Structured Robust Control Against Mixed Uncertainty

Raquel Stella da Silva de Aguiar, Pierre Apkarian, and Dominikus Noll

***Abstract*—We present new approaches to designing structured controllers which are robust in the presence of mixed real parametric and complex dynamic uncertainty. As the synthesis of such controllers is inherently NP-hard, we discuss inner and outer relaxation techniques which make this problem amenable to computations. Our relaxation methods are positively evaluated and compared on the basis of a rich test set and for a challenging missile pilot design problem.**

***Index Terms*—Dynamic uncertainty, mixed uncertainty, nonsmooth optimization NP-hard problem, parametric uncertainty, structured controller, μ-synthesis**

I. Introduction

The design of feedback controllers which are robust in the presence of system uncertainty is a recurrent problem in control engineering, from which designers rarely escape due to the inevitable mismatch between a physical system and its mathematical model. It is generally agreed that one should account for the uncertainty already at the modeling stage, and in this work we follow this paradigm by addressing two types of uncertainty simultaneously: real uncertain parameters in the model equations and complex dynamic uncertainty.

The task of controlling a system with *mixed uncertainty* is further aggravated when the controller has to be *structured*. Structured control laws and control architectures are preferred by practitioners, but it appears that the more natural and easier-to-understand a desired control structure or architecture is, the harder it is to compute it. In robust mixed synthesis the difficulty can be highlighted as follows: it amounts to solving a nonlinear optimization problem globally, where a single evaluation of the cost function is already NP-hard.

The inherent difficulty of mixed uncertainty robust synthesis of structured controllers precludes naive direct approaches and makes the use of intelligent relaxation techniques mandatory. Here we distinguish between inner and outer relaxations of the control problem on a given set $\boldsymbol{\Delta}$ of uncertain scenarios. Outer approximations may relax the problem over $\boldsymbol{\Delta}$ by choosing a larger set $\widetilde{\boldsymbol{\Delta}} \supset \boldsymbol{\Delta}$ of scenarios on which computations are simplified, or may use a computable upper bound of the robust cost function on the constraint set $\boldsymbol{\Delta}$, or may even do both at the same time. The rationale is that as soon as a robust performance or stability certificate over $\widetilde{\boldsymbol{\Delta}}$ is obtained, this certificate automatically applies to the original set $\boldsymbol{\Delta}$. Similarly, any minimum of the computable upper bound on the set $\widetilde{\boldsymbol{\Delta}}$ will give an upper bound of the true underlying robust cost function on $\boldsymbol{\Delta}$. Typical outer relaxation methods include multiplier and scaling techniques, the various LMI-based relaxations like [1], [2], [3], [4], [5], and minimization of upper bounds of the structured singular value μ, the DGK-iteration of [6] being a prominent example. The main drawback of outer relaxations is that they may introduce conservatism, which increases with the complexity of the uncertainty. This is further aggravated by the fact that failures in computing a certificate over $\widetilde{\boldsymbol{\Delta}}$ may occur despite the simplified structure of $\widetilde{\boldsymbol{\Delta}}$.

In contrast, inner approximation techniques relax the problem over $\boldsymbol{\Delta}$ by choosing a simpler, typically finite, subset $\boldsymbol{\Delta}_a \subset \boldsymbol{\Delta}$ on which the robust cost function is computable and can be minimized. This avoids conservatism in the design, but has the disadvantage that no immediate certificates over $\boldsymbol{\Delta}$ are delivered. Inner relaxations may therefore require a post-processing step in which a robustness analysis technique is employed to obtain a certificate over $\boldsymbol{\Delta}$.

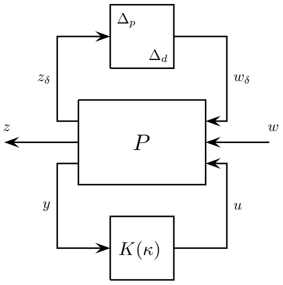

Fig. 1. Robust synthesis interconnection with two types of uncertainty and based on structured control laws $K(\kappa)$

Note that relying on a discrete set of scenarios to improve robustness is not a new idea and can be traced back to the work in [7], [8] with its multi-model approach. Alternatively, probabilistic approaches using randomized scenarios are considered in [9] and references therein.

In this work we compare two relaxation approaches to the robust synthesis problem shown schematically in Fig. 1. In section III we discuss a novel outer relaxation technique, which uses dynamic multipliers and a small gain argument to overestimate the cost function, keeping the uncertainty set $\boldsymbol{\Delta}$ fixed. This leads to a structured H_∞-synthesis problem, amenable to nonsmooth optimization techniques as made available through the functions SYSTUNE and HINFSTRUCT from [10], [11], [12]. In section IV we present an inner relaxation, in which a finite set $\boldsymbol{\Delta}_a$ of active scenarios is computed iteratively in

Ms. Aguiar is with ONERA, Department of Information Processing and Systems, Toulouse, France and Institut de Mathématiques de Toulouse, France.

M. Apkarian is with ONERA,Department of Information Processing and Systems, Toulouse, France.

M. Noll is with Institut de Mathématiques de Toulouse, France.

such a way that stability and performance for the scenarios $\Delta \in \mathbf{\Delta}_a$ assures robustness over the full range $\mathbf{\Delta}$. Finally, in section V, we discuss a hybrid approach, which applies the inner technique to the real uncertainty and the outer to the complex uncertainty. Numerical comparison between these techniques, based on a rich test bench, is presented in section VI, where the results for the classical routine DKSYN made available through [12] is also presented. In the end, a missile pilot design problem is solved with the techniques.

Notation

The terminology follows [13]. Given partitioned matrices

$$M := \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \text{ and } N := \begin{bmatrix} N_{11} & N_{12} \\ N_{21} & N_{22} \end{bmatrix}$$

of appropriate dimensions and assuming existence of inverses, the Redheffer star product [14], [15] of M and N is $M \star N :=$

$$\begin{bmatrix} M \star N_{11} & M_{12}(I - N_{11}M_{22})^{-1}N_{12} \\ N_{21}(I - M_{22}N_{11})^{-1}M_{21} & N \star M_{22} \end{bmatrix}.$$

When M or N do not have an explicit 2×2 structure, we assume consistently that the star product reduces to an LFT (Linear Fractional Transformation). The lower LFT of M and N is denoted $M \star N$ and defined as

$$M \star N := M_{11} + M_{12}N(I - M_{22}N)^{-1}M_{21},$$

and the upper LFT of M and N is denoted $N \star M$ and obtained as

$$N \star M := M_{22} + M_{21}N(I - M_{11}N)^{-1}M_{12}\,.$$

With these definitions, the $\star$ operator is associative.

The dependence of plant $P(s)$ and controller $K(s)$ on the Laplace variable s will be omitted for simplicity.

II. Problem Specification

We consider an LFT plant P with real parametric and dynamic complex uncertainties Δ, as shown in Fig. 1, and in feedback with a structured controller $K(\kappa)$:

$$P : \begin{bmatrix} z_\delta \\ z \\ y \end{bmatrix} = \begin{bmatrix} P_{\delta\delta} & P_{\delta w} & P_{\delta u} \\ P_{z\delta} & P_{zw} & P_{zu} \\ P_{y\delta} & P_{yw} & P_{yu} \end{bmatrix} \begin{bmatrix} w_\delta \\ w \\ u \end{bmatrix}, \tag{1}$$

with $w \in \mathbb{R}^{m_1}$ a vector of exogenous inputs, $z \in \mathbb{R}^{p_1}$ a vector of regulated outputs, $y \in \mathbb{R}^{p_2}$ the measured output, and $u \in \mathbb{R}^{m_2}$ the control input. The uncertainty channel is defined as

$$w_\delta = \Delta z_\delta, \tag{2}$$

where the uncertain matrix Δ is structured as

$$\Delta = \begin{bmatrix} \Delta_p & 0 \\ 0 & \Delta_d \end{bmatrix}, \tag{3}$$

with Δ_p representing real parametric, and Δ_d complex LTI (linear time-invariant) dynamic uncertainty. Without loss of generality, we assume that Δ_p and Δ_d have the following block-diagonal structure:

$$\Delta_p := \operatorname{diag}\left[\delta_1 I_{r_1}, \ldots, \delta_{N_p} I_{r_{N_p}}\right], \tag{4}$$

for real uncertain parameters $\delta_1, \ldots, \delta_{N_p} \in \mathbb{R}$ and their number of repetitions $r_1, \ldots, r_{N_p}$, and

$$\Delta_d := \operatorname{diag}\left[\Delta_1 \ldots, \Delta_{N_d}\right], \tag{5}$$

with $\Delta_i \in \mathbb{C}^{p_i \times q_i}$, $i = 1, \ldots, N_d$ for complex uncertainties. We also assume without loss of generality that the uncertainty is normalized so that Δ belongs to the H_∞-norm unit ball $\mathbf{\Delta} = \{\Delta : \overline{\sigma}(\Delta) \leq 1\}$, with $\Delta = 0$ representing nominal behavior and $\overline{\sigma}$ denoting the maximum singular value of a matrix. This means $\delta_i \in [-1, 1]$ for real parameters and $\overline{\sigma}(\Delta_i) \leq 1$ for complex blocks. For future use, we also introduce the H_∞-norm unit balls

$$\mathbf{\Delta}_p := \{\Delta_p : \overline{\sigma}(\Delta_p) \leq 1\},\ \mathbf{\Delta}_d := \{\Delta_d : \overline{\sigma}(\Delta_d) \leq 1\}\,. \tag{6}$$

The control channel $u \to y$ in (1) is put in feedback with a structured control law

$$u(s) = K(\kappa)y(s),$$

where according to [10] a controller

$$K(\kappa) : \begin{cases} \dot{x}_K &= A_K(\kappa)x_K + B_K(\kappa)y \\ u &= C_K(\kappa)x_K + D_K(\kappa)y \end{cases} \tag{7}$$

in state-space form is called *structured* if $A_K(\kappa), B_K(\kappa), \ldots,$ depend smoothly on a design parameter κ varying in a design space $\mathbb{R}^n$ or in some constrained subset of $\mathbb{R}^n$. Typical examples of structure include PIDs, reduced-order controllers, observer-based controllers, or control architectures combining various controller blocks such as set-point filters, feedforward, washout or notch filters, two degree of freedom controllers, and much else [16], [17]. In contrast, full-order controllers are state-space representations with the same order n_P as P without particular structure and are sometimes referred to as *unstructured*, or as *black-box controllers*.

