



HAL
open science

Advanced probabilistic μ -analysis techniques for AOCS validation

Jean-Marc Biannic, Clément Roos, Samir Bennani, Fabrice Boquet, Valentin Preda, Bénédicte Girouart

► **To cite this version:**

Jean-Marc Biannic, Clément Roos, Samir Bennani, Fabrice Boquet, Valentin Preda, et al.. Advanced probabilistic μ -analysis techniques for AOCS validation. European Journal of Control, 2021, 62, pp.120-129. 10.1016/j.ejcon.2021.06.019 . hal-03479082

HAL Id: hal-03479082

<https://hal.science/hal-03479082>

Submitted on 14 Dec 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Advanced probabilistic μ -analysis techniques for AOCS validation*

Jean-Marc Biannic^a, Clément Roos^{a,*}, Samir Bennani^b, Fabrice Boquet^b,
Valentin Preda^b, Bénédicte Girouart^b

^a*Information Processing and Systems Department, ONERA - The French Aerospace
Lab, 31400 Toulouse, France*

^b*European Space Research & Technology Centre, European Space Agency, 2200 AG
Noordwijk, The Netherlands*

Abstract

Monte-Carlo simulations play a key role in the current Attitude and Orbit Control Systems (AOCS) Verification and Validation (V&V) process, but it is generally time-consuming and it may fail in detecting worst-case configurations, especially in the presence of rare events. In such a case, μ -analysis offers a nice alternative, although it cannot measure the probability of occurrence of the identified worst-cases, which can invalidate a control system on the basis of unlikely events. Probabilistic μ -analysis was introduced in this context 20 years ago to bridge the gap between the two techniques, but until recently no practical tools were available. This paper summarizes recent advances on this topic with a particular emphasis on practical applications to space systems. More precisely, the proposed technique is applied to evaluate AOCS controllers in the context of a challenging high accuracy satellite pointing control problem. The way the proposed tools can be integrated into the traditional AOCS V&V process and used to tighten the V&V analysis gap is also highlighted.

Keywords: Verification & Validation, Robust stability and performance,

*This work is the result of the research project ESA/RFP/3-16071/19/NL/CRS/hh

*Corresponding author.

Email addresses: `jean-marc.biannic@onera.fr` (Jean-Marc Biannic),
`clement.roos@onera.fr` (Clément Roos), `samir.bennani@esa.int` (Samir Bennani),
`fabrice.boquet@esa.int` (Fabrice Boquet), `valentin.preda@esa.int` (Valentin
Preda), `benedicte.girouart@esa.int` (Bénédicte Girouart)

1. Introduction

Novel lightweight materials and deployable structures allow to perform an increasingly wide variety of on-orbit services, which results in stronger interactions between the spacecraft flexible structures (solar arrays, robotic arms, antennas, mirrors) and fuel sloshing in the reservoirs. Efficient AOCS must therefore be designed to ensure a high pointing accuracy. This requires a robust control architecture, but also adequate V&V methods to assess the mission risk and check if performance is guaranteed regardless of the uncertainties and external disturbances affecting the system.

Monte-Carlo simulations [1, 2] are the preferred validation means in the space industry. They are able to quantify the probability of sufficiently frequent phenomena, but they are generally time-consuming, provide only soft bounds [3] and may fail in detecting rare but nevertheless critical events. Less expensive deterministic and simulation-free alternatives exist and have reached a good level of maturity, as is the case for μ -analysis [4, 5, 6]. But unlike Monte Carlo simulations, if worst-case scenarios are no longer missed, their probability of occurrence is also not measured, which can invalidate an AOCS on the basis of very rare and therefore extremely unlikely events [7, 8, 9].

Research to fill the gap between these two approaches is still at a very early stage and only few practical tools are available, although this issue was identified 20 years ago by [10]. This is all the more surprising since the validation process currently accounts for up to 80% of the AOCS total development time, and is becoming longer as the space missions become increasingly complex. In this context, this paper builds on the work of [9, 11, 7, 12] on probabilistic μ -analysis and its ambitions are twofold. First, develop new cheap and reliable tools to improve the characterization of rare but nonetheless possible events, so as to tighten the aforementioned V&V analysis gap. Second, apply these tools to a challenging high accuracy satellite pointing control problem, to show how they can be integrated into the traditional AOCS V&V cycle to improve the current industrial standard and fasten the validation process.

The paper is organized as follows. The latest advances in probabilistic μ -analysis are first presented in Section 2. The resulting computational tool is then described in Section 3. It is finally applied in Section 5 to a challenging and realistic AOCS benchmark introduced in Section 4.

2. Latest advances in probabilistic μ

2.1. Problem statement

Let us consider a continuous-time uncertain linear time-invariant system (usually including control laws):

$$\begin{cases} \dot{x} = A(\delta)x + B(\delta)u \\ y = C(\delta)x + D(\delta)u \end{cases} \quad (1)$$

where the parametric uncertainties $\delta = (\delta_1, \dots, \delta_N) \in \mathbb{R}^N$ are independent random variables with probability density functions $f = (f_1, \dots, f_N)$. It is assumed that $A(\delta)$, $B(\delta)$, $C(\delta)$, $D(\delta)$ are polynomial or rational functions of the δ_i . As a result, system (1) can be transformed into a Linear Fractional Representation (LFR) as in Figure 1 (right): the uncertainties are separated from the nominal (closed-loop) system $M(s)$ and isolated in a block-diagonal operator $\Delta = \text{diag}(\delta_1 I_{n_1}, \dots, \delta_N I_{n_N})$, where I_{n_i} is the $n_i \times n_i$ identity matrix. Note that this paper focuses on real parametric uncertainties, but as for classical μ -analysis, complex uncertainties and neglected dynamics can be considered as well.

