\, p) &:= h(\bar{x} + \delta x,\, \bar{u} + \delta u,\, p) \end{aligned} \tag{9}$$

The linearized model (6) around the equilibrium point is:

$$\begin{aligned} \delta\dot{x} &= A(p)\delta x + B(p)\delta u \\ \delta y &= C(p)\delta x + D(p)\delta u, \end{aligned} \tag{10}$$

Note that a linear representation of the nonlinear system is obtained, where the entries in the matrices depend on m, X_{cg} and Z_{cg} in a rational way.

(3) Correct some matrix entries for the dependence of equilibrium points of p. In step (2) we neglected that in fact $\bar{x} = \bar{x}(p)$ and $\bar{u} = \bar{u}(p)$, and thus that a number of the matrix elements depends on the trim condition. Using text-book approximations[4] we identify and correct these elements, if necessary, using polynomial fits as a function of p. Data for these fits is obtained by trimming and linearizing the RCAM at a range of operating points.

(4) Generate rational expressions of the system matrices by replacing m, X_{cg} and Z_{cg} with normalized expressions:

$$\begin{aligned} m &= m_0 + s_m\delta_m &&= 125000 + 25000\,\delta_m \\ X_{cg} &= X_{cg0} + s_{xcg}\delta_{xcg} &&= 0.23\bar{c} + 0.08\bar{c}\,\delta_{xcg} \\ Z_{cg} &= Z_{cg0} + s_{zcg}\delta_{zcg} &&= 0.105\bar{c} + 0.105\bar{c}\,\delta_{zcg} \end{aligned} \tag{11}$$

their substitution in (10) gives:

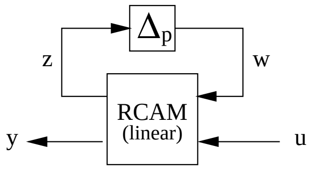
$$\begin{aligned} \delta\dot{x} &= A(\delta p)\delta x + B(\delta p)\delta u \\ \delta y &= C(\delta p)\delta x + D(\delta p)\delta u, \end{aligned} \tag{12}$$

with $\delta p = [\delta_m,\, \delta_{xcg},\, \delta_{zcg}]^T$.

Once the parameter dependent state realization in (12) is available, an LFT representation can be obtained automatically.

Except for the correction of trim-dependencies, the procedure has been automated. For this purpose software tools have been developed utilizing Dymola[9] for modeling, Maple[5] for symbolic computations and MATLAB[16] (with the PUM-toolbox[21] and μ-Tools[2]) for numerical computations. The procedure and the software implementation are described in detail in Refs..[17,22] The intention was to take dependence of the (trimmed) airspeed V_A into account. However, the order of δ_V in Δ obtained from PUM became very large and was certainly non-minimal.[22] Rather than going into implementing better algorithms for obtaining lower Δ-orders or analyzing longitudinal and lateral dynamics separately, we decided to confine to the nominal design speed of 80 m/s.

We end up with an LFT-model † that looks like:



where Δ_p is:

$$\Delta_p = \begin{bmatrix} \delta_m I_{17} & 0 & 0 \\ 0 & \delta_{xcg} I_{15} & 0 \\ 0 & 0 & \delta_{zcg} I_3 \end{bmatrix} \qquad (13)$$

The output vector y contains the measurements that may be used by the control system (see sect. 3) and u contains the control inputs. The obtained realization is non-minimal, but of sufficiently low order to allow the use of μ-analysis.

LFT-model the for uncertain delay

The time delay τ is approximated with a first order Padé-filter:

$$e^{-\tau s} = \frac{2 - \tau s}{2 + \tau s} \qquad (14)$$

This approximation is reasonable up to a frequency of ± 10 rad/s. With $\tau \in [0.05, 0.10]$ s, we scale this parameter as follows:

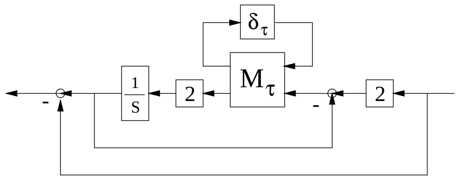
$$\tau = \tau_0 + s_\tau \delta_\tau = 0.075 + 0.025 \delta_\tau \qquad (15)$$

