Probabilistic μ-Analysis for Stability and H ∞ Performance Verification

Sovanna Thai, Clément Roos, Jean-Marc Biannic

Probabilistic μ-Analysis for Stability and $\mathcal{H}_\infty$ Performance Verification

Sovanna Thai, Clément Roos and Jean-Marc Biannic

***Abstract*— In this paper, new μ-based algorithms are proposed in the field of probabilistic robustness analysis. The objective is to compute tight bounds on the probability for a parametrically uncertain and possibly high order system to meet some stability and performance criteria. In this approach, uncertain parameters are treated as random variables with given probability distributions. Internal stability is treated first and $\mathcal{H}_\infty$ performance in the scalar case is considered next. The main contribution is to provide tight bounds on the probability measure in the latter case. The proposed algorithms are based on branch-and-bound techniques and tested on several benchmarks from the literature, demonstrating their efficiency and the potential of the probabilistic setting in reducing the conservatism of μ-analysis. They have been integrated in the SMART Robustness Analysis Library of the SMAC Toolbox developed by ONERA (http://w3.onera.fr/smac).**

***Index Terms*— probabilistic μ-analysis, (skewed) structured singular value, stability margin, $\mathcal{H}_\infty$ performance level**

I. Introduction

μ-analysis is a standard approach to robustness analysis, typically used to identify worst-case configurations and to compute guaranteed robustness margins (see [1, 2]). Despite its recognised maturity by the control community, it often leads to pessimistic results and consequently to unnecessarily conservative designs, at the cost of some performance loss. Probabilistic methods are therefore typically preferred (see e.g. [3, 4]), with Monte Carlo simulations representing the industrial standard. But these approaches have major drawbacks: the non-guaranteed detection of worst-case configurations may lead to erroneous conclusions, and the required sample size to reach satisfactory levels of confidence and accuracy can be prohibitive.

These observations seem to indicate that the two approaches are in fact complementary: deterministic methods could be used as a first validation step, leaving simulations to be performed only on the potentially problematic configurations. But in view of the above, it seems necessary to improve the interface between the two families of methods. This is the purpose of probabilistic μ-analysis, which consists of applying μ-based techniques while considering the uncertainties as random variables with known probability distributions. It guarantees the detection of worst-case scenarios, but it also quantifies their probability of occurrence, allowing comparison with a specified threshold.

Building up on the approach of [5], [6] and [7], this paper deals with LTI systems subject to parametric uncertainties, and provides some efficient probabilistic μ-based algorithms to calculate bounds on the probability of internal stability and $\mathcal{H}_\infty$ performance satisfaction. A particular emphasis is placed on the latter case. More precisely, an original algorithm is proposed to characterise the parametric domain on which performance violation is guaranteed. It is a more difficult problem than determining a guaranteed domain of performance satisfaction, but it offers some interesting complementary information. This is a major contribution of this paper, which is organised as follows. Section II first reminds some key definitions about μ-analysis, as well as the principle of branch-and-bound techniques, which serve as a starting point for probabilistic μ-analysis. The proposed algorithms are then introduced in Section III. Section IV finally presents some numerical results obtained on several benchmarks from the literature, demonstrating the efficiency of the proposed approach.

All authors are with the Information Processing and Systems Department of ONERA, The French Aerospace Lab, Toulouse, France <sovanna.thai, clement.roos, jean-marc.biannic>@onera.fr

II. Problem Statement

Let us consider the standard interconnection for robust stability analysis as depicted in Fig. 1 (left), where $M(s)$ is a stable real-valued LTI plant representing the nominal closed-loop system, and Δ is a block-diagonal LTI operator gathering all the real parametric uncertainties:

$$\Delta = \mathrm{diag}(\delta_1 I_{n_1}, \ldots, \delta_N I_{n_N}) \tag{1}$$

where $(\delta_1, \ldots, \delta_N) \in \mathbb{R}^N$. The set of matrices with such a block-diagonal structure is denoted $\mathbf{\Delta}$. Let then $k\mathcal{B}_{\mathbf{\Delta}} = \{\Delta \in \mathbf{\Delta} : \overline{\sigma}(\Delta) < k\}$, where $\overline{\sigma}(\Delta)$ denotes the largest singular value of Δ. Singularity of the interconnection at frequency $\omega \in \mathbb{R}_+$ is characterised by $\det(I - M(j\omega)\Delta) = 0$.

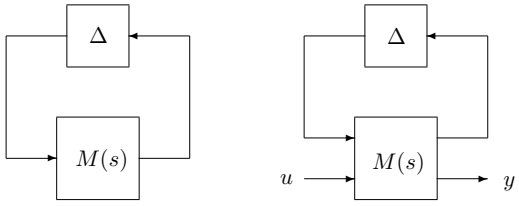

Fig. 1. Standard interconnections for stability analysis (left) and performance analysis (right)

μ-analysis relies on the structured singular value, introduced in [1] and defined as follows.