Given the compact convex set of parametric and dynamic uncertainties $\mathbf{\Delta}$ in (3), including the nominal scenario $\Delta = 0$, the robust structured H_∞-control problem consists in computing a structured output-feedback controller $u = K(\kappa^*)y$ as in (7) with the following properties:

(i) **Robust stability**. The closed-loop system is well-posed and $K(\kappa^*)$ stabilizes $\Delta \star P$ internally for every $\Delta \in \mathbf{\Delta}$.

(ii) **Robust performance**. Given any other robustly stabilizing controller $K(\kappa)$ with the same structure, the optimal controller satisfies

$$\max_{\Delta \in \mathbf{\Delta}} \|T_{zw}(\Delta, \kappa^*)\|_\infty \leq \max_{\Delta \in \mathbf{\Delta}} \|T_{zw}(\Delta, \kappa)\|_\infty.$$

Here $T_{zw}(\Delta, \kappa) := \Delta \star P \star K(\kappa)$ is the closed-loop transfer function of the performance channel $w \to z$ of (1) when the control loop with $K(\kappa)$ and the uncertainty loop with Δ are both closed.

III. Outer relaxation

The synthesis problem (i) - (ii) above is of semi-infinite character and is in consequence not directly tractable. We therefore investigate how the problem can be relaxed into a simpler one, amenable to computations. In this section we consider an approach by outer relaxation.

With the real uncertain parameters $\Delta_p \in \mathbf{\Delta}_p$ in (4) and (6) we associate dynamic multipliers $\Phi \in \mathbf{\Phi}$ by defining

$$\mathbf{\Phi} := \big\{\Phi(s) = \operatorname{diag}(\phi_1(s), \dots, \phi_{N_p}(s)) : \\ \phi_i(s) \ \text{stable}, \ \|\phi_i(s)\|_\infty < 1\big\}, \tag{8}$$

where $\Phi(s)$ has the same block diagonal structure as the $\Delta_p \in \mathbf{\Delta}_p$ and therefore commutes with the $\Delta_p \in \mathbf{\Delta}_p$. We have the following

Proposition 1. *Given $\Phi \in \mathbf{\Phi}$ defined in (8), and $\Delta_p \in \mathbf{\Delta}_p$ defined in (4) and (6), let*

$$\Gamma(\Phi) := \begin{bmatrix} -\Phi & I+\Phi \\ I-\Phi & \Phi \end{bmatrix}.$$

Then the closed loop system $\Delta_p \star \Gamma(\Phi)$ is internally stable and satisfies the estimate

$$\|\Delta_p \star \Gamma(\Phi)\|_\infty \leq 1\,. \tag{9}$$

Proof. Since Δ_p and Φ commute, we get

$$\Delta_p \star \begin{bmatrix} -\Phi & I+\Phi \\ I-\Phi & \Phi \end{bmatrix} = (\Delta_p + \Phi)(I + \Phi\Delta_p)^{-1}, \tag{10}$$

the expression being well-defined due to $\|\Phi\|_\infty < 1$. Since Δ_p and Φ are structured conformably, we can verify internal stability and the estimate (9) in each block $\Delta_p = \delta I$ separately.

Now for $|\delta| \leq 1$, the first term $(\Delta_p + \Phi)$ in (10) is stable since Φ is stable. For the second term $(I + \Phi\Delta_p)^{-1}$, internal stability follows from the Small Gain Theorem [13] and the definition of $\mathbf{\Delta}_p$ and $\mathbf{\Phi}$.

To get the estimate (9), we can again consider a single block. For a fixed frequency ω we have

$$\overline{\sigma}\left((\delta I + \Phi(j\omega))(I + \delta\Phi(j\omega))^{-1}\right) \leq 1$$

if and only if

$$(\delta I + \Phi(j\omega))^H(\delta I + \Phi(j\omega)) \preceq (I + \delta\Phi(j\omega))^H(I + \delta\Phi(j\omega)),$$

and this is the same as

$$(\delta^2 - 1)(I - \Phi^H(j\omega)\Phi(j\omega)) \preceq 0,$$

where $\preceq 0$ means negative semi-definite. But now the result follows because $|\delta| \leq 1$ and $\|\Phi\|_\infty < 1$ together show that the last condition is satisfied. □

Note that proposition 1 bears some resemblance with the general quadratic-separator approach developed in [18] for well-posedness of uncertain systems. In their terminology, our multipliers $\Phi(s)$ in (8) are explicit candidates for characterizing mixed norm-bounded LTI uncertainties.

We now extend Proposition 1 to the case where both types of uncertainty are present. For simplicity we assume that complex blocks Δ_i are square, $p_i = q_i$. If need be, this can be achieved by squaring down the plant $P(s)$ in (1) with respect to the dynamic uncertainty Δ_d. Let us introduce the set $\mathbf{D}$ of D-scalings

$$\mathbf{D} := \Big\{D(s) := \operatorname{diag}(d_1(s)I_{p_i}, \dots, d_{N_d}(s)I_{p_{N_d}}) : \\ d_i(s), d_i(s)^{-1} \ \text{stable}\Big\}\,. \tag{11}$$

Note again that $D\Delta_d = \Delta_d D$ due to the block structure of Δ_d and D.

Proposition 1 is now extended to

Proposition 2. *Given $\Phi(s) \in \mathbf{\Phi}$, $D(s) \in \mathbf{D}$ and $\Delta_p \in \mathbf{\Delta}_p$, $\Delta_d \in \mathbf{\Delta}_d$ defined in (4), (5) and (6), let*

$$\Gamma(\Phi, D) := \begin{bmatrix} -\Phi & 0 & I+\Phi & 0 \\ 0 & 0 & 0 & D \\ I-\Phi & 0 & \Phi & 0 \\ 0 & D^{-1} & 0 & 0 \end{bmatrix}. \tag{12}$$

Then the closed loop system $\begin{bmatrix} \Delta_p & 0 \\ 0 & \Delta_d \end{bmatrix} \star \Gamma(\Phi, D)$ is internally stable and satisfies the estimate

$$\left\|\begin{bmatrix} \Delta_p & 0 \\ 0 & \Delta_d \end{bmatrix} \star \Gamma(\Phi, D)\right\|_\infty \leq 1\,. \tag{13}$$

Proof. The proof is analogous to the one in proposition 1 and is omitted for brevity. □

Proposition 2 provides an alternative characterization of uncertainty with mixed parametric and dynamic blocks, as we explain in the sequel. Note first that the Redheffer star product inverse of $\Gamma(\Phi, D)$ is obtained by swapping Φ and $-\Phi$ and D and D^{-1} in (12). This yields

$$\Gamma(\Phi, D)^{-\star} = \begin{bmatrix} \Phi & 0 & I-\Phi & 0 \\ 0 & 0 & 0 & D^{-1} \\ I+\Phi & 0 & -\Phi & 0 \\ 0 & D & 0 & 0 \end{bmatrix}.$$

This allows us now to answer the question of robust stability, where it suffices to consider the reduced plant P_r with the performance channel $w \to z$ removed:

$$P_r : \quad \begin{bmatrix} z_\delta \\ y \end{bmatrix} = \begin{bmatrix} P_{\delta\delta} & P_{\delta u} \\ P_{y\delta} & P_{yu} \end{bmatrix} \begin{bmatrix} w_\delta \\ u \end{bmatrix}. \tag{14}$$

Here the closed-loop interconnection $T_{zw}(\Delta, \kappa)$ of Fig. 1 reads $\Delta \star P_r \star K(\kappa)$. Inserting the term $\Gamma(\Phi, D)$ and its Redheffer inverse, this is the same as

$$\Delta \star \Gamma(\Phi, D) \star \Gamma(\Phi, D)^{-\star} \star P_r \star K(\kappa)\,.$$

Using associativity of $\star$, we split this suitably. Namely, by Proposition 2, the left-hand term $\Delta \star \Gamma(\Phi, D)$ is internally stable and belongs to the closed unit ball in the H_∞ metric. It follows that if we can find Φ, D and $K(\kappa)$ such that the right-hand term $\Gamma(\Phi, D)^{-\star} \star P_r \star K(\kappa)$ is stable and has H_∞ norm bounded by one, then by the Small Gain Theorem the closed-loop system $\Delta \star P_r \star K(\kappa)$ is robustly stable. We have proved the following

Theorem 1. *Suppose there exist $\Phi \in \mathbf{\Phi}$, $D \in \mathbf{D}$ and a structured controller $K(\kappa)$ such that the closed-loop system $\Gamma(\Phi, D)^{-\star} \star P_r \star K(\kappa)$ is internally stable and satisfies the estimate*

$$\left\|\Gamma(\Phi, D)^{-\star} \star P_r \star K(\kappa)\right\|_\infty < 1. \tag{15}$$

Then the closed-loop system $\Delta \star P_r \star K(\kappa)$ is robustly stable over $\mathbf{\Delta}$. □

The next step is to include robust H_∞ performance into the setup, which uses the Main Loop Theorem [13]. We bring back the performance channel and introduce the scaled plant P_γ as

$$P_\gamma : \quad \begin{bmatrix} z_\delta \\ z \\ y \end{bmatrix} = \begin{bmatrix} P_{\delta\delta} & P_{\delta w} & P_{\delta u} \\ \frac{1}{\gamma}P_{z\delta} & \frac{1}{\gamma}P_{zw} & \frac{1}{\gamma}P_{zu} \\ P_{y\delta} & P_{yw} & P_{yu} \end{bmatrix} \begin{bmatrix} w_\delta \\ w \\ u \end{bmatrix}. \tag{16}$$