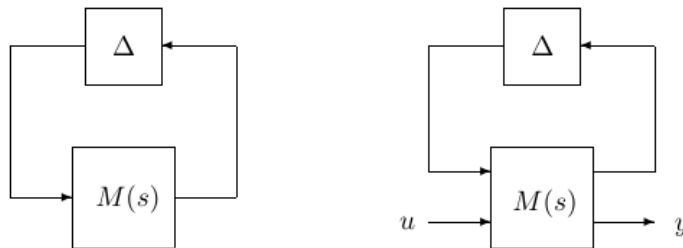


Figure 1: Standard interconnections for robust stability (left) and worst-case performance (right) analysis

The set of matrices with the same block-diagonal structure as Δ is denoted $\mathbf{\Delta}$. It is assumed that the uncertainties are normalized, so that the subset $\mathcal{B}_{\mathbf{\Delta}}$ of $\mathbf{\Delta}$ defined as $\mathcal{B}_{\mathbf{\Delta}} = \{\Delta \in \mathbf{\Delta} : \bar{\sigma}(\Delta) < 1\} = \{\Delta \in \mathbf{\Delta} : |\delta_i| < 1, i \in [1, N]\}$ corresponds to the set of physically meaningful uncertainties $\mathcal{B}_{\delta} = [-1 \ 1]^N$. With these notations in mind, two main problems can be solved using probabilistic μ -analysis:

Problem 2.1 (Probabilistic robust stability). *Compute the probability $\bar{P}_{\Delta, f}(M(s))$ that the interconnection of Figure 1 (left) is unstable when $\Delta \in \mathcal{B}_{\mathbf{\Delta}}$.*

Problem 2.2 (Probabilistic worst-case H_∞ performance). *Given a performance level $\gamma > 0$, compute the probability $\bar{P}_{\Delta,f}^\gamma(M(s))$ that $\|\mathcal{T}_{u \rightarrow y}(s, \Delta)\|_\infty > \gamma$ on Figure 1 (right) when $\Delta \in \mathcal{B}_\Delta$, where $\mathcal{T}_{u \rightarrow y}(s, \Delta)$ is the transfer from u to y .*

Once computed, these probabilities can be confronted to a given tolerance level ϵ , so as to validate or reject the considered control system, depending on whether $\bar{P}_{\Delta,f}(M(s))$ and $\bar{P}_{\Delta,f}^\gamma(M(s))$ are lower or higher than ϵ .

Remark 2.1. *The uncertainties being bounded, their probability distributions must be supported on a bounded interval. Uniform and truncated normal distributions are often used in practice.*

2.2. Probabilistic robust stability

Classical μ -analysis [4, 5] aims at computing the robust stability margin k_r , which satisfies the following properties:

- the interconnection of Figure 1 (left) is stable for all $\Delta \in k_r \mathcal{B}_\Delta = \{\Delta \in \mathbf{\Delta} : |\delta_i| < k_r, i \in [1, N]\}$,
- for all $k > k_r$, there exists at least one $\Delta \in k \mathcal{B}_\Delta$ such that the interconnection is unstable.

This defines an uncertainty box centered at 0 and of radius k_r , which touches the instability domain and where stability is guaranteed. So if $k_r < 1$, there are parts of the uncertainty domain \mathcal{B}_δ that can be stable or unstable, but for which nothing can be concluded.

Let us illustrate this with a simple example extracted from [11]. The considered system is described by the state-space representation:

$$\begin{cases} \dot{x} = \begin{bmatrix} 0 & 1 \\ -a_1(\delta_1) & -a_2(\delta_2) \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \\ y = [1 \ 0] x \end{cases} \quad (2)$$

where $a_1(\delta_1)$, $a_2(\delta_2)$ are two uncertain parameters defined as:

$$\begin{cases} a_1(\delta_1) = 1 + 2\delta_1 \\ a_2(\delta_2) = 0.8 + \delta_2 \end{cases} \quad (3)$$

and δ_1, δ_2 are two normalized real parametric uncertainties. In this particular case, the stability and instability domains can be calculated analytically, using *e.g.* the Routh-Hurwitz stability criterion. They are represented in light/dark green and dark red respectively in Figure 2. The domain $k_r \mathcal{B}_\delta$ where stability is guaranteed by μ -analysis is the light green box, and it is clear that there are both stable and unstable zones outside this area, where no information is available at this stage.

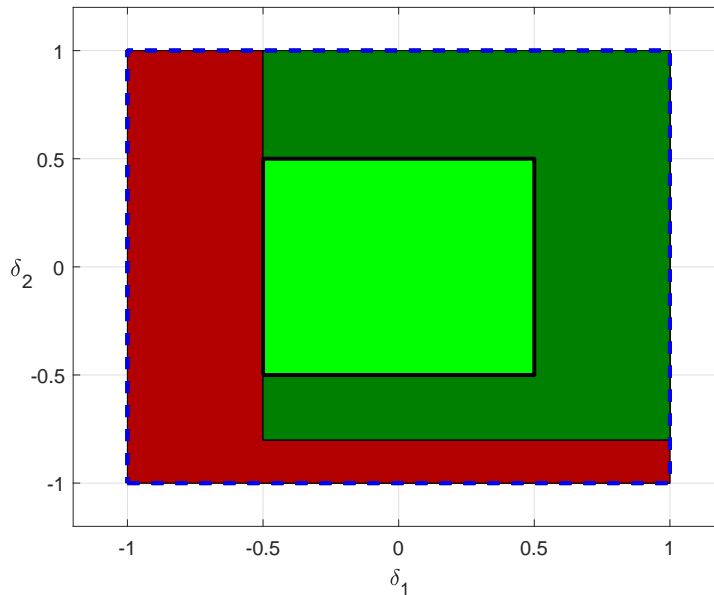


Figure 2: Exact stability domain (light/dark green), exact instability domain (dark red), and guaranteed stability domain obtained with μ -analysis (light green)

Remark 2.2. *Computing k_r is in general NP-hard, so lower and upper bounds are computed instead. Much work has been done to reduce the gap between these bounds, and many efficient algorithms are now available [13]. It can thus usually be assumed that the (almost) exact value of k_r can be computed with a reasonable computational time.*

A branch-and-bound (B&B) algorithm can be used to explore the whole uncertainty domain. The idea is to partition \mathcal{B}_δ into smaller boxes until each

box has guaranteed stability or is sufficiently small to be neglected [14]. Taking the union of the boxes with guaranteed stability leads to an approximation $D_s \subset \mathcal{B}_\delta$ of the exact stability domain. In practice, this approximation is usually quite accurate, as can be seen in the previous example by comparing the green areas in Figures 2 and 3.