The inverse of this expression is written as an LFT, $\mathcal{F}_u(M_\tau, \delta_\tau)$, with:

$$M_\tau = \begin{bmatrix} -\frac{s_\tau}{\tau_0} & \frac{1}{\tau_0} \\ -\frac{s_\tau}{\tau_0} & \frac{1}{\tau_0} \end{bmatrix}$$

† This model is available, contact `Gertjan.Looye@dlr.de`

In a picture the Padé-filter looks like:



In the closed-loop system the delay occurs between the controller and the actuators/engines.[11] Therefore Padé-filters are placed at the five actuator/engine inputs. The five filters are augmented in a single LFT description: $\mathcal{F}_u(\mathrm{M}_\tau, \Delta_\tau)$. Since δ_τ is identical for each input, the diagonal of Δ_τ consists of five repeated δ_τ's: $\Delta_\tau = \delta_\tau I_5$.

Interconnection structure for analysis

For analysis of the controllers, we have to interconnect all the subsystems. This is depicted in Fig. 3. Between

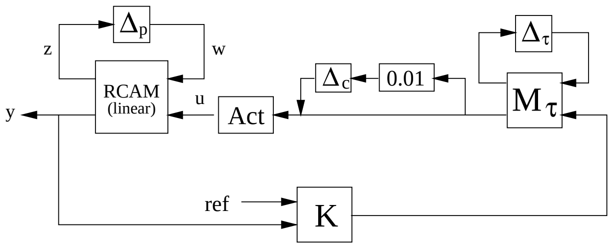


Fig. 3. Interconnection of the sub-systems

the LFT-model for the delay and the actuators/engines (Act) a small extra complex perturbation is added. Since $\bar{\sigma}(\Delta_c) \leq 1$, the perturbation is only 1%. The Δ-structure now contains complex elements, which make the computation of the μ-bounds more tractable.[2] Δ_c is diagonal and consists of five independently varying δ's ($\in \mathcal{C}$).

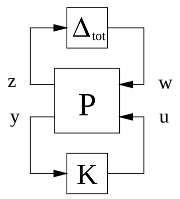


Fig. 4. The final LFT-system for analysis

The final LFT-representation is depicted in Fig. 4. In this figure the system P consists of the RCAM and the actuator/engine models. The controller K is drawn as a separate block. In this block we may implement each (linearized) controller. The closed-loop is obtained as a lower LFT:

$$M_{tot} = \mathcal{F}_l(P, K)$$

$\Delta_{tot} \in \mathbf{\Delta}_{tot}$ is the complete perturbation structure:

$$\mathbf{\Delta}_{tot} := \{\mathrm{diag}\,(\delta_m I_{17},\, \delta_{xcg} I_{15},\, \delta_{zcg} I_3,\, \delta_\tau I_5,\, \delta_{da},\, \delta_{dt},\, \delta_{dr},\, \delta_{dth1},\, \delta_{dth2}) : \delta_m,\, \delta_{xcg},\, \delta_{zcg},\, \delta_\tau \in \mathcal{R}\;\; \delta_{da},\, \delta_{dt},\, \delta_{dr},\, \delta_{dth1},\, \delta_{dth2} \in \mathcal{C}\} \quad (16)$$

The total order of Δ is thus 17 + 15 + 3 + 5 + 5*1 = 45.

LFT-modeling is the most time consuming task in μ-analysis: once the model is available, the actual analysis is a matter of running a single computation routine.

5. Robustness assessment using μ-analysis

The robustness analysis of the controllers with μ consists of the following steps:

(1) Linearize the controller. Prior to numerical linearization with LINMOD,[15] derivative blocks are replaced by first-order high-pass filters.
(2) Make a grid of frequency points along the imaginary axis. The frequency range of interest for the designs is: $10^{-1} \leq \omega \leq 10^{1.5}$ rad/s. In this interval we select 100 logarithmically spaced frequency points.
(3) Interconnect the linear controller with the LFT-model (Fig. 4) using SYSIC in μ-Tools[2] and calculate the frequency response at the selected frequencies.
(4) Calculate the bounds of μ at each frequency point with the μ-Tools command MU.[2] MU returns the bounds, the optimal D–G scales and a destabilizing perturbation.
(5) Plot the μ-bounds, find the peak and the corresponding frequency.
(6) Figure out the worst-case perturbation from the lower bound, at the frequency where the μ-peak occurs.
(7) Verify the perturbation in a nonlinear simulation.