Definition 2.1: Let $\omega \in \mathbb{R}_+$ be a given frequency. If no matrix $\Delta \in \mathbf{\Delta}$ makes the interconnection of Fig. 1 (left) unstable, then the structured singular value $\mu_{\mathbf{\Delta}}(M(j\omega))$ is defined to be equal to zero. Otherwise:

$$\mu_{\mathbf{\Delta}}(M(j\omega)) = \left[\min_{\Delta \in \mathbf{\Delta}} \{\overline{\sigma}(\Delta) : \det(I - M(j\omega)\Delta) = 0\}\right]^{-1} \tag{2}$$

The robust stability margin is then defined as the inverse of the maximum value of $\mu_{\mathbf{\Delta}}(M(j\omega))$ over the whole frequency range $\mathbb{R}_+$:

$$k_r = \left[\sup_{\omega \in \mathbb{R}_+} \mu_{\mathbf{\Delta}}(M(j\omega)) \right]^{-1} \quad (3)$$

If the structure $\mathbf{\Delta}$ is now split into two distinct block structures $\mathbf{\Delta} = \mathrm{diag}(\mathbf{\Delta}_1, \mathbf{\Delta}_2)$, and admissible uncertainties are taken in $\mathbf{\Delta}_s = \mathrm{diag}(\mathcal{B}_{\mathbf{\Delta}_1}, \mathbf{\Delta}_2)$, a skewed version of the structured singular value can be defined [8].

Definition 2.2: Let $\omega \in \mathbb{R}_+$ be a given frequency. If no matrix $\Delta = \mathrm{diag}(\Delta_1, \Delta_2) \in \mathbf{\Delta}_s$ makes the interconnection unstable, then the skewed structured singular value $\nu_{\mathbf{\Delta}_s}(M(j\omega))$ is defined to be equal to zero. Otherwise:

$$\nu_{\mathbf{\Delta}_s}(M(j\omega)) = \left[\min_{\Delta \in \mathbf{\Delta}_s} \{ \overline{\sigma}(\Delta_2) : \det(I - M(j\omega)\Delta) = 0 \} \right]^{-1} \quad (4)$$

Even putting aside the NP-hardness [9] of computing k_r or its skewed version, which usually requires to use upper and lower bounds $\overline{k_r}$ and $\underline{k_r}$ instead of the real value, the robust stability margin remains inherently conservative. Indeed, it only yields the largest open ball $k_r \mathcal{B}_{\mathbf{\Delta}}$ on which stability is guaranteed, regardless of the actual shape of the exact stability domain. To obtain a better approximation of the latter, branch-and-bound algorithms are used [10, 11]. Then, without introducing much additional numerical complexity, a probabilistic framework can be considered: if the parametric uncertainties δ_i are now seen as random variables with given probability density functions f_i, it is of interest to compute the probability measure of the stability domain. In other words, the following problem can be stated.

Problem 2.1: Calculate the probability $P_{\mathbf{\Delta},f}(M(s))$ that the interconnection $M(s) - \Delta$ is stable when $\Delta \in \mathcal{B}_{\mathbf{\Delta}}$.

Let us now consider the interconnection of Fig. 1 (right) corresponding to performance analysis. The transfer from u to y is given by the upper linear fractional transformation:

$$\mathcal{F}_u(M(s), \Delta) = M_{22}(s) + M_{21}(s)\Delta(I - M_{11}(s)\Delta)^{-1} M_{12}(s) \quad (5)$$

For some technical reasons to be clarified in Section III-B, it is assumed hereafter that the transfer function from u to y is single input single output. Such an assumption is not so restrictive in practice. If needed, several performance channels can be processed sequentially. The following proposition makes the link between $\mathcal{H}_\infty$ performance and the skewed structured singular value.

Proposition 2.1: The following statements are equivalent:

1) $\max_{\Delta \in \mathcal{B}_{\mathbf{\Delta}}} \|\mathcal{F}_u(M(s), \Delta)\|_\infty \leq \gamma$
2) $\forall \omega \in \mathbb{R}_+, \nu_{\mathbf{\Delta}_a}(M(j\omega)) \leq \gamma$ where $\mathbf{\Delta}_a = \mathrm{diag}(\mathbf{\Delta}, \mathbb{C})$

In the same vein as with robust stability, branch-and-bound techniques can be used to compute a domain of guaranteed performance satisfaction, whose probability measure can then be evaluated. The problem of interest is as follows:

Problem 2.2: Given a performance level $\gamma > 0$, compute the probability $P^{\gamma}_{\mathbf{\Delta},f}(M(s))$ that $\|\mathcal{F}_u(M(s), \Delta)\|_\infty \leq \gamma$ when $\Delta \in \mathcal{B}_{\mathbf{\Delta}}$.