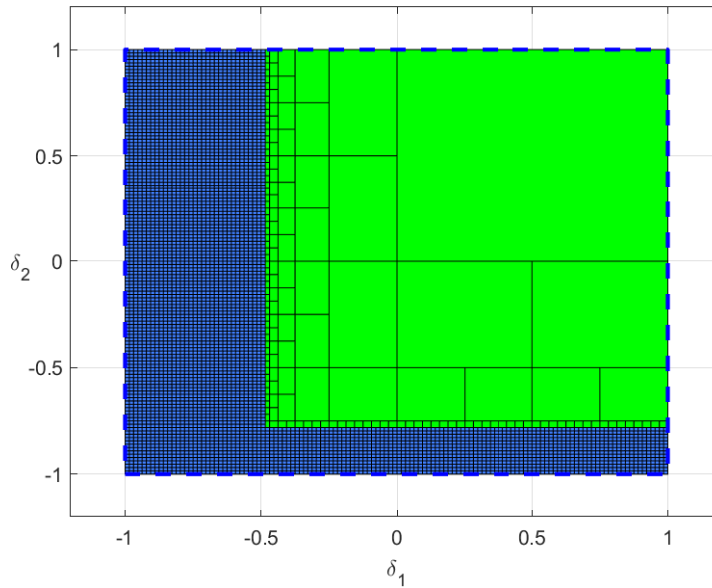


Figure 3: Guaranteed stability domain obtained with μ -analysis and B&B (light green), and domain of undetermined stability (blue)

This strategy can however result in a significant computational time for large-scale problems. In particular, the boxes where the uncertain system is unstable are never identified as such by the algorithm, since μ -analysis requires to start with a nominally stable system. They are partitioned until they reach a negligible size, which leads to an unnecessary generation of boxes. This is materialized in Figure 3 by the use of the blue color to represent these very small undetermined boxes. But this issue can be addressed easily. μ -analysis indeed detects when the poles of the nominally stable system $M(s)$ reach the imaginary axis as the size of Δ increases. The same strategy can be applied to a nominally unstable system. Integrated into the previous B&B scheme, this directly yields a domain $D_{\bar{s}} \subset \mathcal{B}_\delta$ of guaranteed instability.

The domain D_s of guaranteed stability conveniently takes the form of a finite union of disjoint boxes $D_s^{(k)}$:

$$D_s = \bigcup_k D_s^{(k)} \text{ where } D_s^{(k)} = [\underline{\delta}_1^{(k)}, \bar{\delta}_1^{(k)}] \times \cdots \times [\underline{\delta}_N^{(k)}, \bar{\delta}_N^{(k)}]$$

making the associated probability $p(D_s)$ easy to compute:

$$p(D_s) = \sum_k p(D_s^{(k)}) \quad (4)$$

$$= \sum_k \prod_{i=1}^N \int_{\underline{\delta}_i^{(k)}}^{\bar{\delta}_i^{(k)}} f_i(\delta_i) d\delta_i \quad (5)$$

The same applies to $D_{\bar{s}}$, which finally leads to both lower and upper bounds on the exact probability $\bar{P}_{\Delta, f}(M(s))$ of instability, thus solving Problem 2.1:

$$p(D_{\bar{s}}) \leq \bar{P}_{\Delta, f}(M(s)) \leq 1 - p(D_s) \quad (6)$$

The considered control system can then be either validated if $1 - p(D_s) < \epsilon$ or rejected if $p(D_{\bar{s}}) > \epsilon$, where ϵ is the tolerance level introduced in Section 2.1. In practice, B&B is executed until the gap between the bounds becomes small enough and one of the two previous conditions occurs. The uncertainty domain is finally partitioned as follows:

$$\mathcal{B}_\delta = D_s \cup D_{\bar{s}} \cup D_{s_u} \quad (7)$$

where D_{s_u} denotes the domain of undetermined stability, with probability $p(D_{s_u})$. The B&B algorithm can indeed only approximate D_s and $D_{\bar{s}}$, and not compute them exactly. The application to the previous academic example is shown in Figure 4, where it can be seen that very good approximations of the exact stability and instability domains are obtained.

Remark 2.3. *When applying B&B, the boxes can be divided along the direction with the highest μ -sensitivity (i.e. corresponding to the uncertainty with the greatest influence on stability, see [15]), instead of the longest edge as usually done. This significantly reduces the number of boxes and the computational time in most cases [9, 16]. This is observed here by comparing the number of green boxes in Figures 3 and 4, obtained without and with the μ -sensitivities respectively.*

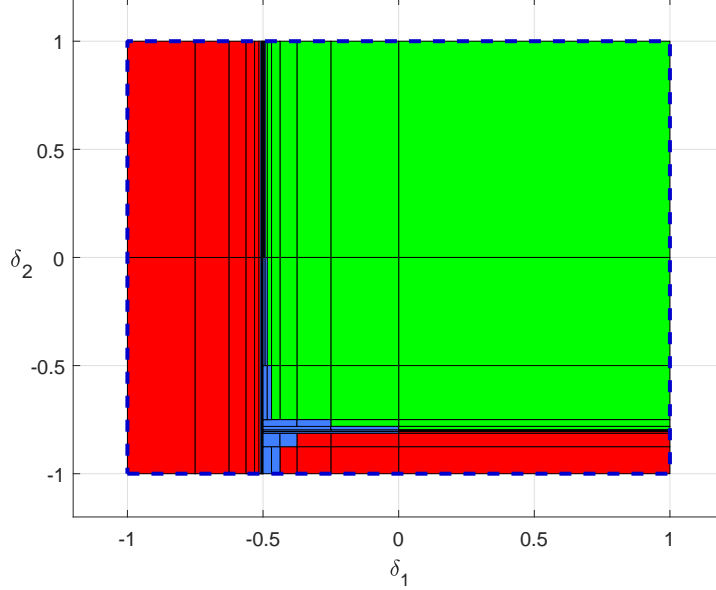


Figure 4: Domains of guaranteed stability D_s (light green), guaranteed instability $D_{\bar{s}}$ (light red) and undetermined stability D_{s_u} (blue) obtained with μ -analysis and B&B