We will discuss the analysis of the preliminary RCAM design[1] in detail, and comment on some of these steps. Next, we present the results with the other RCAM controllers.

5.1 Example μ-analysis

The design of the preliminary controller for RCAM is described in.[1] This design is basically linear, so that linearization of the SIMULINK structure goes without problems. The steps 1–5 are straightforward. The obtained μ-plot is given in Fig. 5. We can see that

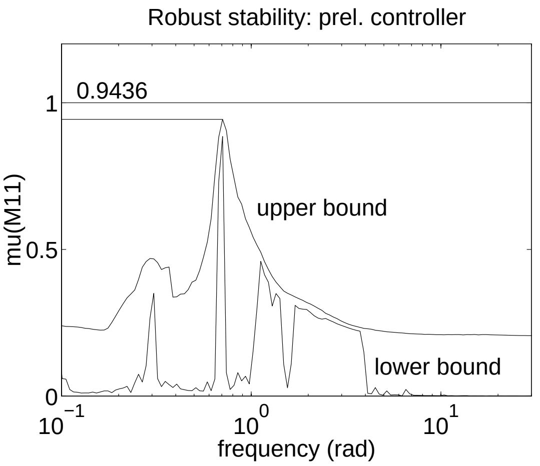


Fig. 5. μ-plot of the example controller

the upper bound shows a peak of 0.94 at a frequency of $\omega \approx 0.7$ rad/s. The lower bound calculation is poor, except near the peak.

The peak of the upper bound is less than 1. This guarantees that we can not find any $\Delta \in \mathbf{\Delta}_{tot}$, $\bar{\sigma}(\Delta) \leq 1$ that destabilizes the system. In other words, at least one of δ_m, δ_{xcg} etc. has to be larger than one to destabilize the closed-loop system, and thus the worst-case constellation of parameters is *outside* the operating range.
On the other hand, the robust stability margin is small. The critical $\bar{\sigma}(\Delta)$ will be slightly larger than one: $1/\mu_{peak} \approx 1/0.94 = 1.06$. The stability margin is thus only 6%. It is expected that the system gets unstable just outside the operating range of the parameters.

It is possible to find the lowest values of the parameters leading to instability. We can obtain a critical $\Delta = \Delta_{crit}$ from the maximum of the lower bound, close to the worst-case.
We can check the stability of eigenvalues by substitution of Δ_{crit} in the analysis structure (Fig. 4). With the complex δ's set to zero, we found a worst-case real perturbation. The critical δ's can be found in Table 2.

parameter	$\delta_{..}$	value
mass	δ_m	1.1092
X_{cg}	δ_{xcg}	1.1092
Z_{cg}	δ_{zcg}	1.1092
delay	δ_τ	1.1092
compl. pert.	$\delta_{da...dth2}$	0

Table 2. Worst-case perturbations

Note that $\bar{\sigma}(\Delta_{crit}) = 1.1092$, and thus $1/\bar{\sigma}(\Delta_{crit}) \approx 0.90$. The small gap compared to the upper bound peak of 0.94, is caused by the complex δ's and the use of D-G scalings[2] for computation of the upper bound.

In table 3 we give the critical eigenvalue pair of the closed-loop system for three different parameter combinations. We observe that, increasing the perturbation from $0.95\Delta_{crit}$ to $1.05\Delta_{crit}$, the eigenvalue pair moves across the imaginary axis. Note that the real part

$0.95\ \Delta_{crit}$	Δ_{crit}	$1.05\ \Delta_{crit}$
-0.0137 ± 0.6954i	0.0 ± 0.6899i	+0.0131 ± 0.6844i

Table 3. Eigenvalues passing the imag. axis

corresponds to the location of the μ-peak, found at $\omega \approx 0.7$ rad/s.

It is remarkable that all critical $\delta's$ are identical. This appeared to be inherent to the RCAM; for all controllers stability improves or deteriorates monotonously with each of the scaled parameters near the edges of the operating range. The equal values then result from the norm we use for Δ. For example, $\Delta = \operatorname{diag}(1.1092, 1.1092, 1.1092, ...)$ and $\Delta = \operatorname{diag}(1.1092, 0, 0, ...)$ both have a norm $\bar{\sigma}(\Delta) = 1.1092$, but the first case is worse, as in the second case we need to increase the non-zero parameter considerably (and thus $\bar{\sigma}(\Delta)$) in order to destabilize the system.