Then with the same notations as above and the scaling set $\mathbf{D}$ suitably generalized to account for the new performance block, we have

Corollary 1. *Suppose there exist $\Phi \in \mathbf{\Phi}$, $D \in \mathbf{D}$, and a structured controller $K(\kappa)$ such that $\Gamma(\Phi, D)^{-\star} \star P_\gamma \star K(\kappa)$ is internally stable and satisfies the estimate*

$$\left\| \Gamma(\Phi, D)^{-\star} \star P_\gamma \star K(\kappa) \right\|_\infty < 1. \tag{17}$$

Then the closed-loop system $\Delta \star P \star K(\kappa)$ is robustly stable over $\mathbf{\Delta}$, and has worst-case H_∞ performance γ over $\mathbf{\Delta}$. □

The reader is referred to Fig. 2 for a schematic view. What we have derived is a novel outer relaxation of the μ synthesis problem in the form of a 2-disk H_∞-synthesis problem. We refer to [17] for the algorithmic approach to multi-disk synthesis where all H_∞ constraints are handled simultaneously. As constraints we have internal stability, the performance estimate (17), and $\|\Phi\|_\infty < 1$. With a small tolerance $\eta > 0$, this may be turned into the following optimization program:

$$\begin{array}{ll} \text{minimize} & \gamma \\ \text{subject to} & \left\| \Gamma(\Phi, D)^{-\star} \star P_\gamma \star K(\kappa) \right\|_\infty \leq 1 - \eta \\ & \Gamma(\Phi, D)^{-\star} \star P_\gamma \star K(\kappa) \text{ internally stable} \\ & \|\Phi\|_\infty \leq 1 - \eta \\ & \Phi \in \mathbf{\Phi}, D \in \mathbf{D}, \kappa \in \mathbb{R}^n, \gamma \in \mathbb{R}_+. \end{array} \tag{18}$$

Nonsmooth solvers such as `HINFSTRUCT` or `SYSTUNE`, available through [11], [12], can be used to compute locally optimal controllers for (18).

A major obstacle apparent in all known outer relaxation methods lies in the phenomenon of repetitions of uncertain parameters δ_i in (4). Large numbers of repetitions r_i quickly lead to challenging numerical problems, since the number of variables in the Φ_i's, hence in (18), increases as $\mathcal{O}(r_i^2)$. In contrast, no such drastic increase in the number of variables arises from the complex uncertainty, as the scalings only contribute moderately, and no phenomenon analogous to the high number of repetitions occurs.

This suggests the use of an alternative strategy to overcome the difficulty of large r_i. One possible line of attack is to switch to an inner relaxation, which we discuss in the following section. Yet another line is to treat real parametric and complex dynamic uncertainty individually. This leads to a hybrid approach, which we discuss in section V.

Let us point to a difference between the outer relaxation (18), and LMI-based relaxations as for instance [1], [3]. In (18) we do not fully convexify the problem, which is beneficial in so far as less conservatism is introduced. LMI-relaxations not only may introduce conservatism, they may also be difficult to solve numerically due to the presence of Lyapunov and multiplier variables. It is fair to say that these methods are not appropriate if one aims at practical applications.

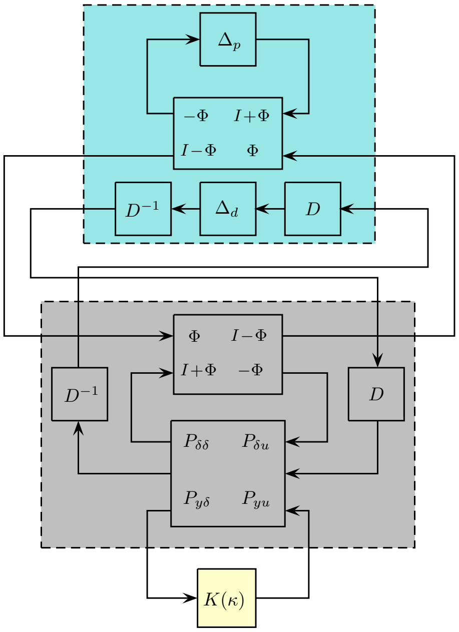

Fig. 2. Illustration of Theorem 1. Fictive new plant $\Gamma(\Phi, D)^{-\star} \star P_r$ shown in gray is in upper feedback with new mixed uncertainty $\Delta \star \Gamma(\Phi, D)$ shown in greenish, and in lower feedback with structured controller $K(\kappa)$. For Corollary 1 the scaled plant P_γ is used.

IV. Inner relaxation

In this section we discuss an inner relaxation technique, where the infinite set of scenarios $\mathbf{\Delta}$ is approximated by a suitably chosen finite subset $\mathbf{\Delta}_a$, which we call the set of *active scenarios*. Once this set is specified, this leads to an optimization program of the form

$$\min_{\kappa} \max_{\Delta \in \mathbf{\Delta}_a} \|\Delta \star P \star K(\kappa)\|_\infty , \tag{19}$$

which due to the finiteness of $\mathbf{\Delta}_a$ is a multi-disk H_∞-synthesis problem in the sense of [17]. The rationale is that, once the worst-case scenarios $\Delta \in \mathbf{\Delta}_a$ are controlled simultaneously, the *locally* optimal controller $K(\kappa^*)$ computed in (19) assures robust stability and performance not only over the set $\mathbf{\Delta}_a$, but hopefully over the full scenario set $\mathbf{\Delta}$.

This seems appealing since program (19) can be solved to local optimality with tools like `SYSTUNE` or `HINFSTRUCT` from [11], [12]. However, there is a disclaimer. The approach quickly succumbs for exceedingly large sets $\mathbf{\Delta}_a$, and it is therefore mandatory to build $\mathbf{\Delta}_a$ diligently. The way we select these active scenarios $\Delta \in \mathbf{\Delta}_a$ is shown in algorithm 1.

In the sequel we discuss the individual steps of this scheme, which can also be seen graphically in Fig. 3. To begin with, note that the solution of program α^* in step 3 and program h^* in step 4 is discussed in detail in references [16], [19], [20], [21]. (In step 3 $A(\Delta, \kappa)$ denotes the A-matrix of the closed

Algorithm 1. Robust synthesis by inner approximation

Parameters: $\varepsilon > 0$.

- ▷ **Step 1** (`Nominal synthesis`)**.** Initialize the set of active scenarios as $\mathbf{\Delta}_a = \{0\}$.
- ▷ **Step 2** (`Multi-model synthesis`)**.** Given the current finite set $\mathbf{\Delta}_a \subset \mathbf{\Delta}$ of active scenarios, compute a structured multi-model H_∞ controller by solving the multi-disk H_∞-program

  $$h_* = \min_{\kappa\in\mathbb{R}^n} \max_{\Delta\in\mathbf{\Delta}_a} \|\Delta \star P \star K(\kappa)\|_\infty .$$

  The solution is the structured controller $K(\kappa^*)$.
- ▷ **Step 3** (`Worst-case stability`)**.** Try to destabilize the closed-loop system $\Delta \star P_r \star K(\kappa^*)$ by computing its worst case spectral abscissa

  $$\alpha^* = \max_{\Delta\in\mathbf{\Delta}} \alpha\left(A(\Delta,\kappa^*)\right).$$

  The solution is the worst stability scenario Δ^*. If $\alpha^* = \alpha\left(A(\Delta^*,\kappa^*)\right) \geq 0$, then include Δ^* in the set $\mathbf{\Delta}_a$ and go back to step 2. Otherwise continue with step 4.
- ▷ **Step 4** (`Worst-case performance`)**.** Try to degrade performance of the closed-loop system $\Delta \star P \star K(\kappa^*)$ by computing its worst case H_∞ norm

  $$h^* = \max_{\Delta\in\mathbf{\Delta}} \|\Delta \star P \star K(\kappa^*)\|_\infty .$$

  The solution is the worst performance scenario $\Delta^\sharp$.
- ▷ **Step 5** (`Stopping test`)**.** If $\alpha(A(\Delta^\sharp,\kappa^*)) < 0$ and $h^* < (1+\varepsilon)h_*$, degradation of performance is marginal. Then exit loop and goto step 6 for posterior certification. Otherwise include $\Delta^\sharp$ in the set $\mathbf{\Delta}_a$ and continue loop with step 2.
- ▷ **Step 6** (`Certification`)**.** Use μ-analysis tools to certify *a posteriori* that $K(\kappa^*)$ is robustly stable over $\mathbf{\Delta}$ and has robust H_∞ performance h_* over $\mathbf{\Delta}$.

loop system (14) with the control loop (7) closed by $K(\kappa)$, and the uncertain loop (2) closed by Δ).

All programs occurring in the scheme are nonsmooth and have to be addressed by bundle or bundle trust-region methods. The main difference between α^*, h^* on the one hand, and h_* on the other, is analyzed in [16], [19].

For the general understanding we stress that, in this work, all programs h_*, h^*, α^* are addressed by local solvers, so that only local optimality certificates are obtained. In particular, even when the local loop exits with the successful flag *acceptable performance*, this is by no means a global certificate. This is why posterior certification by a global method, like μ analysis, in the post-processing step is needed. Experiments with branch-and-bound and other global methods are reported in [21].

It is generally agreed that inner relaxations outperform the conservative outer relaxations in practice. Notwithstanding, it is often held against them that they are heuristic and do not guarantee certificates over $\mathbf{\Delta}$ unless followed by post-certification. In contrast, so it is argued, outer relaxations provide such certificates directly.