2.3. Probabilistic worst-case H_∞ performance

As with robust stability, B&B is combined with μ -analysis to compute domains of guaranteed performance D_γ and guaranteed non-performance $D_{\bar{\gamma}}$, as well as the associated probability measures $p(D_\gamma)$ and $p(D_{\bar{\gamma}})$. This leads to bounds on the exact probability $\bar{P}_{\Delta,f}^\gamma(M(s))$ of non-performance, thus solving Problem 2.2:

$$p(D_{\bar{\gamma}}) \leq \bar{P}_{\Delta,f}^\gamma(M(s)) \leq p(D_s) - p(D_\gamma) \quad (8)$$

Performance is guaranteed on a given box D , *i.e.* $D \subset D_\gamma$, if:

$$\max_{\Delta \in D} \|\mathcal{T}_{u \rightarrow y}(s, \Delta)\|_\infty \leq \gamma \quad (9)$$

This can be reformulated as a skew μ problem using the main loop theorem [17] and solved using existing μ -based tools [6]. On the other hand, checking if non-performance is guaranteed on D , *i.e.* if $D \subset D_{\bar{\gamma}}$, requires to solve:

$$\min_{\Delta \in D} \|\mathcal{T}_{u \rightarrow y}(s, \Delta)\|_\infty > \gamma \quad (10)$$

This is a minimax problem, which cannot be directly reformulated as a skew- μ problem as above. To address this issue, a sufficient condition for inequality (10) to hold is introduced in [9], in the form of a skew μ calculation involving the inverse transfer $\mathcal{T}_{u \rightarrow y}^{-1}(s, \Delta)$. It is very efficient from a computational point of view, but it may introduce conservatism, although this is usually not the case in practice. The main limitation is that this condition is restricted to a scalar performance channel, *i.e.* $u \in \mathbb{R}$ and $y \in \mathbb{R}$ in (1).

A B&B algorithm can then be implemented as for stability. The investigated domain is limited to the domain of guaranteed stability D_s , since performance analysis only makes sense for stable systems. This leads to:

$$D_s = D_\gamma \cup D_{\bar{\gamma}} \cup D_{\gamma_u} \quad (11)$$

where D_{γ_u} is the domain of undetermined performance. The following partition of \mathcal{B}_δ is finally obtained by combining (7) and (11):

$$\mathcal{B}_\delta = D_\gamma \cup D_{\bar{\gamma}} \cup D_{\gamma_u} \cup D_{\bar{s}} \cup D_{s_u} \quad (12)$$

3. A new computational tool

All the results and algorithms presented in Section 2 have been implemented in the Matlab function `mupb`, the main routine of the new STOWAT¹ (**STO**chastic **W**orst **C**ase **A**nalysis **T**ools) package to be integrated in a forthcoming version of the SMAC (**S**ystems **M**odeling **A**nalysis and **C**ontrol) Toolbox². This function solves Problems 2.1 and 2.2 by computing *guaranteed* lower and upper bounds on the probabilities $\bar{P}_{\Delta, f}(M(s))$ and $\bar{P}_{\Delta, f}^\gamma(M(s))$ with the desired accuracy. It is fully compatible with the Generalized State-Space (GSS) Library of the SMAC Toolbox [18], which offers a powerful and user-friendly way to model LFR, including the ability to define probability distributions for parametric uncertainties. This library is itself compatible with the standard `uss` object of the Robust Control Toolbox, and it provides automated tools to convert `uss` objects to `gss` objects, as well as to incorporate probability distributions initially not present in `uss` objects. Finally, the function `mupb` is also fully interfaced with the Skew Mu Analysis Robustness Tools (SMART) Library of the SMAC Toolbox [6], which allows the user to

¹These tools were developed under ESA contract RFP/3-16071/19/NL/CRS/hh

²The SMAC toolbox is available at `w3.onera.fr/smac`

benefit from a number of state-of-the-art μ -analysis based algorithms. It can be called quite easily as follows:

```
[pbnds,partout]=mupb(sys,pb,partin,options);
```

where:

- `sys` is a `gss` or `uss` object describing the LFR of Figure 1,
- `pb` specifies whether Problem 2.1 or 2.2 is to be solved,
- `partin` can be used to provide an initial description of D_s , $D_{\bar{s}}$, D_{s_u} , D_γ , $D_{\bar{\gamma}}$ and D_{γ_u} coming from a previous call to `mupb`,
- `options` contains tuning parameters related to the stopping criterion of the B&B algorithm, the use of μ -sensitivities and the accuracy of the μ computation,
- `pbnds` gives the guaranteed bounds on $\bar{P}_{\Delta,f}(M(s))$ or $\bar{P}_{\Delta,f}^\gamma(M(s))$,
- `partout` provides the final list of boxes which make up D_s , $D_{\bar{s}}$, D_{s_u} , D_γ , $D_{\bar{\gamma}}$ and D_{γ_u} (same format as `partin`).

4. Benchmark description

As already emphasized in the introduction, pointing performance is increasingly demanding on both scientific and observation space missions of future generations [19]. In the presence of uncertainties, performance degradation remains very challenging to quantify reliably as it results from complex interactions between external perturbations and structural flexible modes of the spacecraft at very specific frequencies. Inspired by previous works presented in [19, 20, 21], the proposed benchmark focuses on the effects on pointing accuracy of the micro-perturbations induced by a Solar Array Drive Mechanism (SADM).

4.1. Model description

The system under consideration, illustrated by Figure 5, is essentially composed of a main body, two solar arrays SA_1 and SA_2 , an isolated payload PL , and a wheel W whose mass and inertia are neglected here.