III. Probabilistic μ-analysis

A. Stability Analysis

The exact computation of $P_{\mathbf{\Delta},f}(M(s))$ is unsurprisingly not achievable in the general case. But branch-and-bound algorithms provide a subdomain D_s of the normalised parametric domain $\Theta = \{\delta \in \mathbb{R}^N : |\delta_i| \leq 1\} = [-1, 1]^N$ on which stability is guaranteed, which gives a lower bound on the sought probability. Moreover, D_s conveniently takes the form of a finite union of disjoint boxes

$$D_s = \bigcup_k D_s^{(k)} \quad \text{where} \quad D_s^{(k)} = [a_1^{(k)}, b_1^{(k)}] \times ... \times [a_N^{(k)}, b_N^{(k)}] \quad (6)$$

whose probability measures are straightforward to compute:

$$p(D_s^{(k)}) = \prod_{i=1}^{N} \int_{a_i^{(k)}}^{b_i^{(k)}} f_i(\delta_i) d\delta_i \quad (7)$$

There is much interest in also calculating an upper bound on $P_{\mathbf{\Delta},f}(M(s))$, in order not only to get a more accurate picture of the stability domain, but also potentially to quickly invalidate the current design with respect to a specified tolerance level. In other words, to further reduce the conservatism of the proposed probabilistic approach, the computation of a domain of guaranteed instability $D_{\bar{s}}$ is desirable, so that $P_{\mathbf{\Delta},f}(M(s))$ can be bounded as follows:

$$p(D_s) \leq P_{\mathbf{\Delta},f}(M(s)) \leq 1 - p(D_{\bar{s}}) \quad (8)$$

This proves to be straightforward if one realises that methods to calculate the robustness margin rely on detecting when the poles of the nominally stable system $M(s)$ reach the imaginary axis as the size of Δ increases. Thus, the same methods applied to a nominally unstable interconnection leads to a "robustness margin for instability" [5]. Integrated into a branch-and-bound scheme, this yields the desired domain of guaranteed instability.

While the branch-and-bound and the probability calculations could be kept separate, it is computationally more interesting to combine the two aspects into a single algorithm. Indeed, this allows to integrate probability-related stopping criteria into the algorithm, e.g. interrupting the algorithm if all the boxes have a probability measure less than a given threshold p_{min}. Such a strategy is described in Algorithm 1, whose outputs are a partition of $\Theta = [-1, 1]^N$ into

$$\Theta = D_s \cup D_{\bar{s}} \cup D_{s_u} \quad (9)$$

and the probability measure of each component of this partition. The third term D_{s_u} corresponds to the domain of uncategorised stability. Its presence stems from the fact that, realistically, the branch-and-bound algorithm can only approximate the domains of stability and instability. Ideally, D_{s_u} should be as small as possible, although depending on the validation requirements, a coarse result can be sufficient to decide if the investigated system is satisfactory.

For inconclusive iterations, the partitioning of the box is classically done along its longest edge, but other strategies can be implemented. In particular, μ-sensitivities can be used to reduce computational time [12], especially for systems with numerous parametric uncertainties.

Algorithm 1 Probabilistic μ-analysis for stability

Inputs: $M(s)$, $\mathbf{\Delta}$, $f_i(\delta_i)$ for $i = 1, \dots, N$, p_{min}, p_{max}
Outputs: D_s, $D_{\bar{s}}$, D_{s_u}, $p(D_s), p(D_{\bar{s}}), p(D_{s_u})$
$\mathcal{L} \leftarrow \{\Theta\}$
$D_s, D_{\bar{s}}, D_{s_u} \leftarrow \emptyset$
$p(D_s), p(D_{\bar{s}}), p(D_{s_u}) \leftarrow 0$
while $\mathcal{L} \neq \emptyset$ and $p(D_s) + p(D_{\bar{s}}) < p_{max}$ **do**
    select and remove the box $\tilde{\Theta}$ from $\mathcal{L}$ with the highest probability measure and compute the interconnection $\tilde{M}(s) - \tilde{\Delta}$ normalised w.r.t. $\tilde{\Theta}$
    compute a guaranteed robust stability margin $\underline{k_r}$ for $\tilde{M}(s) - \tilde{\Delta}$
    if $\underline{k_r} \geq 1$ **then** ▷ *analysis is conclusive*
        if $\mathcal{F}_u(\tilde{M}(s), \tilde{\Delta})$ is nominally stable **then**
            add $\tilde{\Theta}$ to D_s
            $p(D_s) \leftarrow p(D_s) + p(\tilde{\Theta})$
        else
            add $\tilde{\Theta}$ to $D_{\bar{s}}$
            $p(D_{\bar{s}}) \leftarrow p(D_{\bar{s}}) + p(\tilde{\Theta})$
        end if
    else ▷ *analysis is inconclusive*
        if $p(\tilde{\Theta}) > p_{min}$ **then**
            partition $\tilde{\Theta}$ and add the boxes obtained in $\mathcal{L}$
        else
            add $\tilde{\Theta}$ to D_{s_u}
            $p(D_{s_u}) \leftarrow p(D_{s_u}) + p(\tilde{\Theta})$
        end if
    end if
end while
if $\mathcal{L} \neq \emptyset$ **then**
    add all remaining boxes $\tilde{\Theta}_k$ of $\mathcal{L}$ in D_{s_u}
    $p(D_{s_u}) \leftarrow p(D_{s_u}) + \sum_k p(\tilde{\Theta}_k)$
end if