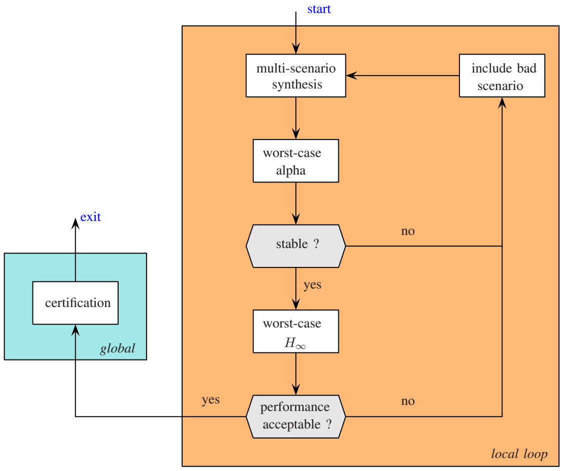

Fig. 3. Generating active scenarios $\Delta^*, \Delta^\sharp \in \mathbf{\Delta}_a$ by worst-case stability and worst-case H_∞ optimization. Candidate controllers $K(\kappa^*)$ are computed by multi-disk H_∞ optimization, and certificates are obtained *a posteriori* using global methods.

Closer inspections reveals this reasoning as superficial, as we now explain. Namely, there is no reason why outer relaxations in turn should give any guarantee of success. Relaxing the problem posed on $\mathbf{\Delta}$ on a larger and easier to handle set $\widetilde{\mathbf{\Delta}} \supset \mathbf{\Delta}$ does not mean that success on $\widetilde{\mathbf{\Delta}}$ is in any sense *guaranteed*. And using an upper bound of the true objective on $\mathbf{\Delta}$ does not mean that minimization of this upper bound will succeed and *guarantee* a result. This is even the case for the notorious convex relaxations of the robust synthesis problem, because even when the problem is convexified to an LMI, there is no *guarantee* that this LMI will be feasible.

Once it is agreed that in this sense neither inner nor outer relaxations can guarantee success, both approaches are at equal rights, the competition is open, and the better will win. It turns out that this is in all cases the inner relaxation technique, as it avoids conservatism in the synthesis phase. The fact that conservative analysis tool are used in the final certification phase does not change this picture. The conclusion is that it is not a good idea to introduce conservatism at an early stage, e.g. by including it in the synthesis step. Instead, delaying the use of conservative techniques to the very end, and using them in robustness and performance analysis *only*, has the better end.

It is possible to split the search for scenarios $\Delta \in \mathbf{\Delta}_a$ with bad H_∞ performance into two consecutive steps, where scenarios bad for $\mathbf{\Delta}_p$, and scenarios bad for $\mathbf{\Delta}_d$, are generated separately. The idea to proceed in this way springs from the observation that the function `WCGAIN` of [12] works particularly well in the case of sole complex uncertainty $\mathbf{\Delta}_d$, as observed in [22], but is less precise in the case of mixed uncertainty or sole real uncertainty. One could therefore split step 4 of algorithm 1 into halves. In the first half-step, for fixed κ^* and for a fixed complex uncertainty Δ_d^* (computed in the previous sweep), a search over $\mathbf{\Delta}_p$ for a worst case Δ_p^* is made based on an optimization program as analyzed in [19].

In the second half-step this scenario Δ_p^*, along with κ^*, are held fixed, and a search over $\mathbf{\Delta}_d$ for a worst-case Δ_d^* based on `WCGAIN` is made. This option was also evaluated in our experiments.

V. Hybrid relaxation approach

As we had observed in section III, strong conservatism in the outer relaxation (18) occurs, particularly if parametric uncertainty $\mathbf{\Delta}_p$ with a large number r_i of repetitions is present, due to substantially increase in the number of unknowns $\Phi \in \mathbf{\Phi}$. Even though this suggests the use of inner relaxations, we have to be aware that complex uncertainty $\mathbf{\Delta}_d$ also comes with some encumbrance. Namely, it makes the computation of bad scenarios $\Delta^*, \Delta^\sharp$ in steps 3 and 4 of algorithm 1 harder than in the case of pure real parametric uncertainty discussed in [16]. Computations of α^* and h^* then turn out more challenging, but more seriously, the number of active scenarios $\Delta \in \mathbf{\Delta}_a$ needed to cover the set $\mathbf{\Delta}$ may be significantly larger than in the case of pure $\mathbf{\Delta}_p$. This may become a major challenge in the computation of h_* in step 2 of algorithm 1. For short, the situation is not as straightforward as on first glance. This rises the question whether the two types of uncertainty could not be handled individually. A natural idea is to use inner relaxation for $\mathbf{\Delta}_p$, and stick to a multiplier approach for $\mathbf{\Delta}_d$. This is what we have termed the *hybrid approach.* We start with the following result, the notations being those of section III.

Theorem 2. *Suppose there exist $D \in \mathbf{D}$ and a structured controller $K(\kappa)$ such that the closed loop system $\Delta_p \star \Gamma(0, D)^{-\star} \star P_\gamma \star K(\kappa)$ is internally stable and satisfies the estimate*

$$\left\|\Delta_p \star \Gamma(0, D)^{-\star} \star P_\gamma \star K(\kappa)\right\|_\infty < 1 \qquad (20)$$

for all $\Delta_p \in \mathbf{\Delta}_p$. Then the closed-loop system $\Delta \star P \star K(\kappa)$ is robustly stable and has worst-case H_∞ performance γ over $\mathbf{\Delta}$.

Proof. It suffices to note that

$$\Delta_p \star \Gamma(0, D)^{-\star} \star P_\gamma \star K(\kappa) = \begin{bmatrix} D & 0 \\ 0 & I \end{bmatrix} (\Delta_p \star P_\gamma \star K(\kappa)) \begin{bmatrix} D^{-1} & 0 \\ 0 & I \end{bmatrix}.$$

It then follows that internal stability of $\Delta_p \star \Gamma(0, D)^{-\star} \star P_\gamma \star K(\kappa)$ in tandem with (20) is a complex μ upper bound condition for stability and performance in the sense of [23]. Therefore, for any $\Delta_p \in \mathbf{\Delta}_p$, the system $\Delta_p \star P_\gamma \star K(\kappa)$ is stable and has worst-case H_∞ performance γ for all $\Delta_d \in \mathbf{\Delta}_d$, which concludes the proof. □

This result offers yet another algorithmic option, which will be tested and compared to the other approaches in our experiments under the name *hybrid* approach.

In the above approach we have chosen $D \in \mathbf{D}$ as independent of $\Delta_p \in \mathbf{\Delta}_{p,a}$. Choosing an individual $D_p = D(\Delta_p)$ for each performance constraint indexed by $\Delta_p \in \mathbf{\Delta}_{p,a}$ would reduce conservatism, but at the same time increase the number of variables in the synthesis program of step 2.

Algorithm 2. Robust synthesis by the hybrid method

```
Parameters: $\varepsilon > 0$, $\eta > 0$.
▷ Step 1 (Nominal synthesis). Initialize set of active real scenarios as $\mathbf{\Delta}_{p,a} = \{0\}$.
▷ Step 2 (Multi-model synthesis). Given current finite set $\mathbf{\Delta}_{p,a} \subset \mathbf{\Delta}_p$ of active real scenarios, compute structured multi-model $H_\infty$ controller by solving the $H_\infty$-program
      minimize $\gamma$
      subject to $\|\Delta_p \star \Gamma(0, D)^{-\star} \star P_\gamma \star K(\kappa)\|_\infty \leq 1 - \eta$
                 for all $\Delta_p \in \mathbf{\Delta}_{p,a}$
                 $D \in \mathbf{D}, \kappa \in \mathbb{R}^n, \gamma \in \mathbb{R}_+$
    The locally optimal solution $(\gamma_*, D^*, \kappa^*)$ gives rise to the structured multi-scenario $H_\infty$ controller $K(\kappa^*)$.
▷ Step 3 (Worst-case stability). Try to destabilize the closed-loop system $\Delta_p \star \Gamma(0, D^*)^{-\star} \star P_{\gamma_*} \star K(\kappa^*)$ by computing its worst case spectral abscissa
      $\alpha^* = \max_{\Delta_p \in \mathbf{\Delta}_p} \alpha\left(A(\Delta_p, \kappa^*)\right).$
    The solution is the worst stability scenario $\Delta_p^*$. If $\alpha^* = \alpha\left(A(\Delta_p^*, \kappa^*)\right) \geq 0$, then include $\Delta_p^*$ in the set $\mathbf{\Delta}_{p,a}$ and go back to step 2. Otherwise continue with step 4.
▷ Step 4 (Worst-case performance). Try to degrade performance of the closed-loop system $\Delta_p \star \Gamma(0, D^*)^{-\star} \star P_{\gamma_*} \star K(\kappa^*)$ by computing its worst case $H_\infty$ norm
      $\gamma^* = \max_{\Delta_p \in \mathbf{\Delta}_p} \|\Delta_p \star \Gamma(0, D^*)^{-\star} \star P_{\gamma_*} \star K(\kappa^*)\|_\infty.$
    The solution is the worst performance scenario $\Delta_p^\sharp$.
▷ Step 5 (Stopping test). If $\alpha(A(\Delta_p^\sharp, \kappa^*)) < 0$ and $\gamma^* < (1 + \varepsilon)\gamma_*$, degradation of performance is marginal. Then exit loop and goto step 6 for posterior certification. Otherwise include $\Delta_p^\sharp$ in the set $\mathbf{\Delta}_{p,a}$ and continue loop with step 2.
▷ Step 6 (Certification). Use $\mu$-analysis tools from [12] to certify a posteriori that $K(\kappa^*)$ is robustly stable over $\mathbf{\Delta}$ and has robust $H_\infty$ performance $\gamma_*$ over $\mathbf{\Delta}$.
```

VI. Test Cases

The efficiency of the three approaches from sections III, IV and V was compared on the basis of a test bench consisting of thirty systems adapted from the literature. Evaluation was based on the best worst-case H_∞ performance, or gain, achieved by each method.