The single axis case is considered in this work. Let us denote $X = [\theta \ x_{SA_1} \ x_{SA_2} \ x_{PL}]'$ the position vector of the plant, where θ is the pointing error. Using the $M - D - K$ formalism, the model is described as:

$$\begin{bmatrix} \dot{X} \\ \ddot{X} \end{bmatrix} = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}D \end{bmatrix} \begin{bmatrix} X \\ \dot{X} \end{bmatrix} + M^{-1}Z \begin{bmatrix} \Gamma_B \\ \Gamma_{SA} \end{bmatrix} \quad (13)$$

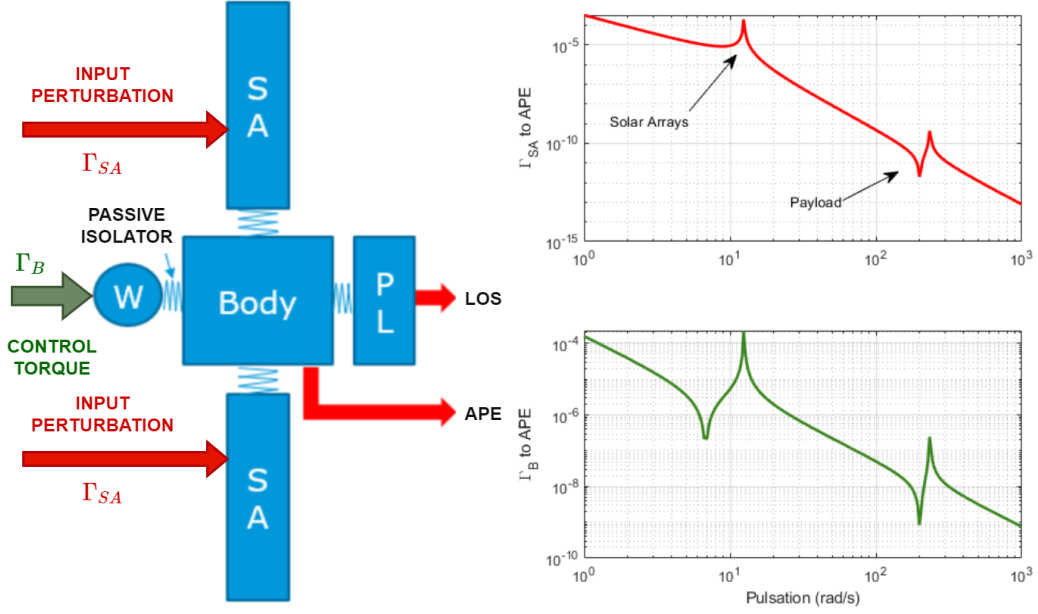


Figure 5: Simplified view of the nominal plant

The generalized inertia, damping and stiffness matrices are defined as:

$$\begin{cases} M = \text{diag}(J_B, J_{SA_1}, J_{SA_2}, J_{PL}) = 10^2 \times \text{diag}(13, 22, 22, 5) \\ D = S(V_D) \\ K = S(V_K) \end{cases} \quad (14)$$

where:

$$S(V) = \begin{bmatrix} \text{sum}(V) & -V' \\ -V & \text{diag}(V) \end{bmatrix} \quad (15)$$

and:

$$\begin{cases} V_D = [D_{SA_1} \ D_{SA_2} \ D_{PL}]' = 10^2 \times [0.5 \ 0.5 \ 8]' \\ V_K = [K_{SA_1} \ K_{SA_2} \ K_{PL}]' = 10^5 \times [1 \ 1 \ 200]' \end{cases} \quad (16)$$

Finally, the input matrix:

$$Z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}' \quad (17)$$

distributes the control torque Γ_B applied to the main body and the input perturbation torques $\Gamma_{SA} = \Gamma_{SA_1} = \Gamma_{SA_2}$ which, for simplicity, are assumed to affect similarly the two solar arrays.

Uncertainties mainly affect the first three elements of the generalized inertia matrix as follows:

$$\begin{cases} J_B &= (1 + 0.1 \delta_0) J_{B_0} \\ J_{SA_1} &= (1 + 0.2 \delta_1) J_{SA_{10}} \\ J_{SA_2} &= (1 + 0.2 \delta_2) J_{SA_{20}} \end{cases} \quad (18)$$

where the δ_i – denoting normalized real parametric uncertainties – respectively introduce 10% variations on the main body and 20% on each solar array. Resulting from these uncertainties, rather significant variations (see Table 1) can be observed on the main flexible modes respectively located near 6.8 rad/s and 12.5 rad/s in the nominal case. Such variations require a robust control design, which is described below.

Element	Damping	Frequency (rad/s)
Mode #1	$[1.6 \ 1.8] \times 10^{-3}$	$[6.35 \ 7.39]$
Mode #2	$[3.0 \ 3.3] \times 10^{-3}$	$[11.9 \ 13.1]$
Mode #3	$[4.7 \ 4.8] \times 10^{-3}$	$[231 \ 238]$

Table 1: Flexible modes characteristics

The particular structure of equation (13) where the uncertainties, through M^{-1} , enter the model in a rational way, strongly suggests to use the LFR framework. Either using `uss` or `gss` objects [18], a minimal LFR $\mathcal{F}_u(G(s), \Delta)$, with $\Delta = \text{diag}(\delta_0, \delta_1, \delta_2)$, is readily obtained and can be integrated in a robust H_∞ control design scheme (see Figure 6). In this scheme, $A(s) = (1 + 0.05s)^{-1}$ denotes the actuator model, while the $W_X(s)$ are the standard input/output weighting functions of the H_∞ framework. Here, the most specific ones are $W_{SA}(s)$ and $W_{APE}(s)$, which shape the disturbance inputs interacting with the flexible modes of the solar arrays and the absolute pointing error (APE) output respectively. Based on previous work introducing relevant metrics for pointing accuracy [22, 23, 24], it appears that the latter can be quantified via the H_∞ norm of a weighted transfer. In this application, the proposed weighting functions, adapted from [21], are all described in Table 2. The main functions of interest (W_{SA} and W_{APE}) have been calibrated in such a way that the pointing performance requirement is met when:

$$\gamma_{ape} = \|\mathcal{T}_{w_{SA} \rightarrow z_{APE}}(s)\|_\infty \leq 1 \quad (19)$$

Note that $W_{SA}(s)$ is a very selective band-pass filter designed to excite the system around 12.45 rad/s , while $W_{APE}(s)$ is a high-pass filter.