B. $\mathcal{H}_\infty$ Performance Analysis

Let us now consider Problem 2.2. As for robust stability, the idea is to combine branch-and-bound with μ-analysis to compute both a domain of guaranteed performance satisfaction D_γ and a domain of guaranteed performance violation $D_{\bar{\gamma}}$, whose probability measures can then be calculated. Proposition 2.1 provides the means to calculate D_γ, but a difficulty arises in the computation of $D_{\bar{\gamma}}$. Indeed, performance violation on Θ is equivalent to

$$\min_{\Delta \in \mathcal{B}_{\mathbf{\Delta}}} \sup_{\omega \in \mathbb{R}_+} |\mathcal{F}_u(M(j\omega), \Delta)| > \gamma \tag{10}$$

which, as a minimax problem, is more complicated than verifying performance satisfaction. As a preliminary step to tackle this problem, a useful property on LFT inversion is reminded (see e.g. [13]).

Lemma 3.1: Let $M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}$, and assume that M_{22} is nonsingular. Then $\mathcal{F}_u(M, \Delta)^{-1} = \mathcal{F}_u(N, \Delta)$, where $N = \begin{bmatrix} M_{11} - M_{12}M_{22}^{-1}M_{21} & -M_{12}M_{22}^{-1} \\ M_{22}^{-1}M_{21} & M_{22}^{-1} \end{bmatrix}$.

Remark 3.1: For a scalar performance channel, the assumption that M_{22} is nonsingular simply means that it is nonzero, i.e. there is effectively a transfer from u to y. Since it is the case of interest for performance analysis, this is not restrictive.

Proposition 3.1 then derives a sufficient condition for Inequality (10) to hold, in the form of a skew-μ calculation involving the inverse transfer from u to y.

Proposition 3.1: Let $\mathcal{I} = \{\omega \in \mathbb{R}_+ : |M_{22}(j\omega)| > \gamma\}$. If there exists $\omega_0 \in \mathcal{I}$ such that

$$\nu_{\mathbf{\Delta}_a}(N(j\omega_0)) < \frac{1}{\gamma} \tag{11}$$

where $\mathbf{\Delta}_a = \mathrm{diag}(\mathcal{B}_{\mathbf{\Delta}}, \mathbb{C})$ and $N(j\omega)$ is such that $(\mathcal{F}_u(M(j\omega), \Delta))^{-1} = \mathcal{F}_u(N(j\omega), \Delta)$, then Inequality (10) holds, and performance violation is guaranteed on Θ, i.e. $\forall \Delta \in \mathcal{B}_{\mathbf{\Delta}}$.

Proof: First, note that nominal performance violation is equivalent to

$$\sup_{\omega \in \mathbb{R}_+} |M_{22}(j\omega)| > \gamma \tag{12}$$

which guarantees that $\mathcal{I}$ is nonempty, and that $N(j\omega)$ is well-defined for each $\omega \in \mathcal{I}$. Assume now that there exists $\omega_0 \in \mathcal{I}$ such that (11) holds. From the main loop theorem ([13]), this is equivalent to

$$\begin{cases} \mu_{\mathbf{\Delta}}(N_{11}(j\omega_0)) < 1 \\ \max\limits_{\Delta \in \mathcal{B}_{\mathbf{\Delta}}} |\mathcal{F}_u(N(j\omega_0), \Delta)| < \dfrac{1}{\gamma} \end{cases} \tag{13}$$

Since $|\mathcal{F}_u(N(j\omega_0), \Delta)| = |\mathcal{F}_u(M(j\omega_0), \Delta)|^{-1}$ $\forall \Delta \in \mathcal{B}_{\mathbf{\Delta}}$, the second inequality of (13) can be rewritten as

$$\min_{\Delta \in \mathcal{B}_{\mathbf{\Delta}}} |\mathcal{F}_u(M(j\omega_0), \Delta)| > \gamma \tag{14}$$

which is a sufficient condition for Inequality (10) to hold. ■

Remark 3.2: The step from (13) to (14) strongly relies on the scalar nature of the performance channel and cannot be easily generalised to higher dimensions although this will be investigated in a near future.

A strategy to verify performance violation directly follows from Proposition 3.1, as outlined in Algorithm 2.

Algorithm 2 Verification of performance violation on Θ

Inputs: $M(s)$, $\mathbf{\Delta}$, γ
Output: `true` if performance violation is guaranteed on Θ, `false` otherwise
if $\|M_{22}(s)\|_\infty \leq \gamma$ **then**
    return `false`
else
    select $\omega_0 \in \mathcal{I}$
    compute $\overline{\nu}_{\mathbf{\Delta}_a}(N(j\omega_0))$, where $N(j\omega_0)$ is given by Lemma 3.1
    if $\overline{\nu}_{\mathbf{\Delta}_a}(N(j\omega_0)) < 1/\gamma$ **then**
        return `true`
    else
        return `false`
    end if
end if

Note that the output `false` of Algorithm 2 merely means that the analysis was inconclusive, and does not exclude performance violation on Θ. More precisely, inconclusive cases can result from three causes:

1) the sufficient condition of Proposition 3.1 is not necessary,
2) the selected ω_0 in Algorithm 2 is not satisfactory,
3) the computation of the skew-μ upper bound is not sufficiently accurate.