Table I shows that test cases 1-7 have only complex dynamic uncertainty, four test cases, 8-11, have pure real parametric uncertainty, whereas the remaining nineteen cases 12-30 have mixed uncertainty. The column labeled 'Δ-structure'in Table I shows the structure of the uncertainty Δ. Positive numbers give the size of square complex blocks Δ_d of dynamic uncertainties in (5), negative numbers represent a real parametric uncertainty and its repetition in (4). For instance, case 9 has $N_d = 0$, $N_p = 2$ with $r_1 = 18$ and $r_2 = 2$. Case 23 has

TABLE I
TEST CASES

No.	Δ-structure		n_P	z-w	y-u
1	complex	1	9	3 3	2 1
2	complex	3	7	1 2	1 1
3	complex	3	8	4 4	3 1
4	complex	8	12	6 2	2 2
5	complex	1,1	22	2 2	2 2
6	complex	1	3	1 1	1 1
7	complex	2	26	6 5	5 2
8	real	-1	3	2 3	1 1
9	real	-18,-2	23	3 2	3 1
10	real	-20	10	2 1	1 1
11	real	-21	5	2 2	1 1
12	mixed	1,-1,-1	9	1 1	1 1
13	mixed	1,-2,-2	7	4 2	1 1
14	mixed	1,-1,-3	8	3 4	2 1
15	mixed	1,-1	8	2 4	2 2
16	mixed	2,-1,-1,-1,-1,-1,-1	14	6 2	2 2
17	mixed	1,-1,-1,-1,-1,-3	9	2 1	1 1
18	mixed	1,-1,-2,-2	6	2 1	1 1
19	mixed	1,-1,-1,-3,-3,-3	6	2 2	1 1
20	mixed	1,-1,-1,-1,-1,-1,-3,-3,-3,-3,-3	11	2 1	1 1
21	mixed	4,-1,-1,-1,-1	8	6 2	2 2
22	mixed	1,-1,-1,-1,-1,-1,-1,-1,-2,-2,-2,-2	19	2 3	1 1
23	mixed	3,-1,-1,-6	8	4 4	3 1
24	mixed	3,-1	7	1 2	1 1
25	mixed	1,-1,-1,-1,-6,-6,-6	24	3 2	3 1
26	mixed	1,1,1,1,-1,-1,-1,-1	8	6 2	2 2
27	mixed	1,-1	7	2 2	1 1
28	mixed	1,-1,-5	7	2 3	2 1
29	mixed	1,-1	4	2 3	1 1
30	mixed	1,-1	8	2 2	1 1

TABLE II
COMPARISON OF OPTIMIZATION-BASED RELAXATION TECHNIQUES

No.	n_K	*inner*			*hybrid*			*outer*	
		gain	certified	$\lvert\Delta_a\rvert$	gain	certified	$\lvert\Delta_{p,a}\rvert$	gain	certified
1	2	1.003	1.003	10	1.008	0.999	2	0.989	0.996
2	1	1.000	1.000	6	1.001	0.999	2	0.991	0.999
3	4	0.977	0.978	28	1.444	1.521	2	1.488	1.493
4	3	0.999	1.000	2	1.006	0.999	2	0.991	0.999
5	5	0.989	0.991	23	1.182	1.209	2	1.601	1.555
6	1	1.000	1.000	3	1.008	1.003	2	0.991	1.000
7	4	1.027	1.026	26	1.045	1.043	2	1.035	1.043
8	2	1.000	1.000	3	1.000	1.000	3	1.116	1.126
9	2	0.999	0.998	5	1.001	1.001	5	–	86.26
10	2	1.000	1.000	2	1.000	1.000	2	–	4.687
11	1	1.000	1.000	1	1.000	1.000	2	–	1.000
12	3	1.000	0.998	10	1.052	1.052	3	1.217	1.227
13	1	1.000	0.993	2	1.002	0.994	2	18.83	1.135
14	2	1.000	1.000	3	1.000	1.000	3	1.005	1.000
15	3	1.000	1.000	8	1.001	1.000	3	9.999	6.373
16	4	1.178	1.230	16	1.238	1.225	14	2.227	1.732
17	4	0.906	–	13	1.155	1.143	3	6.449	2.475
18	3	0.992	0.992	6	1.093	1.090	5	1817	1785
19	1	1.000	1.000	1	1.000	1.000	2	10.190	1.692
20	2	1.210	1.210	5	1.223	1.222	3	20.00	18.52
21	1	1.000	1.005	2	1.005	1.000	2	0.990	1.000
22	3	0.976	0.976	10	1.042	1.041	4	–	36339
23	4	1.070	1.079	37	4.238	4.168	2	10.180	7.881
24	4	0.997	0.997	8	0.997	0.994	3	1.813	1.762
25	3	1.000	0.999	8	1.103	1.103	4	–	60.338
26	1	1.000	1.000	2	1.007	0.999	2	0.990	0.999
27	3	0.998	0.997	7	1.000	0.999	5	1.598	1.589
28	4	1.001	1.001	5	1.000	0.999	6	10.02	3.027
29	2	1.020	1.020	5	1.011	1.019	3	1.228	1.188
30	5	1.088	1.085	14	1.071	1.057	3	6.935	6.721

$N_d = 1$, with $p_1 = q_1 = 3$, and $N_p = 3$, with $r_1 = r_2 = 1$ and $r_3 = 6$. Column n_P is the order of the generalized plant P, column z-w shows the number of exogenous outputs and inputs and column y-u shows the number of control outputs and inputs.

In all plants, the performance channel $w \to z$ was scaled so that the worst case performance of the closed-loop system computed by the inner approach in algorithm 1 was close to the value one. This re-scaling renders the comparison between the different techniques more straightforward. All performance values were certified by the μ-analysis-based routine WCGAIN from [12]. The only exception is case 17, where WCGAIN failed, indicated by '–'in column 4 of Table II. Note that test cases are available at http://rss-aguiar.site88.net/ along with more detailed information.

A. Comparison between the relaxation techniques

In Table II the results of inner relaxation from algorithm 1, hybrid approach from algorithm 2, and outer relaxation (18) are compared. These methods are labeled *inner*, *hybrid* and *outer*. Column n_K gives the order of the synthesized controller, which is the same for the 3 approaches.

Columns 3-5 give the results of the inner relaxation, algorithm 1. Column 3, named 'gain', corresponds to the value h_* found on exit of the local loop (see the scheme in Fig. 3). The number of times that algorithm 1 executed the local loop, equivalent to the number scenarios $|\mathbf{\Delta}_a|$, is given in column 5. The global certification by WCGAIN is given in column 4, labeled 'certified'. For instance, in case study 1, algorithm 1 found the value $h_* = 1.003$ for a controller of order $n_K = 2$ and required $|\mathbf{\Delta}_a| = 10$ scenarios. Certification with WCGAIN confirmed this value as correct.

Similarly, columns 6-8 of Table II show the results for the *hybrid* method. The value γ_* found on exit of the local loop (see Fig. 3) is given in column 6, labeled 'gain'. The number $|\mathbf{\Delta}_{p,a}|$ of sweeps made by the local loop, that is also the number of scenarios, is presented in column 8. Column 7 shows what WCGAIN certified when given this controller on input. For instance, in case study 1, algorithm 2 estimated the robust gain as $\gamma_* = 1.008$, and needed only two scenarios to achieve this, and in the end WCGAIN showed that the controller was even slightly better, and certified a robust gain of 0.998.

Finally, columns 9-10 of Table II correspond to the results of the *outer* relaxation. The estimated gain value is given in column 9, and what WCGAIN obtained is in column 10. Note that *outer* failed to satisfy the constraints $\|\Gamma(\Phi, D)^{-\star} \star P_\gamma \star K(\kappa)\|_\infty < 1$ and $\|\Phi\|_\infty < 1$ simultaneously in cases 9, 10, 11, 22 and 25. This means the iterate $(\gamma^\sharp, \Phi^\sharp, D^\sharp, \kappa^\sharp)$, where optimization of (18) stopped, was not a local minimum of (18), indicated by the failure sign '—'in column 'gain'. Even though, the $K(\kappa^\sharp)$ were used for certification.

Closer inspection of the results reveals the following details. Within an error margin of 1%, WCGAIN certified the gain value h_* obtained by *inner* in all cases, except test case 16 where *inner* is 4.4% below the certified value. This means *inner* was never conservative, but was optimistic in one case. Note that WCGAIN failed to certify the gain value in study 17. This represents the sole case where we have observed failure of WCGAIN.

The values returned by *hybrid* were certified by WCGAIN, for a 1% error margin, in 26 out of 30 cases. In two studies *hybrid* was slightly optimistic, providing values below the certification of WCGAIN. This concerned study 3 with 5.33% and study 5 with 2.28%. In studies 23 and 30 hybrid provided a slightly conservative value of 1.6% and 1.3%, respectively, above WCGAIN value. For the same error tolerance, *outer* and

TABLE III
COMPARISON OF DGK-ITERATION WITH INNER RELAXATION

No.	*dksyn*			*inner*		No.	*dksyn*			*inner*	
	γ_{dk}	n_K^{dk}	$q\boldsymbol{\Delta}$	n_K	γ_{inner}		γ_{dk}	n_K^{dk}	$q\boldsymbol{\Delta}$	n_K	γ_{inner}
1	0.994	13	1.006	2	1.003	16	1.209	96	0.827	4	1.230
2	1.011	19	0.989	1	1.000	17	1.489	117	0.731	4	0.912
3	1.046	38	0.956	4	0.978	18	4.507	6	0.222	3	0.992
4	0.983	20	1.002	3	1.008	19	3.765	6	0.266	1	1.000
5	1.229	22	0.814	5	0.991	20	3.746	11	0.267	2	1.210
6	0.934	7	1.071	1	1.000	21	0.998	20	1.002	1	1.005
7	0.965	42	1.037	4	1.026	22	25118	19	0.000	3	0.976
8	1.140	11	0.877	2	1.000	23	2.258	304	0.285	4	1.079
9	2.704	253	0.370	2	0.998	24	1.080	34	0.926	4	0.997
10	2.452	10	0.408	2	1.000	25	–	–	–	3	0.999
11	1.181	5	0.847	1	1.000	26	0.936	16	1.068	1	1.000
12	1.180	33	0.847	3	0.998	27	1.714	27	0.583	3	0.997
13	1.014	93	0.986	1	0.993	28	244.7	7	0.004	4	1.001
14	1.002	8	0.998	2	1.000	29	1.257	16	0.796	2	1.020
15	1.009	28	0.991	3	1.000	30	0.982	30	1.018	5	1.085

WCGAIN agreed in 36.7% of the cases, while in the remaining 63.3% *outer* had a slight tendency to be conservative.