Name	Numerical expression	Main characteristic
W_{SA}	$W_{SA}(s) = \left(\frac{0.01245 s}{155 + 0.01245 s + s^2} \right)^2$	selective band-pass
W_{APE}	$W_{APE}(s) = 2 \times 10^5 \times \frac{1+300s}{1+3s}$	high-pass
W_{UI}	$W_{UI}(s) = 10^{-4} \times \frac{1}{1+0.1s}$	low-pass
W_Y	$W_Y(s) = 10^{-7} \times \frac{1}{1+2s}$	low-pass
W_{UO}	$W_{UO}(s) = 10 \times \frac{1+0.002s}{1+0.8s}$	low-pass

Table 2: Weighting functions

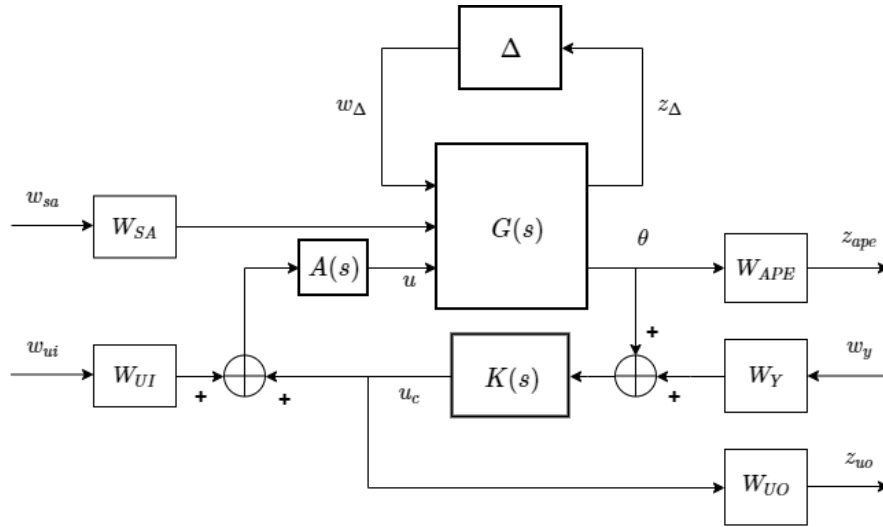


Figure 6: Weighted LFT-based closed-loop model

4.2. Controller design

Control design is performed in two steps. A preliminary PD-like first order controller $K_{PD}(s)$ is initially designed by a modal-based approach that robustly stabilizes the double integrator of the plant for all admissible uncertainties δ_i . In a second step, the poor performance level of this initial controller is improved by a multi-model structured H_∞ design approach based on `systemtune` and a general strategy exposed in [25]. Note however that in the

particular case of this challenging application, the optimization problem is not only nonsmooth but also strongly nonconvex when a low order is required. Then, despite its high efficiency, the optimization algorithm implemented in `systeme` generally fails or requires a large number of random starts. This issue is solved by providing the solver with a non-minimal third-order stabilizing initial condition $K_{INIT}(s)$ directly derived from $K_{PD}(s)$ as follows:

$$K_{INIT}(s) = K_{PD}(s) \times \underbrace{\left(\frac{s^2 + 2\xi_0\omega_0 s + \omega_0^2}{s^2 + 2\xi_0\omega_0 s + \omega_0^2} \right)}_{=1} = K_{PD}(s) \quad (20)$$

where $\omega_0 = 12.5 \text{ rad/s}$ and $\xi_0 = 0.001$ are chosen close to the frequency and damping parameters of the band-pass filter $W_{SA}(s)$. Interestingly, by this approach, a very low (third) order controller is rapidly obtained despite the relatively high order of the weighted design interconnection ($n = 17$).

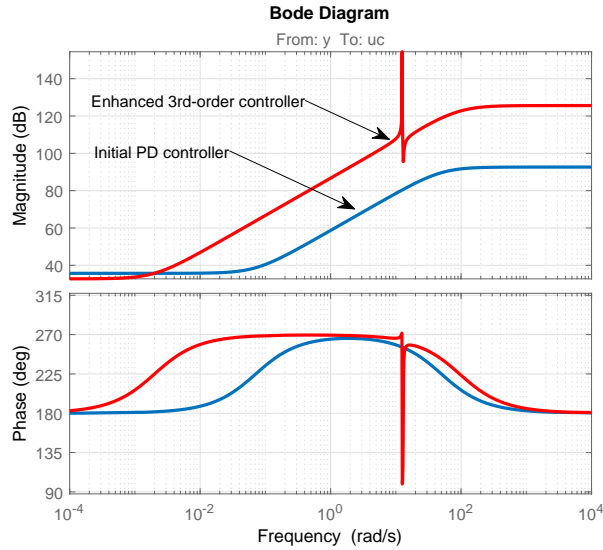


Figure 7: Controllers Bode diagrams

As is visible on the Bode plot of Figure 7, this controller exhibits poorly damped poles and zeros near 12.5 rad/s and 13 rad/s respectively. These are the necessary ingredients to ensure a good rejection of the input perturbations on the solar arrays. However, with such a low-order solution, it was not possible to ensure *a priori* that the performance requirement (19) is rigorously met for all admissible uncertainties.

4.3. Preliminary robustness analysis

A preliminary analysis is realized to evaluate the robustness of the pointing performance. This is first achieved by a standard evaluation of the H_∞ norm of $\mathcal{T}_{w_{SA} \rightarrow z_{APE}}(s)$ for 2000 randomly generated configurations according to a uniform distribution for each uncertainty δ_i . The results are presented in Figure 8 which also displays a worst-case plot detected by a skew μ -analysis approach. At this stage, the controller seems to be validated by the standard Monte-Carlo approach since the H_∞ norm never gets larger than 1, while it is invalidated by μ -analysis.