In particular, the heuristics used to select ω_0 can prove critical. It is readily apparent that exploring a frequency grid or frequency interval would be computationally inefficient, since only one frequency is sought, which is not even guaranteed to exist. On the other hand, the peak frequency of $M_{22}(s)$, while appearing as a natural choice for ω_0, offers no guarantee of optimality.

The methods to validate performance satisfaction (Proposition 2.1) and performance violation (Proposition 3.1) on a given box can be integrated into a branch-and-bound scheme structurally similar to Algorithm 1. The essential difference lies in lines 8 through 24 of the algorithm, which must be replaced by a performance analysis:

- if $\|\tilde{M}_{22}(s)\|_\infty < \gamma$, then attempt to verify performance satisfaction on $\tilde{\Theta}$ using Proposition 2.1,
- otherwise, attempt to verify performance violation on $\tilde{\Theta}$ using Algorithm 2.

$\tilde{\Theta}$ is then either partitioned if the aforementioned test fails or added to the adequate domain (D_γ, $D_{\tilde{\gamma}}$, or the domain of uncategorised performance D_{γ_u}), whose probability measure is then updated. Note that all three issues raised on Algorithm 2 are alleviated by the branch-and-bound. Indeed, barring extreme cases of sensitivities to the uncertainties, it is realistic to assume that the transfer $\mathcal{F}_u(M(s), \Delta)$ varies less and less as Δ is taken in smaller and smaller boxes. Thus, the sufficient condition of Proposition 3.1 becomes not only more relevant, but also less sensitive to the choice of ω_0 as a result. Moreover, the gap between the bounds of μ is known to become arbitrarily small as the domain Θ is partitioned into smaller boxes [14], which resolves item 3. However, relying on the asymptotic behaviour of the branch-and-bound algorithm may be unsatisfactory from a computational point of view.

IV. Applications

A. Illustrative Example

The following example, taken from [7], aims to illustrate the proposed algorithms on a two-dimensional problem that allows graphical representation. Consider the system given by the state-space representation

$$\begin{cases} \dot{x} = \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \\ y = \begin{bmatrix} 1 & 0 \end{bmatrix} x \end{cases} \tag{15}$$

where a_0 and a_1 are two uncertain parameters described by

$$\begin{aligned} a_0 &= 1 + q_0, \quad |q_0| \leq 2 \\ a_1 &= 0.8 + q_1, \quad |q_1| \leq 1 \end{aligned} \tag{16}$$

The parametric domain Θ is therefore the set of all possible values of the ordered pair (q_0, q_1), i.e. $\Theta = [-2, 2] \times [-1, 1]$. For such an example, the Routh criterion can be used to determine that the domain of stability is exactly $]-0.5, 1]\times]-0.8, 1]$. Assuming a uniform distribution of the uncertainties on Θ, this represents 67.5% of the domain. The proposed algorithm for stability analysis yields $P_{\mathbf{\Delta},f}(M(s)) \in [0.672, 0.679]$, which is a good estimation.

The same problem is now solved assuming $q_0 \sim \mathcal{N}_{[-2,2]}(0, (2/3)^2)$ and $q_1 \sim \mathcal{N}_{[-1,1]}(0, (1/3)^2)$, where $\mathcal{N}_{[a,b]}(m, \sigma^2)$ denotes the truncated normal distribution. This type of distribution refers to a case where the nominal configuration is deemed more likely, and with the probability of occurrence of a configuration becoming smaller as it gets farther from the nominal point. Algorithm 1 now leads to $P_{\mathbf{\Delta},f}(M(s)) \in [0.925, 0.931]$, which is a more optimistic result. This highlights the interest and potential of the probabilistic framework in reducing the conservatism of μ-analysis. Regarding $\mathcal{H}_\infty$ performance analysis and setting the performance level $\gamma = \sqrt{2}$, a uniform distribution of the uncertainties leads to $P^\gamma_{\mathbf{\Delta},f}(M(s)) \in [0.344, 0.359]$, while the truncated normal distribution changes this estimation to $P^\gamma_{\mathbf{\Delta},f}(M(s)) \in [0.411, 0.441]$. The improvement is less drastic, but this can be explained by the fact that the nominal configuration is close to the limit of performance. This case is less likely to occur for a controlled system whose controller is designed using robust synthesis methods.

Table I compares the results obtained with Algorithm 1 from [7] (whose Matlab code has been provided by the authors) and with the algorithms from the present paper. A dramatic reduction in the computational time can be observed. This is due to the computation of guaranteed instability or non-performance domains, which is done here but not in [7]. The difference is further illustrated by Fig. 2 and 3. The comparison was done on a CentOS Linux 7.4 Workstation with a CPU Intel Xeon W3530 running at 2.80GHz and 6 GB of RAM, with Matlab R2017b installed.