For the critical δ's substituted in (11) and (15), we obtain:

$$\begin{aligned} m &= 125\,000 + 25\,000 * 1.092 = 152\,300\,\mathrm{kg} \\ X_{cg} &= 0.23\bar{c} + 0.08\bar{c} * 1.092 = 0.317\bar{c} \\ Z_{cg} &= 0.105\bar{c} + 0.105\bar{c} * 1.092 = 0.220\bar{c} \\ \tau &= 0.075 + 0.025 * 1.092 = 0.102\,\mathrm{s} \end{aligned} \tag{17}$$

Note that these values are only slightly outside the ranges specified in table 1.

As a final verification we implement the controller with the original nonlinear aircraft model and perform simulations. As we did for Δ_{crit} in (17), we use (11) and (15) to compute two additional parameter combinations, corresponding to $0.95\Delta_{crit}$ and $1.05\Delta_{crit}$.

We trim the aircraft with the three parameter sets and perform nonlinear simulations. The results of a small single block-shaped wind input are shown in Fig. 6.

It is interesting to see that for $1.05\Delta_{crit}$ the simulation shows unstable, and for $0.95\Delta_{crit}$ it shows stable behavior. For Δ_{crit} the response is oscillatory, being at the limit of stability.

5.2 Analysis results

The analysis of the other controllers goes in a very similar way. We will give the results and only comment on interesting details.

The μ-plots (upper bounds) of all designs can be found on the last two pages. The results are summarized in Table 4. Most of the columns in the table are self-explanatory. The designs and methods are described

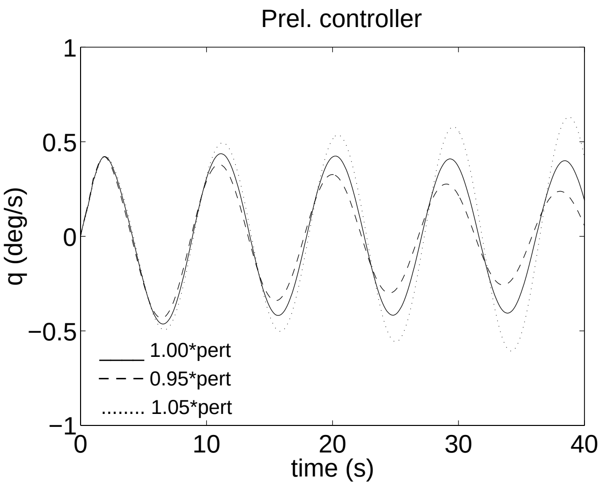


Fig. 6. Nonlinear simulations with worst-case parameters

in.[14] μ_{peak} is the peak of the upper bound, *freq.* is the frequency in the μ-plot where this peak occurs. At this frequency the critical eigenvalue crosses the imaginary axis. The worst-case (scaled) parameter combination is also given, with the motion that becomes unstable (lateral or longitudinal).
The value $1/\bar{\sigma}(\Delta_{crit})$ gives an indication of the conservatism of the found upper bound peak. This conservatism is caused by setting the complex perturbations to zero (table 2), and by approximating μ with an upper bound using D-G scalings.[2] The percentage in the column *consv.* is computed from $[(\mu_{peak} - 1/\bar{\sigma}(\Delta_{crit})]/[1/\bar{\sigma}(\Delta_{crit})]$.

There are considerable differences between the controllers. The controllers LY-14(1), MS-19 and HI-21 do not meet the robustness specifications, as $\mu_{peak} > 1$. Controllers MO-16 and MM-12 are very robust in face of the considered parameters: $\mu_{peak} \ll 1$. The $\delta's$ may be increased with almost a factor three outside their range of $[-1, 1]$ without the closed-loop system going unstable.

In μ-synthesis design[3,19] a μ-value of 0.5 for robust stability is considered as a good value in the trade-off between performance and robustness. It must be noted that both μ-synthesis controllers result from completely different uncertainty descriptions in the synthesis model.