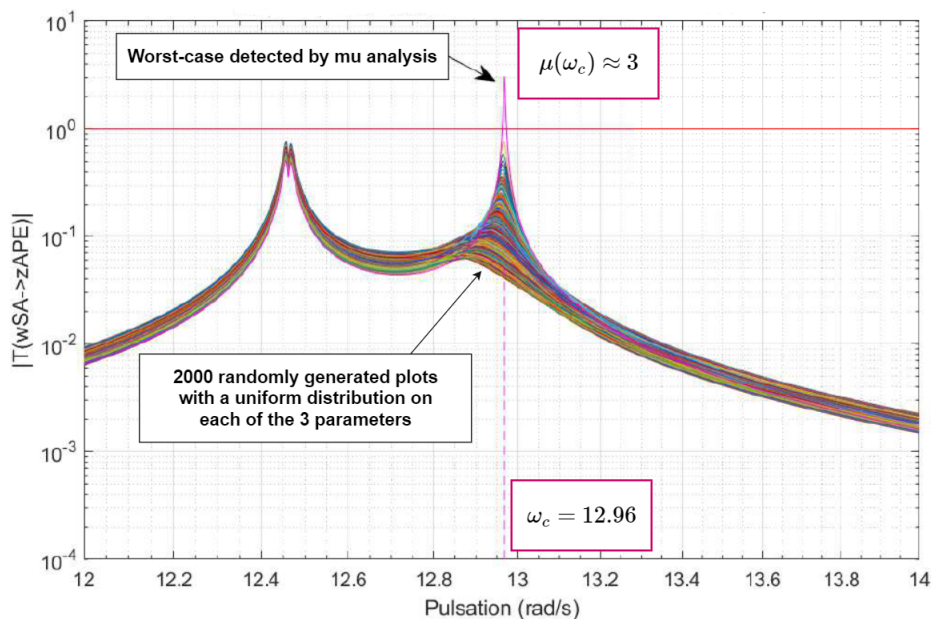


Figure 8: Robust performance: Monte-Carlo vs skew μ -analysis

A refined but also much more time consuming³ analysis involving from $N = 5000$ to 200000 samples is then performed. As can be seen in Table 3, a minimum number of 50000 samples is required to obtain a reasonably tight approximation $\hat{\mathcal{P}}_N(\gamma_{\text{ape}} > \mathbf{1}) \approx 10^{-4}$ of the probability of failure $\mathcal{P}_F = \mathcal{P}(\gamma_{\text{ape}} > \mathbf{1})$, which as already proven by skew μ -analysis is not zero.

³All computations have been performed without parallelization on a standard laptop equipped with a processor Intel i5-8400H, 2.50GHz with 16Gb RAM installed.

Method	# samples (N)	$\hat{\mathcal{P}}_N(\gamma_{\text{ape}} > 1)$	Worst case ($\hat{\gamma}_N$)	CPU
MC	5000	0	0.76	20 s
MC	15000	4.9×10^{-4}	1.00	60 s
MC	25000	2.4×10^{-4}	1.34	100 s
MC	50000	1×10^{-4}	1.76	190 s
MC	100000	9×10^{-5}	1.89	410 s
MC	200000	9.5×10^{-5}	2.23	860 s
skew μ	NA	NA	$\gamma_{\text{ape}}^* = \hat{\gamma}_\infty = \mathbf{3.08}$	6 s

Table 3: Refined Monte-Carlo (MC) robustness analysis

Remark 4.1. As recalled in [26], hard bounds on the probability of failure \mathcal{P}_F cannot be obtained by this approach. However, given tolerance levels ϵ and δ , the **Chernoff bound** enables to predict a minimum number of samples N to ensure a certain accuracy in a probabilistic setting:

$$N > \frac{1}{2\epsilon^2} \log\left(\frac{2}{\delta}\right) \Rightarrow \mathcal{P}\left(|\mathcal{P}_F - \hat{\mathcal{P}}_N| > \epsilon\right) < \delta \quad (21)$$

But this bound is very conservative here. Assuming for example that $\epsilon = \delta = 10^{-4}$, one actually get $N \geq 5 \times 10^8$! A less conservative point of view is proposed in [26] from which, after minor adaptations to our context, one obtains:

$$N > \frac{\log \delta}{\log(1 - \epsilon)} \Rightarrow \mathcal{P}\left(\mathcal{P}(\gamma_{\text{ape}} > \hat{\gamma}_N) > \epsilon\right) < \delta \quad (22)$$

Let us choose $N = 15000$ so that $\hat{\gamma}_N \approx 1$ (see Table 3) and set $\delta = \epsilon$. From (22), it is then readily checked that $\epsilon = (1 - \epsilon)^N \Rightarrow \epsilon \approx 5 \times 10^{-4}$. This can be summarized as:

$$\mathcal{P}\left(\mathcal{P}_F > 5 \times 10^{-4}\right) < 5 \times 10^{-4} \quad (23)$$

Interestingly this result is obtained after "only" 15000 simulations and, as is the case with the Chernoff bound (21), this number of required simulations is **independent of the number of uncertain parameters**. Unfortunately, equation (22) does not permit to evaluate \mathcal{P}_F independently of N which must be fixed here such that $\hat{\gamma}_N \approx 1$. As a result, the tolerance ϵ cannot be freely defined either. However, it provides a good insight into the required number of simulations to be performed, which is confirmed by the numerical experiments summarized in Table 3.

5. Application of probabilistic μ -analysis

In this example involving a limited number of uncertain parameters and *a priori* a very low probability of failure of the control system, Monte-Carlo techniques are time-consuming and certainly not the best suited. On the other hand, μ -analysis can quickly prove non-performance but does not provide any associated risk probability. To fill this gap, the new probabilistic μ tool presented in Section 3 is then used to address Problem 2.2 as detailed in Section 2.3.

5.1. Initial characterization of the performance regions

Considering first uniform distributions for all parametric uncertainties δ_i , the routine `mupb` is called on the uncertain (but robustly stable) closed-loop LFR described in Section 4 to evaluate the performance regions D_γ , $D_{\bar{\gamma}}$ and D_{γ_u} , from which guaranteed probability bounds are deduced. A first analysis is performed with default settings (first row in Table 4). Next, the algorithm is resumed with a refined stopping criterion to obtain tighter (but still guaranteed) bounds on the probability of failure \mathcal{P}_F while keeping a reasonable computational time (second row in Table 4). Note that in both cases, the computed intervals confirm the previous Monte-Carlo analysis (with 50000 samples), which is recalled on the last row of Table 4. Cross-sectional visualizations (with $\delta_0 = 1$) of the performance regions in the plane $\langle \delta_1 \times \delta_2 \rangle$ are then presented in Figures 9 and 10 for both settings.