Algorithm	Stability analysis $p_{min} = 10^{-5}$			Performance analysis $p_{min} = 10^{-4}$		
	$p(D_s)$	$p(D_{\bar{s}})$	time	$p(D_\gamma)$	$p(D_{\tilde{\gamma}})$	time
[7]	92.52%	n/a	2861s	41.30%	n/a	6843s
present paper	92.58%	6.94%	341s	41.56%	48.71%	268s

TABLE I
COMPARISON BETWEEN ALGORITHM 1 FROM [7] AND THE ALGORITHMS OF THE PRESENT PAPER

Remark 4.1: The adaptation of Algorithm 1 to $\mathcal{H}_\infty$ performance analysis is only applicable to boxes on which stability is already guaranteed. In the proposed implementation, a preliminary stability analysis is therefore performed, and the identified stable boxes serve as inputs for the performance analysis. Both analyses are accounted for in Table I.

Remark 4.2: The limiting threshold on the probability measure of the analysed boxes is used slightly differently between the algorithm from [7] and Algorithm 1. The former bisects all uncategorised boxes, then checks the probability measure of the resulting boxes to determine if they are to

be analysed or not. The latter uses the probability measure of uncategorised boxes to determine if partitioning should be done. Thus, for the same limiting threshold p_{min}, Algorithm 1 analyses more boxes.

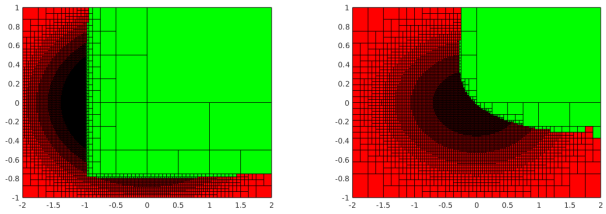

Fig. 2. Stability (left) and performance (right) analyses without instability/non-performance verification [7]: the green domain satisfies the tested criterion, the red domain is uncategorised

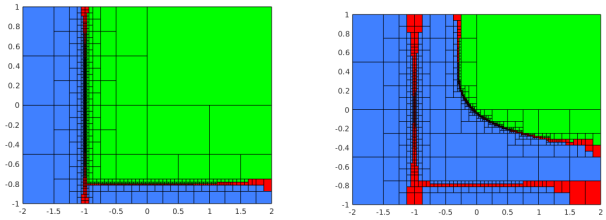

Fig. 3. Stability (left) and performance (right) analyses with instability/non-performance verification: the green domain satisfies the tested criterion, the blue domain invalidates it, and the red domain is uncategorised

B. Numerical Results

Algorithm 1 is now applied to several benchmarks described in the literature. For these tests, all normalised uncertainties follow the truncated normal distribution $\mathcal{N}_{[-1,1]}(0,(1/3)^2)$, and two partitioning strategies are used: bisection along the longest edge, and bisection along the highest μ-sensitivity. The limiting threshold for bisecting boxes is set to $p_{min} = 10^{-4}$. The results are summarised in Table II. The notation $m \times p$ in the description of Δ means that Δ contains m blocks of size $p \times p$, all blocks being real diagonal. R denotes the ratio of the probability measure of the categorised boxes (stable or unstable) over the total probability measure of the operating domain Θ, i.e. $R = \frac{p(D_s)+p(D_{\bar{s}})}{p(D_s)+p(D_{\bar{s}})+p(D_{s_u})} = p(D_s) + p(D_{\bar{s}})$. The obtained ratios are close to 1 for all benchmarks, meaning that the domains of stability and instability are well approximated by D_s and $D_{\bar{s}}$ respectively, leaving only a small domain D_{s_u} as uncategorised (in the sense of probability measure).

Regarding the partitioning strategies, while the use of μ-sensitivities tends to improve both the domains computation and the running time, this is not always the case, benchmark 10 being a counterexample. However, this system aside, the use of μ-sensitivities leads to significant time reduction as the dimension of the problem increases, as observed on benchmarks 19, 21 and 26, whose parametric domains have respective dimensions 9, 9 and 23. This shows that, in spite of the effectiveness of naive branching schemes [15], there is still an interest in investigating more sophisticated ones.

Table III shows test results for probabilistic performance analysis, where R now denotes the ratio $R = \frac{p(D_\gamma)+p(D_{\bar{\gamma}})}{p(D_\gamma)+p(D_{\bar{\gamma}})+p(D_{\gamma_u})}$ (which differs from $p(D_\gamma) + p(D_{\bar{\gamma}})$ if stability is not achieved on the complete domain Θ). As mentioned in Section III-B, the peak frequency of $M_{22}(s)$ is selected as the frequency ω_0. The relevance of μ-sensitivities for partitioning is less consistent, leading to a slight decrease of the probability measure of D_{γ_u} at the cost of increased execution time for all but benchmark B5. The latter benefits the most from the use of μ-sensitivities, with a drastic reduction of $p(D_{\gamma_u})$ (more than 25%) and an execution time reduced by a factor of approximately 3.