Within the 1% error margin *inner* computed a smaller gain value than *hybrid* in 10 out of 30 studies and their values agreed in the remaining 20 cases. The gain values achieved by *outer* and *inner* agreed in 26.7% of the cases, in the remaining cases *outer* is conservative; *outer* and *hybrid* agreed in 33.3% of the cases.

All computations were performed on MATLAB® R2015b running in Ubuntu 12.04, Intel Core 2 DUO E6850 @ 3.00GHz and 3.8 GB RAM.

We observed that splitting the generation of bad scenarios $\Delta^\sharp \in \boldsymbol{\Delta}$ into two half steps, where one half step generates bad scenarios for parametric uncertainty and the other generates bad scenarios for complex uncertainty, is feasible. However it is not an improvement over the method proposed in algorithms 1 and 2. Therefore this line was not further explored in the present testing.

B. Comparison with DKSYN

In Table III we compare algorithm 1 to the standard DKSYN synthesis tool of [12]. This method is originally based on properties of the structured singular value and DGK-iteration [6], [23], [24].

We run DKSYN for the test cases of Table I with $\boldsymbol{\Delta}$ as input, where it returns an upper bound for the structured singular value μ, achieved with a controller K_{dk} of order n_K^{dk}. This μ value represents simultaneously the robust gain γ_{dk}, and the factor $q = 1/\gamma_{\mathrm{dk}}$ for the scaled box $q\boldsymbol{\Delta}$, on which this performance and stability are certified. Columns 4 and 10 display the factor q and the results are shown in columns 2-4 and 8-10 of Table III. Columns 5-6 and 11-12 of Table III are the results for *inner* from Table II, repeated for convenience.

For instance, in study 1 DKSYN achieved a robust gain of $\gamma_{\mathrm{dk}} = 0.994$ on the ball $q\boldsymbol{\Delta}$ with $q = 1.006$, using a controller of order $n_K^{\mathrm{dk}} = 13$, as algorithm 1 achieved the robust gain $h_* = 1.003$ on the original ball $\boldsymbol{\Delta}$, with a controller of order $n_K = 2$.

Even though a direct comparison is difficult due to the fact that DKSYN modifies the given ball $\boldsymbol{\Delta}$ and the controller order, we can see that DKSYN performs better than the inner approximation in cases with $q > 1$. One can observe by comparing columns n_K and n_K^{dk} that the price for this improvement is generally a much higher controller order $n_K \ll n_K^{\mathrm{dk}}$. On the other hand, in those cases where $q < 1$, DKSYN was not able to find a stabilizing controller on the original box $\boldsymbol{\Delta}$ despite the higher controller order, and had to reduce the box to the smaller size $q\boldsymbol{\Delta}$ to get a certified result.

C. Comparison of all four methods

In order to allow an even better comparison between DKSYN and the optimization-based approaches, we proceeded as follows with the experiment. We accepted the ball $q\boldsymbol{\Delta}$ found by DKSYN as the new uncertainty ball, and agreed to compare all four methods on this ball. This means the methods *inner*, *outer*, and *hybrid* were re-run on $q\boldsymbol{\Delta}$.

Note that in the cases where DKSYN returns $q > 1$, this requires even harder work from the optimization based method. This extra work should result in an even higher gain estimate, leaving DKSYN further in the lead. On the other hand, for $q < 1$ the work for the optimization based methods is made easier, so here it is expected that they return even better results, gaining further on DKSYN.

TABLE IV
COMPARISON OF THE FOUR METHODS

No.	*inner*		*hybrid*		*outer*		*dksyn*	
	gain	n_K	gain	n_K	gain	n_K	gain	n_K
1	0.982	4	0.983	3	0.994	3	0.994	13
2	0.999	1	1.000	1	1.011	1	1.011	19
3	0.949	4	1.085	8	1.101	10	1.046	38
4	1.009	4	1.010	4	1.010	4	0.983	20
5	0.856	5	1.107	5	1.010	5	1.229	22
6	0.914	2	0.923	2	0.914	2	0.934	7
7	0.965	9	0.984	13	0.977	9	0.965	42
8	0.923	2	0.923	2	1.061	2	1.140	11
9	0.896	2	0.895	2	79.65	15	2.704	253
10	0.972	2	0.972	2	2.168	10	2.452	10
11	0.999	1	1.000	1	1.181	5	1.181	5
12	0.472	3	0.480	3	0.550	3	1.180	33
13	0.990	1	0.991	1	0.995	1	1.014	93
14	0.999	2	1.000	2	1.000	2	1.002	8
15	0.983	7	1.119	11	4.559	6	1.009	28
16	1.106	4	1.111	4	1.277	11	1.209	96
17	0.617	4	0.710	4	0.969	4	1.489	117
18	0.203	3	0.207	3	4.193	5	4.507	6
19	1.000	1	1.000	1	1.418	1	3.765	6
20	0.964	2	0.964	2	3.167	5	3.746	11
21	1.000	1	1.000	1	1.000	1	0.998	20
22	0.439	3	0.439	3	9777	2	25118	19
23	0.622	4	0.709	4	2.258	4	2.258	304
24	0.904	4	0.902	4	1.080	4	1.080	34
25	1.000	3	1.103	3	200*	3	x	x
26	1.051	1	1.051	1	1.051	1	0.936	16
27	0.801	3	0.801	3	1.172	3	1.714	27
28	0.006	4	0.006	4	0.006	4	244.686	7
29	0.847	2	0.862	2	0.963	2	1.257	16
30	1.080	5	1.159	5	6.445	6	0.982	30

In addition to changing the ball to $q\boldsymbol{\Delta}$, in the cases where $q > 1$ and $n_K < n_K^{\mathrm{dk}}$, we allow the optimization-based methods to increase n_K to n_K^+, keeping $n_K^+ < n_K^{\mathrm{dk}}$. This is for fairness, as $n_K < n_K^{\mathrm{dk}}$ represents a huge advantage for DKSYN. This increase in the order allows the optimization-based methods to improve their score in a number of cases. The new results are shown in Table IV. The columns n_K^{dk} and DKSYN are repeated from Table III for convenience.

For instance, in case 1 of Table IV we see a situation, where DKSYN increased the original box by a factor $q = 1.006$, on which a robust gain $\gamma_{\mathrm{dk}} = 0.994$ was achieved with a controller of order $n_K^{\mathrm{dk}} = 13$ (line 1 of Table III). In that case *inner* achieved only the gain $h_* = 1.003$ on the original ball, using a controller of order $n_K = 2$. Now in line 1 of Table IV we allowed *inner* a controller of slightly larger order $n_K^+ = 4$, which is still way below 13, and we re-run it on the larger ball $q\boldsymbol{\Delta}$. This leads to $h_* = 0.982$, which means *inner* was able to recover due to the slightly more versatile controller structure, and it was able to deal with the enlarged ball $q\boldsymbol{\Delta}$. The results for *hybrid* and *outer* in Table IV are to be understood in the same sense.

The details of Table IV are as follows. Within the error margin of 1%, *outer* is equivalent to `DKSYN` on $q\boldsymbol{\Delta}$ in 23.3% of the cases, and has better results in 50% of the cases. Similarly, *hybrid* is equivalent to `DKSYN` in 6.7% of the cases, and is better than `DKSYN` in 73.3% of the cases. Finally, *inner* is equivalent to `DKSYN` in 10% of the cases, and has better results than `DKSYN` in 80% of the studies.

Altogether comparison with `DKSYN` was based on the following steps:

Algorithm 3. Comparison with `DKSYN`

▷ **Step 1.** Choose uncertainty box $\boldsymbol{\Delta}$ and run optimization-based relaxation methods with imposed controller structure of order n_K. The results of the three optimization based methods *inner*, *hybrid* and *outer* are compared in Table II.

▷ **Step 2.** Run `DKSYN` with $\boldsymbol{\Delta}$ on input. `DKSYN` returns $\gamma_{\rm dk}$ and a modified box $q\boldsymbol{\Delta}$ on which $\gamma_{\rm dk}$ is certified, achieved with a controller of order $n_K^{\rm dk}$. Comparison with the result for *inner* are shown in Table III.

▷ **Step 3.** If $q > 1$ and $n_K < n_K^{\rm dk}$ and $\gamma_{\rm dk} < \gamma$, increase n_K slightly to n_K^+. In all other cases keep $n_K^+ = n_K$.

▷ **Step 4.** Re-run optimization based methods on box $q\boldsymbol{\Delta}$ with controllers of order n_K^+. Compare estimate of robust H_∞ performance in all 4 cases (Table IV).

VII. Case Study

In this section, the three approaches presented in this paper are applied to a challenging engineering problem and their closed-loop system responses are compared.

The tail fin controlled missile described in [25] was used as the basis of the problem presented here, but new uncertainties were added to make the problem more challenging. The closed-loop interconnection is presented in Fig. 4, showing the controller to be designed, K, the generalized plant P, composed of 5 blocks, the uncertainties, and the weights for the performance channels, W_e and W_d.