Note that, as in the example, in nearly all cases the critical values of the δ's have equal magnitude: only signs differ. If $\mu_{peak} > 1$, the critical parameter values are all within the operating envelope. We can check this by substitution in (11) and (15).

Typical computation times for μ-analysis on a 166MHz-Pentium PC are about half an hour. Computations for the highest order controllers take somewhat more, but do not cause any numerical problems. This is inher-

No.	Design method	μ_{peak} bnd.	freq. *rad/s*	worst-case parameters				mode (lon/lat)	$1/\bar{\sigma}(\Delta_{crit})$	consv.
				δ_m	δ_{xcg}	δ_{zcg}	δ_τ			
MO-16	Multi-objective Param. Synth.	0.35	6.0	-2.92	2.92	2.92	2.92	lon	0.34	3%
MM-12	Modal Multi-model Control	0.36	8.0	-2.95	-2.95	-2.95	2.95	lon	0.34	6%
EA-22	Eigenstructure Assignment	0.39	0.5	2.67	2.67	2.67	2.67	lon	0.38	3%
FL-15	Fuzzy Logic	0.44	5.5	-2.35	-1.82	-2.35	2.35	lon	0.43	2%
MS-11	μ-Synthesis	0.49	2.9	-2.21	-2.21	2.21	2.21	lat	0.45	9%
			0.6	2.21	-2.21	-2.21	2.21	lon	0.45	9%
CC-13	Classical Control	0.51	0.8	2.01	2.01	2.01	2.01	lon	0.50	2%
LY-14(2)	Lyapunov	0.57	0.8	1.84	1.84	1.84	1.84	lon	0.54	6%
MF-25	Model Following	0.65	0.6	1.54	1.54	1.54	1.54	lon	0.65	1%
EA-18	Eigenstructure Assignment	0.83	7.0	-1.30	1.30	1.30	1.30	lon	0.77	8%
HI-prel	H_∞ Loop Shaping	0.94	0.7	1.11	1.11	1.11	1.11	lon	0.90	4%
LY-14(1)	Lyapunov	1.14	0.5	0.90	0.90	0.90	0.90	lon	1.11	3%
MS-19	μ-Synthesis	1.36	15.1	-0.76	-0.41	0.76	0.76	lon	1.32	3%
HI-21	H_∞ Loop Shaping	1.53	1.3	0.67	0.67	0.67	0.67	lon	1.49	3%

Table 4. Results of the μ-analysis

ent to the followed approach: the algorithm computes tight bounds, rather than considering the behaviour of individual poles due to parameter variations. The computation time is primarily determined by the order of the Δ-block and the number of frequency points. In the following paragraphs we briefly discuss some interesting aspects of our analysis.

Controller MS-11 The upper bound for this controller is depicted in Fig. 7: it shows two equal peaks. Two Δ's (table 4) are found for which an eigenvalue passes the imaginary axis. For the left peak a longitudinal and for the right peak a lateral mode goes unstable. Since the critical Δ's have magnitude 2.21 the instability occurs far outside the specified parameter ranges. The stability margin is thus good.

Controller LY-14(1,2). Although an improved controller was submitted (LY-14(2)), the original design LY-14(1) has an interesting feature. In the original RCAM software a small bug existed in the implementation of the vertical center of gravity location. The robustness assessment is based on the corrected model. For the original controller instability is found within the envelope ($\mu > 1$), which is mainly caused by δ_{zcg} (see figs 11). Verification in a nonlinear closed-loop simulation is depicted in fig. 19. With $1.05\Delta_{crit}$ the system becomes unstable (corrected model). If we simulate with the design model (not corrected), the simulation is stable.

Controllers EA-18 and MS-19. These controllers have problems with the time delay τ. The μ-upper bounds in Fig. 14 and 15 show sharp peaks in the higher frequency range. The approximation has been sufficient to detect the problem, but apparently, a first-order Padé approximation is not accurate enough for determining the critical δ_τ. Simulation with the critical Δ immediately results in instability (we removed rate limiters and saturations in the actuators, otherwise limit-cycles arose). Therefore we first verified the analysis with Padé-filters in nonlinear simulations. However, looking for example at EA-18 (Fig. 20), the simulation with $0.95\Delta_{crit}$ is stable with the Padé approximation, but unstable with a pure time delay.