To assess the relevance of the chosen ω_0, sample configurations have been drawn in each uncategorised box to see whether a configuration satisfying the performance level could be found. This would mean that a suitable ω_0 does not exist, i.e. there is nothing to gain from improving the choice of ω_0. For each nominally non-performing N-dimensional box, the samples drawn are the corners of the box, for a total of 2^N samples per box. The reason behind this sampling method is that the farthest points from the centre of the box are empirically the most likely to exhibit the most substantial changes on their transfer function. Also, the relatively small number of dimensions in the considered benchmarks keeps the sample size from being prohibitive. Results show that the loss on $p(D_{\bar{\gamma}})$ remains reasonable on the tested benchmarks, not exceeding 3%. This means that no change in the heuristics can lead to dramatic improvements of the algorithm. However, if improvements are sorely needed, then a possible solution could be to consider as additional candidates for ω_0 the peak frequencies of the $2N$ configurations located at the centre of the $(N-1)$-dimensional faces of the currently studied N-dimensional box. This choice constitutes a compromise between taking configurations that are far from the centre of the box, and limiting the number of candidate frequencies, since taking the corners as was done for the sampling would lead to 2^N additional frequencies.

V. Conclusion

In this paper, new contributions are proposed to improve the efficiency of robustness analysis by combining deterministic μ-based methods with a probabilistic framework. Both robust stability and $\mathcal{H}_\infty$ performance are addressed. The proposed algorithms are implemented in Matlab and tested on several benchmarks taken from the literature. The resulting routines are available in the SMART Library of the SMAC Toolbox at `http://w3.onera.fr/smac`. Beside the efficiency of the algorithms, test results showcase that probabilistic μ-analysis can be an adequate response to reduce the conservatism of a purely deterministic approach. In the overall validation process of a controlled system, probabilistic μ-analysis complements Monte Carlo-based methods nicely. It provides relevant inputs for the sampling strategy of the simulation phase, and could ultimately lighten the tedium of control laws development. In light of the above, probabilistic μ-analysis appears as an attractive method for the validation of controlled systems with parametric uncertainties, provided that reliable probability density functions are available. Future work will focus on extending the presented $\mathcal{H}_\infty$ performance analysis method to the MIMO case.

Benchmark	States	Structure of $\mathbf{\Delta}$	Bisection along the longest edge		Bisection along the highest μ-sensitivity	
			ratio R	time (s)	ratio R	time (s)
3: Academic example [16]	4	2×2	0.9836	249	0.9846	262
4: Inverted pendulum [17]	4	3×1	0.9994	14	0.9997	6
5: Anti-aliasing filter [18]	2	$3 \times 1 + 1 \times 2$	0.9998	9	0.9999	6
9: Bank-to-turn missile [19]	6	4×1	0.9600	492	0.9903	229
10: Aeronautical vehicle [20]	8	4×1	0.9817	716	0.9695	1327
19: Drive-by-wire vehicle [21]	4	$2 \times 1 + 7 \times 2$	0.9970	418	0.9999	15
20: Re-entry vehicle (model developed by ONERA)	7	$3 \times 1 + 1 \times 4 + 1 \times 6$	0.9999	51	1-2E-5	45
21: Space shuttle [22]	34	9×1	0.9634	335	0.9992	10
26: Hard disk drive [23]	29	$19 \times 1 + 4 \times 2$	0.9490	2073	0.9993	66

TABLE II

STABILITY ANALYSIS RESULTS ON SELECTED BENCHMARKS; BENCHMARK NUMBERS REFER TO THOSE USED IN [24]

Benchmark	States	Structure of $\mathbf{\Delta}$	Bisection along the longest edge		Bisection along the highest μ-sensitivity	
			ratio	time (s)	ratio	time (s)
B1: Academic example [25]	4	3×1	0.8965	614	0.9017	951
B2: Inverted pendulum [17]	4	3×1	0.9945	65	0.9964	80
B3: DC motor [26]	4	$3 \times 1 + 1 \times 2$	0.9238	1094	0.9545	1144
B4: Bus steering system [27]	9	$1 \times 2 + 1 \times 3$	0.9320	601	0.9379	1268
B5: Hydraulic servo system [28]	9	8×1	0.6864	12906	0.9470	4269

TABLE III

PERFORMANCE ANALYSIS RESULTS ON SELECTED BENCHMARKS

REFERENCES

[1] J. Doyle, "Analysis of feedback systems with structured uncertainties," *IEE Proceedings D - Control Theory and Applications*, vol. 129, no. 6, pp. 242–250, 1982.

[2] M. G. Safonov, "Stability margins of diagonally perturbed multivariable feedback systems," *IEE Proceedings D - Control Theory and Applications*, vol. 129, no. 6, pp. 251–256, 1982.

[3] X. Chen, J. L. Aravena, and K. Zhou, "Risk analysis in robust control - making the case for probabilistic robust control," in *Proceedings of the American Control Conference*, vol. 3, 2005, pp. 1533–1538.

[4] G. Calafiore, F. Dabbene, and R. Tempo, "A survey of randomized algorithms for control synthesis and performance verification," *Journal of Complexity*, vol. 23, no. 3, pp. 301 – 316, 2007.