The missile dynamics, illustrated in Fig. 5, include the rigid body dynamics $G_{\tt r}$, three flexible modes of the system $G_{\tt f}$ in parallel with $G_{\tt r}$, and the actuators and sensors dynamics, which are represented by second-order systems $G_{\tt act}$,$G_{\tt acc}$, $G_{\tt gyr}$. The plant P features two additional performance and robustness filters W_e, W_d, which altogether leads to $n_P = 29$ states. The control input of the missile is the tail fin deflection angle d_f through the actuator $G_{\tt act}$ and the measured output is $y = [\eta_m \quad q_m]^{\sf T}$, with acceleration η_m obtained from the accelerometer $G_{\tt acc}$ and pitch rate q_m obtained from the gyroscope $G_{\tt gyr}$. The actuator has a fin deflection limit of 40 deg. and a fin rate limit of 300 deg./s with description $u_{act} = G_{\tt act} \cdot d_f$ where

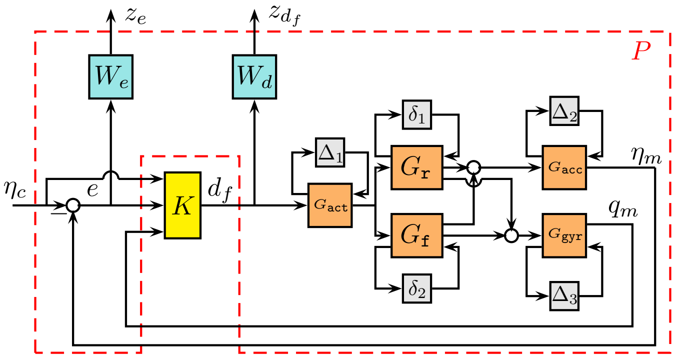

Fig. 4. Uncertain missile plant with controller. Real uncertainty is represented by the δ-blocks. Complex uncertain blocks are labeled Δ. For instance, the Δ_1 in loop with $G_{\tt act}$ stands short for $W_{\tt act}^{\Delta}\Delta_{\tt act}$, etc.

$$G_{\tt act}(s) = \frac{\omega_{\tt act}^2}{s^2 + 2\cdot 0.7\cdot \omega_{\tt act} s + \omega_{\tt act}^2}.$$

Similar second-order models are used for the accelerometer $G_{\tt acc}$ and the gyroscope $G_{\tt gyr}$, with numerical values given in Table V.

The rigid body dynamics, $G_{\tt r}$ of the missile is described by the state-space representation below, where the input is provided by the actuator and the output is the vector $[\eta_{rigid}\ q_{rigid}]^{\sf T}$:

$$G_{\tt r}\colon \begin{array}{l} \begin{bmatrix}\dot\alpha\\ \dot q\end{bmatrix} = \begin{bmatrix} Z_\alpha & 1\\ M_\alpha & M_q\end{bmatrix}\begin{bmatrix}\alpha\\ q\end{bmatrix} + \begin{bmatrix} Z_d\\ M_d\end{bmatrix} u_{act}\\ \begin{bmatrix}\eta_{\rm rigid}\\ q_{\rm rigid}\end{bmatrix} = \begin{bmatrix} V/kG\, Z_\alpha & 0\\ 0 & 1\end{bmatrix}\begin{bmatrix}\alpha\\ q\end{bmatrix} + \begin{bmatrix} V/kG\, Z_d\\ 0\end{bmatrix} u_{act}.\end{array}$$

Three flexible modes are added to represent the bending dynamics of the missile. We have

$$G_{\tt f}: \begin{bmatrix}\eta_{\rm flex}\\ q_{\rm flex}\end{bmatrix} = \sum_{i=1}^{3}\begin{bmatrix}\eta_i(s)\\ q_i(s)\end{bmatrix} u_{act},$$

where

$$\begin{bmatrix}\eta_i(s)\\ q_i(s)\end{bmatrix} = \frac{1}{s^2 + 2\cdot 0.02\cdot\omega_i s + \omega_i^2}\begin{bmatrix} s^2 K_{\eta_i}\\ sK_{q_i}\end{bmatrix},$$

$i = 1, 2, 3$, and then the overall dynamics are

$$\begin{bmatrix}\eta\\ q\end{bmatrix} = \begin{bmatrix}\eta_{\rm rigid}\\ q_{\rm rigid}\end{bmatrix} + \begin{bmatrix}\eta_{\rm flex}\\ q_{\rm flex}\end{bmatrix}.$$

The final measured outputs are then $\eta_m = G_{\tt acc}\cdot\eta$ and $q_m = G_{\tt gyr}\cdot q$. The values of the parameters of the plant and their respective ranges are presented in Table V.

Unmodeled high frequency dynamics at the actuator and sensor locations are assumed as of 0.1% uncertain at low frequency, and of 100% at high frequency. Explicitly, this corresponds to including the weight

$$W_{\tt act}^{\Delta}(s) = \frac{(s + \omega_{\tt act})^2}{(s + 10\cdot\omega_{\tt act})(s + 100\cdot\omega_{\tt act})},$$

TABLE V
VALUES AND UNCERTAINTY OF THE MISSILE PARAMETERS

Parameter	Nominal	Uncertainty	Parameter	Nominal	Uncertainty
Z_a	-5.24	±30%	Z_d	-0.73	0
M_a	-46.97	±15%	M_d	-1134	0
M_q	-4.69	±30%	V/kG	1.182	0
ω_1	368	±15%	$\omega_{\mathtt{acc}}$	188.5	0
ω_2	937	±15%	$\omega_{\mathtt{act}}$	377.0	0
ω_3	1924	±15%	$\omega_{\mathtt{gyr}}$	500.0	0
$K_{\eta 1}$	-0.943	0	K_{q1}	1024.1	0
$K_{\eta 2}$	0.561	0	K_{q2}	406.5	0
$K_{\eta 3}$	-0.312	0	K_{q3}	-1408.4	0

TABLE VI
MISSILE PLANT

Δ-structure		n_P	z-w	y-u
mixed	-1,-1,-1,-6,-6,-6,1,1,1	29	2 1	3 1

for the actuator, and similarly, for accelerometer and gyrometer with their respective frequencies $\omega_{\mathtt{acc}}$ and $\omega_{\mathtt{gyr}}$, shown in Table V.

As in (3), gathering all uncertain blocks for the missile yields $\Delta = \text{diag}\,[\Delta_p, \Delta_d]$, with $\Delta_p = \text{diag}\left[\delta_{Z_\alpha}, \delta_{M_\alpha}, \delta_{M_q}; \delta_{\omega_1} I_6, \delta_{\omega_2} I_6, \delta_{\omega_3} I_6\right]$ and $\Delta_d = \text{diag}\left[\Delta_{\mathtt{act}}, \Delta_{\mathtt{acc}}, \Delta_{\mathtt{gyr}}\right]$. Table VI summarizes the uncertainty in the system, terminology being that of Table I of the previous section, and Fig. 5 illustrates the variations in singular values.

Finally, the performance weights were chosen to reflect the following design requirements: Firstly, acceleration η_m should track the command η_c with a rise time of about 0.5 seconds. Hence the weighting function $W_e(s)$ for the transfer function from η_c to the tracking error $e := \eta_c - \eta_m$ was chosen as $W_e(s) := \frac{1}{100}\frac{s/10+100}{s/10+0.05}$.

Secondly, for robustness, the high-frequency rate of variation of the control signal and roll-off are captured and penalized through the constraint $||W_d(s)T_{d_f \eta_c}||_\infty \leq 1$, where $W_d(s)$ is a high-pass weighting $W_d(s) := (s/200(0.001s+1))^2$. This also permits to meet the imposed actuator deflection magnitude and rate limits of respectively 40 deg. and 300 deg./sec. Altogether the regulated output is $z = [W_e e \;\; W_d d_f]^\mathsf{T} = [z_e \;\; z_{d_f}]^\mathsf{T}$.

Using the methods discussed in this work, we then compute robust controllers K of order 6 with 3 inputs η_c, $e = \eta_c - \eta_m$, and q_m, and one output d_f.

The results are presented in Table VII, again with the terminology of Table II. The *hybrid* and *inner* relaxation methods found controllers with very similar performance, whereas the *outer* relaxation approach was not able to find a solution. The performance value returned by DKSYN was Inf, meaning that it could not find a solution either.

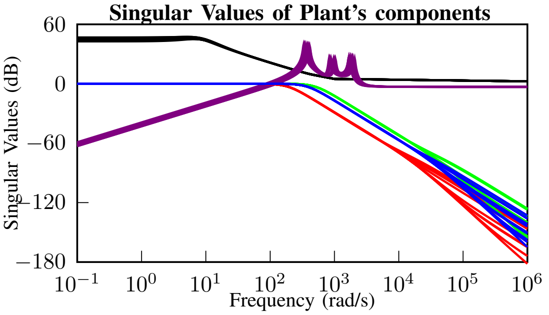

Fig. 5. Singular value plot for the components of plant G.

TABLE VII
RESULTS FOR MISSILE SYNTHESIS

n_K	*inner*			*hybrid*			*outer*	
	gain	certified	$\lvert\Delta_a\rvert$	gain	certified	$\lvert\Delta_{p,a}\rvert$	gain	certified
6	0.4310	0.4353	13	0.4416	0.4409	6	-	-

Responses of η_m, q_m and δ_c to η_c step inputs are shown in Fig. 6 for *inner* and in Fig. 7 for *hybrid*. Both of theses techniques were able to achieve prescribed design requirements.