6. Conclusions

We applied μ-analysis for stability robustness assessment of the RCAM design entries. As this assessment is only a part of the complete evaluation of all entries, we will not comment on the quality of the designs; for this we refer to Ref.[14]

We found μ-analysis a potentially useful tool for post-design robustness analysis. Most of the work consists of obtaining a sufficiently accurate LFT-description of the model with uncertain parameters. Especially for aircraft models this is not a trivial task.

In an LFT the uncertain scaled parameters are pulled out of the system and put in the Δ-block, leading to a highly structured uncertainty description. Furthermore, interconnecting LFTs preserves the LFT structure. These aspects make LFTs a very powerful standard form for representing uncertainties.
We succeeded partially in automating the LFT modeling,[17,22] but for example the dependency of the trimmed states on the uncertain parameters required additional fitting of elements in the state-space matrices. The procedure also led to very high orders of Δ.

Once the LFT description is available and we have interconnected the controller, μ-analysis is a matter of a single computation run, for which software tools are readily available.[2] We were able to find the actual worst-case parameter combinations, and to verify them in nonlinear simulations with the original aircraft model.

μ-Analysis is performed by computation of upper and lower bounds over a grid of frequency points. The upper bound gives hard guarantees for the stability margin. We found conservativeness levels between 1 and 9%. Part of this conservativeness was caused by introducing additional complex elements to the Δ-block. This was necessary in order to 'smoothen' the usually very thin peaks in the μ-plots, that could otherwise easily be missed due to the gridding.
It is therefore interesting to use computation methods that avoid frequency gridding. Such a method for the lower bound is described in.[6]

Automated LFT-generation is a great relief. Therefore we intend to improve the algorithms, especially addressing the high Δ-orders, using advanced order-reduction schemes.

7. References

(1) Flight Mechanics Action Group 08. RCAM Preliminary Design Document. Technical Report TP-088-09, GARTEUR, November 1995.

(2) G. J. Balas, J. C. Doyle, K. Glover, A. Packard, and R. Smith. *μ-Analysis and Synthesis TOOLBOX, For Use with MATLAB*. The Math Works Inc., July 1993.

(3) Samir Bennani and Gertjan Looye. RCAM Design Challenge Presentation Document: The μ-Synthesis Approach. Technical Report TP-088-11, GARTEUR, April 1997.

(4) Rudolf Brockhaus. *Flugregelung*. Springer-Verlag, Berlin Heidelberg, 1994.

(5) B.W. Char et al. *Maple V Language Reference Manual*. Springer Verlag, 1991.

(6) Carsten Döll, Jean-François Magni, Gertjan Looye, and Samir Bennani. Robustness analysis applied to autopilot design part 2: Evaluation of a new tool for μ-analysis. In *Proceedings of the 21st ICAS Congress*, Melbourne, Australia, September 1998.

(7) John Doyle, Andy Packard, and Gary Balas. Theory of robust control. Lecture notes of short course given at Delft University of Technology, September 13–15, 1995.

(8) John Doyle, Andy Packard, and Kemin Zhou. Review of LFTs, LMIs, and μ. In *Proceedings of the 30th Conference on Decision and Control, Brighton, England*, pages 1227–1232, December 1991.

(9) H. Elmqvist. Object-oriented modeling and automatic formula manipulation in dymola. In *SIMS '93, Scandinavian Simulation Society*, Kongberg, Norway, June 1993. available from `http://www.dynasim.se/publications.html`.

(10) H. Elmqvist. *Dymola – Dynamic Modeling Language*. Dynasim AB, 1994.

(11) Flight Mechanics Action Group 08. Robust Flight Control Design Challenge Problem Formulation and Manual: the Research Civil Aircraft Model (RCAM). Technical Report TP-088-03, GARTEUR, April 1997. Version 3.

(12) Flight Mechanics Action Group 08. Robust Flight Control Design Challenge Problem Formulation and Manual: the High Incidence Research Model (HIRM). Technical Report TP-088-04, GARTEUR, April 1997. Version 3.

(13) G. Looye, A. Varga, D. Moormann, and S. Bennani. Post-design stability robustness assessment of the rcam controller design entries. Technical Report TP-088-35, GARTEUR, April 1997.