[5] S. Khatri and P. A. Parrilo, "Guaranteed bounds for probabilistic μ," in *Proceedings of the 37th IEEE Conference on Decision and Control*, 1998, pp. 3349–3354.

[6] G. Balas, P. Seiler, and A. Packard, "Analysis of an uav flight control system using probabilistic mu," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, 2012.

[7] A. Falcoz, D. Alazard, and C. Pittet, "Probabilistic μ-analysis for system performances assessment." *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 399 – 404, 2017, 20th IFAC World Congress.

[8] M. K. Fan and A. Tits, "A measure of worst-case $\mathcal{H}_\infty$ performance and of largest acceptable uncertainty," *Systems & Control Letters*, vol. 18, no. 6, pp. 409 – 421, 1992.

[9] R. P. Braatz, P. M. Young, J. C. Doyle, and M. Morari, "Computational complexity of μ calculation," *IEEE Transactions on Automatic Control*, vol. 39, no. 5, pp. 1000–1002, 1994.

[10] M. P. Newlin and P. M. Young, "Mixed mu problems and branch and bound techniques," in *Proceedings of the 31st IEEE Conference on Decision and Control*, 1992, pp. 3175–3180 vol.4.

[11] C. Roos and J. Biannic, "Efficient computation of a guaranteed stability domain for a high-order parameter dependent plant," in *Proceedings of the 2010 American Control Conference*, 2010, pp. 3895–3900.

[12] J. Lesprier, C. Roos, and J.-M. Biannic, "Improved μ upper bound computation using the μ-sensitivities," *IFAC-PapersOnLine*, vol. 48, no. 14, pp. 215 – 220, 2015, 8th IFAC Symposium on Robust Control Design.

[13] J. Doyle, A. Packard, and K. Zhou, "Review of LFTs, LMIs, and μ," in *Proceedings of the 30th IEEE Conference on Decision and Control*, 1991, pp. 1227–1232.

[14] C. Roos, F. Lescher, J.-M. Biannic, C. Döll, and G. Ferreres, "A set of μ-analysis based tools to evaluate the robustness properties of high-dimensional uncertain systems," in *Proceedings of the IEEE International Symposium on Computer-Aided Control System Design*, 2011, pp. 644–649.

[15] X. Zhu, Y. Huang, and J. Doyle, "Soft vs. hard bounds in probabilistic robustness analysis," in *Proceedings of 35th IEEE Conference on Decision and Control*, vol. 3, 1996, pp. 3412–3417.

[16] T. Matsuda, M. Kawanishi, and T. Narikiyo, "Computation of real structured singular value by stability feeler," in *Proceedings of the 7th Asian Control Conference*, 2009, pp. 672–677.

[17] A. Yazıcı, A. Karamancıoğlu, and R. Kasimbeyli, "A nonlinear programming technique to compute a tight lower bound for the real structured singular value," *Optimization and Engineering*, vol. 12, no. 3, pp. 445–458, 2011.

[18] P. Seiler, A. Packard, and G. J. Balas, "A gain-based lower bound algorithm for real and mixed μ problems," *Automatica*, vol. 46, no. 3, pp. 493 – 500, 2010.

[19] K. A. Wise, "Comparison of six robustness tests evaluating missile autopilot robustness to uncertain aerodynamics," *AIAA Journal of Guidance, Control and Dynamics*, vol. 15, no. 4, pp. 861–870, 1992.

[20] P. Seiler, G. Balas, and A. Packard, "A gain-based lower bound algorithm for real and mixed μ problems," in *Proceedings of the 45th IEEE Conference on Decision and Control*, 2006, pp. 3548–3553.

[21] M. Halton, M. J. Hayes, and P. Iordanov, "State-space μ-analysis for an experimental drive-by-wire vehicle," in *International Journal of Robust and Nonlinear Control*, vol. 18, 2008, pp. 975–992.

[22] G. Balas, J. Doyle, K. Glover, A. Packard, and R. Smith, "Mu-analysis and synthesis toolbox for use with matlab," 2001, user's guide.

[23] D.-W. Gu, P. Petkov, and M. Konstantinov, *Robust Control Design with MATLAB*. Springer-Verlag, 2005.

[24] C. Roos and J.-M. Biannic, "A detailed comparative analysis of all practical algorithms to compute lower bounds on the structured singular value," *Control Engineering Practice*, vol. 44, pp. 219 – 230, 2015.

[25] R. R. E. de Gaston and M. G. Safonov, "Exact calculation of the multiloop stability margin," *IEEE Transactions on Automatic Control*, vol. 33, no. 2, pp. 156–171, 1988.

[26] The MathWorks, "Robust control toolbox, available at http://www.mathworks.com/help/robust/examples/robustness-of-servo-controller-for-dc-motor.html," 2014.

[27] J. Ackermann and W. Sienel, "Robust control for automatic steering," in *Proceedings of the American Control Conference*, 1990, pp. 795–800.

[28] C. Yi and B. L. R. D. Moor, "Robustness analysis and control system design for a hydraulic servo system," *IEEE Transactions on Control Systems Technology*, vol. 2, no. 3, pp. 183–197, 1994.