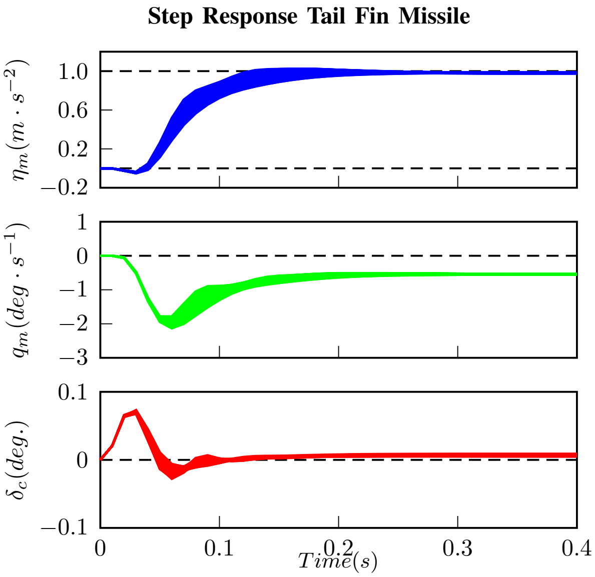

Fig. 6. Step responses for 50 sampled closed-loop models in the uncertainty range of the controlled missile with controller obtained by the *inner* approach.

VIII. CONCLUSION

We presented three relaxation approaches to the structured mixed parametric synthesis problem, based on different strategies, termed *inner*, *outer* and *hybrid*. A bench of 30 challenging test cases with mixed parametric and dynamic uncertainty was used to evaluate and compare these approaches. The inner relaxation generally produced the best results, its advantage being the most striking in situations with a large number r_i of repetitions. The approach termed *hybrid* came in second. The *outer* relaxation approach turned out to be more conservative and came third. As expected, this technique experienced difficulties for large repetitions of parameter uncertainty.

An out-of-competition comparison with the classical DGK-iteration based routine DKSYN was also organized. Owing to the fact that this technique modifies the uncertainty set $\mathbf{\Delta}$ given on entry, rendering a direct comparison impossible, we devised an evaluation procedure (given in section VI), which shows that, despite the age, the DKSYN function performs honorably. When comparing outer relaxation and DKSYN on

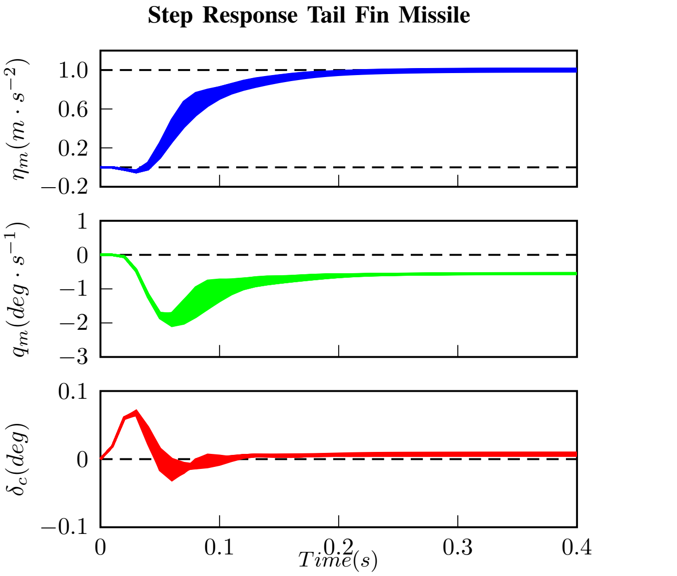

Fig. 7. Step responses for 50 sampled models in the uncertainty range of the controlled missile with controller obtained by the *hybrid* approach.

a same ball $q\mathbf{\Delta}$, the outer relaxation approach still performs better in half the test cases.

A more detailed study of a controlled missile with 6 real uncertain parameters with up to 6 repetitions and three uncertain complex blocks was presented in section VII.

Finally, both *inner* and *hybrid* approaches were confirmed as practical and non-conservative synthesis techniques.

References

[1] P. Apkarian, E. Feron, and P. Gahinet, “A parameter-dependent Lyapunov approach to robust control with real parametric uncertainty,” in *Proc. European Control Conf.*, Rome, 1995, pp. 2275–2280.

[2] V. Balakrishnan, “Linear matrix inequalities in robustness analysis with multipliers,” *Syst. Control Lett.*, vol. 25, no. 4, pp. 265–272, July 1995.

[3] J. P. How, W. M. Haddad, and S. R. Hall, “Application of popov controller synthesis to benchmark problems with real parametric uncertainty,” *AIAA J.*, vol. 17, no. 4, pp. 759–768, July–Aug. 1994.

[4] D. Peaucelle and D. Arzelier, “Robust multi-objective control toolbox,” in *Proc. IEEE Conf. Computer Aided Control Systems Design*, Munich, 2006, pp. 1152–1157.

[5] C. W. Scherer and I. E. Köse, “Gain-scheduled control synthesis using dynamic D-scales,” *IEEE Trans. Autom. Control*, vol. 57, pp. 2219–2234, Sept. 2012.

[6] M. K. H. Fan, A. L. Tits, and J. C. Doyle, “Robustness in the presence of mixed parametric uncertainty and unmodeled dynamics,” *IEEE Trans. Autom. Control*, vol. 36, pp. 25–38, Jan. 1991.

[7] J. Ackermann, *Multi-model approaches to robust control system design*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1985, pp. 108–130.

[8] J.-F. Magni, Y. Le Gorrec, and C. Chiappa, “A multimodel-based approach to robust and self-scheduled control design,” in *Proc. IEEE Conf. on Decision and Control*, vol. 3, 1998, pp. 3009–3014.

[9] G. Calafiore and M. Campi, “The scenario approach to robust control design,” *IEEE Trans. Autom. Control*, vol. 51, pp. 742–753, May 2006.

[10] P. Apkarian and D. Noll, “Nonsmooth H_∞ synthesis,” *IEEE Trans. Autom. Control*, vol. 51, pp. 71–86, Jan. 2006.

[11] P. Apkarian, Gahinet, and C. Buhr, “Multi-model, multi-objective tuning of fixed-structure controllers,” in *European Control Conf.* Strasbourg: IEEE, 2014, pp. 856–861.

[12] *Robust Control Toolbox 2015b*, The MathWorks Inc, 2015.

[13] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*. Englewood Cliffs, New Jersey: Prentice Hall, 1996.

[14] W. M. Lu, K. Zhou, and J. C. Doyle, “Stabilization of LFT systems,” in *Proc. IEEE Conf. Decision and Control*, Brighton, 1991, pp. 1239–1244.

[15] R. M. Redheffer, “On a certain linear fractional transformation,” *J. Math. Physics*, vol. 39, pp. 269–286, 1960.

[16] P. Apkarian, M. N. Dao, and D. Noll, “Parametric robust structured control design,” *IEEE Trans. Autom. Control*, vol. 60, pp. 1857–1869, July 2015.

[17] P. Apkarian and D. Noll, “Nonsmooth optimization for multidisk H_∞ synthesis,” *European J. Control*, vol. 12, no. 3, pp. 229–244, 2006.

[18] T. Iwasaki and S. Hara, “Well-posedness of feedback systems: insights into exact robustness analysis and approximate computatios,” *IEEE Trans. Autom. Control*, vol. 43, pp. 619–630, May 1998.

[19] P. Apkarian, D. Noll, and L. Ravanbod, “Nonsmooth bundle trust-region algorithm with applications to robust stability,” *Set-Valued and Variational Anal.*, vol. 24, no. 1, pp. 115–148, March 2016.

[20] ——, “Computing the structured distance to instability,” in *Proc. SIAM Conf. Control and its Applications*, Paris, 2015, pp. 423–430.

[21] L. Ravanbod, D. Noll, and P. Apkarian, “Branch and bound algorithm with applications to robust stability,” *J. Global Optimization*, Mar. 2016.

[22] M. Newlin and S. Glavaski, “Advances in the computation of the μ lower bound,” in *Proc. American Control Conf.*, Seatle, 1995, pp. 442–446.

[23] A. Packard and J. Doyle, “The complex structured singular value,” *Automatica*, vol. 29, no. 1, pp. 71–109, 1993.

[24] M. G. Safonov and R. Y. Chiang, “Real/complex K_m-synthesis without curve fitting,” in *Control and Dynamic Systems*. New York: Academic Press, 1993, vol. 56, pp. 303–324.

[25] D. L. Krueger, “Parametric uncertainty reduction in robust multivariable control,” Ph.D. dissertation, Dept. Elect. Comp. Eng., Naval Postgraduate School, Monterey, CA, 1993.

Raquel Stella da Silva de Aguiar received the M. Sc. degree in electrical engineering from the Instituto Militar de Engenharia (IME), Brazil, in 2013 and is currently preparing her Ph.D. at the Institut Supérieure de l’Aéronautique et de l’Espace (ISAE), France. Her research interests include automatic control and non-smooth optimization methods applied to control system design.

Pierre Apkarian received the Ph.D. degree in control engineering from the Ecole Nationale Supérieure de l’Aéronautique et de l’Espace (ENSAE), France, in 1988. He was qualified as a Professor from University of Toulouse (France) in both control engineering and applied mathematics in 1999 and 2001, respectively. Since 1988, he has been a Research Scientist at ONERA (Office National d’Etudes et de Recherches Aérospatiales) and an Associate Professor at the University of Toulouse. Pierre Apkarian has served as an associate editor for the IEEE Transactions on Automatic Control. His research interests include robust and gain-scheduling control. More recently, his research has focused on specialized non-smooth programming for control system design. He is co-developer of the HINFSTRUCT and SYSTUNE software in MATLAB’s Control System Toolbox.

Dominikus Noll received his Ph.D. and habilitation in 1983 and 1989 from Universität Stuttgart (Germany). Since 1995 he is a professor of applied mathematics at the University of Toulouse (France), and a distinguished professor of mathematics since 2009. Dr. Noll has held visiting positions at Uppsala University, Dalhousie University, the University of Waterloo, Simon Fraser University, and the University of British Columbia. Dr. Noll’s current research interests include nonlinear optimization, optimal control, projection-based iterative schemes, and robust feedback control design. He is co-inventor of the synthesis tools HINFSTRUCT and SYSTUNE. Dr. Noll is Associate Editor of the Journal of Convex Analysis.