(14) Jean-François Magni, Samir Bennani, and Jan Terlouw (Eds). *Robust Flight Control – A Design Challenge*. Lecture Notes in Control and Information Sciences 224. Springer Verlag, London, 1997.

(15) The Math Works Inc. *Simulink User's Guide*, 1992.

(16) The Math Works Inc. *MATLAB Reference Guide*, 1993.

(17) D. Moormann, A. Varga, G. Looye, and G. Grübel. Robustness analysis applied to autopilot design, part3: Automated generation of lft-based parametric uncertainty descriptions applied to the rcam aircraft model. In *Proceedings of the 21st ICAS Congress*, Melbourne, Australia, September 1998.

(18) A. Packard and J. Doyle. The complex structured singular value. *Automatica*, 29:71–109, 1993.

(19) J. Schuring and R.M.P. Goverde. RCAM Design Challenge Presentation Document: A μ-Synthesis Method. Technical Report TP-088-19, GARTEUR, April 1997.

(20) Brian L. Stevens and Frank L. Lewis. *Aircraft Control and Simulation*. Wiley-Interscience Publication. John Wiley & Sons, Inc., New York, 1992.

(21) J.C. Terlouw and P.F. Lambrechts. A MATLAB Toolbox for Parametric Uncertainty Modelling. Technical Report CR 93455 L, National Aerospace Laboratory, NLR, Amsterdam, 1993.

(22) A. Varga, G. Looye, D. Moormann, and G. Grübel. Automated Generation of LFT-Based Parametric Uncertainty Descriptions from Generic Aircraft Models. Technical Report TP-088-36, GARTEUR, April 1997.

(23) Kemin Zhou with John C. Doyle and Keith Glover. *Robust and Optimal Control*. Prentice Hall Inc., 1996.

(24) Peter M. Young, Matthew P. Newlin, and John C. Doyle. μ Analysis with Real Parametric Uncertainty. In *Proceedings of the 30th Conference on Decision and Control, Brighton, England*, pages 1251–1256, December 1991.

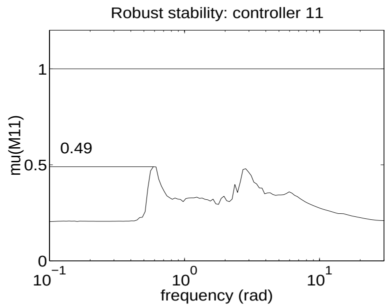


Fig. 7. MS-11 – μ-Synthesis

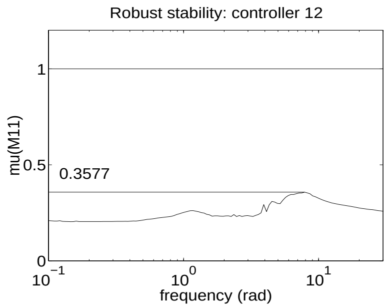


Fig. 8. MM-12 – Modal Multi-Model Synthesis

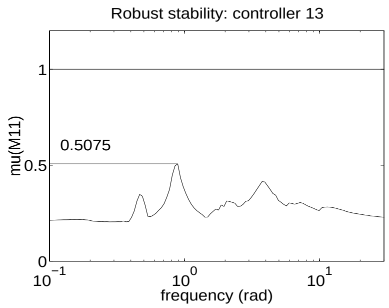


Fig. 9. CC-13 – Classical Control

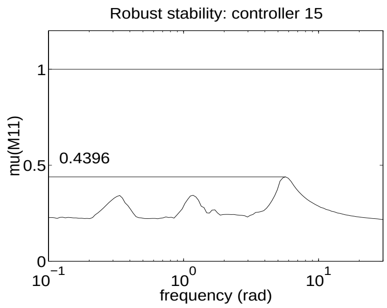


Fig. 10. FL-15 – Fuzzy Logic (linearized version)

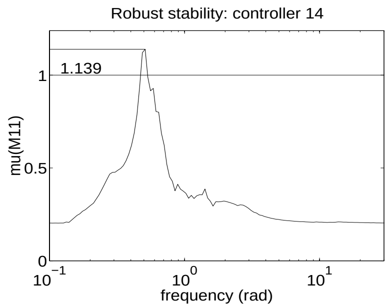


Fig. 11. LY-14(1) – Lyapunov Approach, original entry

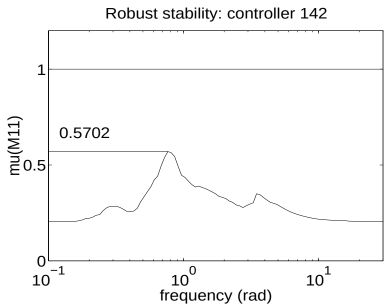


Fig. 12. LY-14(2) – Lyapunov Approach, improved entry

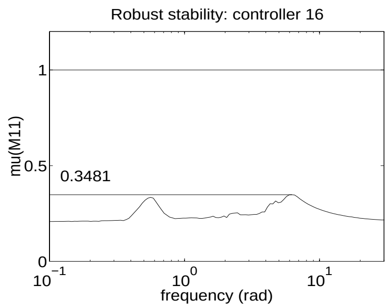


Fig. 13. MO-16 – MOPS approach

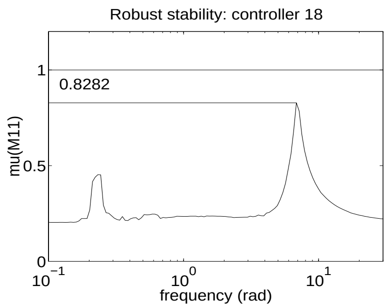


Fig. 14. EA-18 – Eigenstructure Assignment

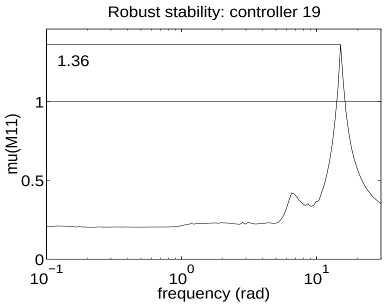


Fig. 15. MS-19 – μ-Synthesis

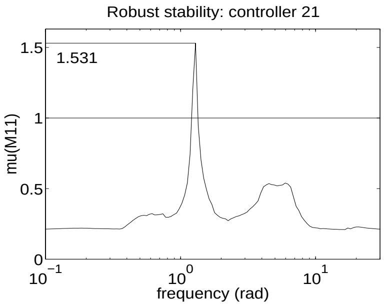


Fig. 16. HI-21 – H_∞

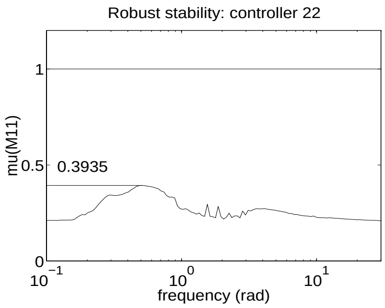


Fig. 17. EA-22 – Eigenstructure Assignment

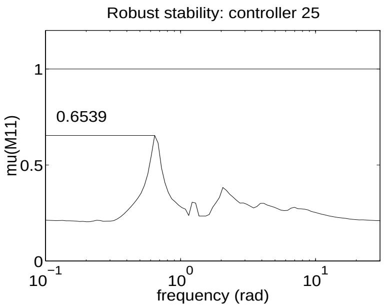


Fig. 18. MF-25 – Model Following

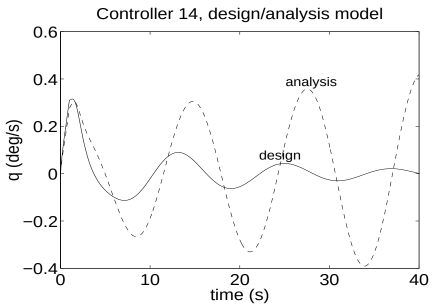


Fig. 19. LY-14 – comparison design model with analysis model

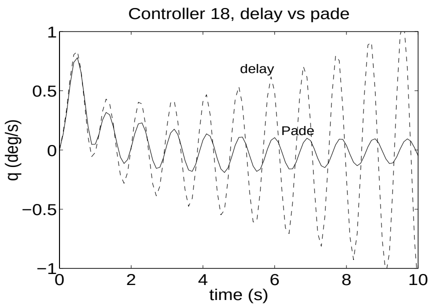


Fig. 20. EA-18 – comparison Padé filter with pure time delay