Method	$\mathcal{P}_F = \mathcal{P}(\gamma_{\text{ape}} > 1)$	CPU time
Probabilistic μ (default)	$[0.53 \ 2.02] \times 10^{-4}$	2 s
Probabilistic μ (refined)	$[0.86 \ 1.47] \times 10^{-4}$	10 s
Monte Carlo (5×10^4 samples)	$\approx 1 \times 10^{-4}$	190 s

Table 4: Guaranteed bounds on \mathcal{P}_F obtained with `mupb` (uniform distribution)

Figure 10 clearly shows that the undetermined region (visualized in blue) is significantly reduced in the second case using some refined settings. How-

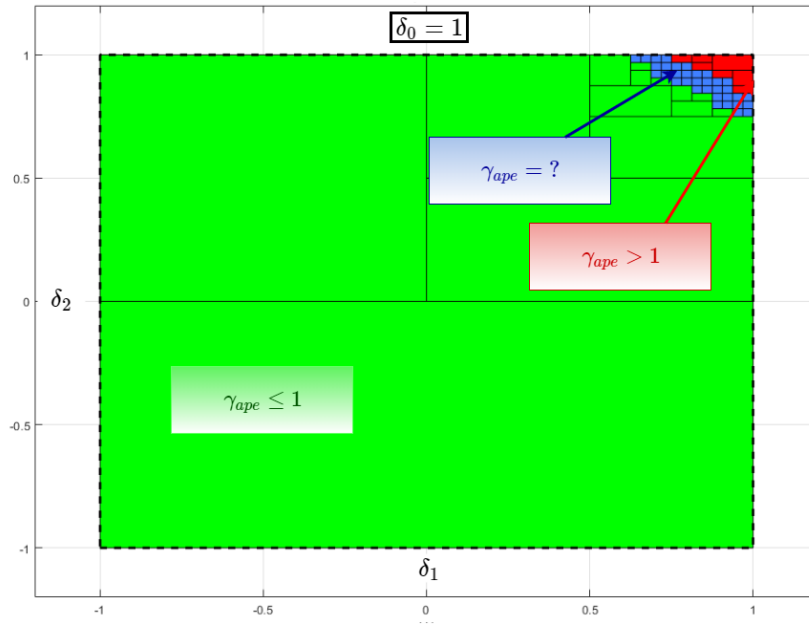


Figure 9: Cross-sectional view ($\delta_0 = 1$) of the performance regions (default settings)

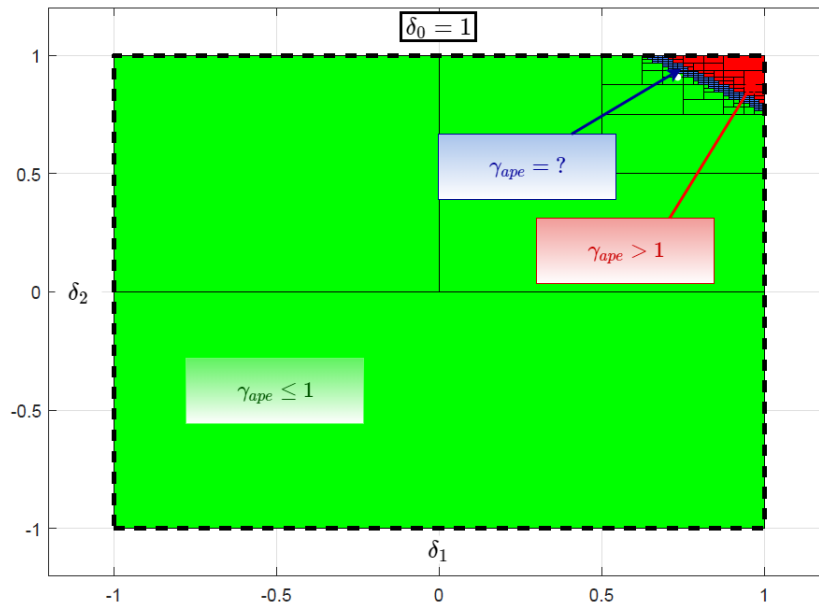


Figure 10: Cross-sectional view ($\delta_0 = 1$) of the performance regions (refined settings)

ever, as is clarified next, this will not systematically have a huge impact on the probability level to be demonstrated.

Remark 5.1. *In this application involving 3 uncertain parameters, the CPU time required by the proposed μ -based approach is very competitive when compared to Monte-Carlo techniques and provides (beyond guaranteed results) much more information. But the computational effort can significantly increase with the number of uncertain parameters, although the latter is not necessarily a good measure of the computational burden. A model with many uncertain parameters, but only a few of which have a large influence on stability or performance, will indeed be easier to analyze than a model with 5 or 10 uncertain parameters of equivalent importance. This is where the μ -sensitivities introduced in Remark 2.3 play a key role. That being said, although there is no general rule as explained above, numerical difficulties are likely to occur beyond 5 to 10 parameters. A promising approach to push this limit is to exploit parallel computing. This is easily achieved for Monte-Carlo techniques, and turns out to be slightly more complex (but feasible!) for the proposed μ -based approach. The proposed algorithm involves a list of domains, each of which can be analyzed independently of the others. We can therefore consider the following implementation: each available CPU first takes the domain with the highest probability from the list, then performs the analysis, and in case (in)stability or (non-)performance cannot be guaranteed finally divides the domain and returns the resulting subdomains to the list. This important issue clearly deserves further investigations and will be the subject of future works.*

5.2. Probability levels cheap reassessment

Beyond the guaranteed results (through hard bounds on the probability of failure) provided by probabilistic μ -analysis, the most interesting feature is that the computed sets of domains (illustrated in Figures 9 and 10) can be reused at a very low cost to evaluate new probability levels for any distribution of the uncertain parameters in their respective intervals. In this application, this is achieved by considering truncated normal distributions with varying standard deviations $\sigma \in [0.3, 5]$ for each uncertainty. The results, obtained in a few seconds without recalling the analysis routine `mupb`, are presented in Figures 11 and 12, which show the evolution of the guaranteed upper and lower bounds on $\mathcal{P}(\gamma_{\text{ape}} > 1)$ as a function of σ . As expected,

the gap between upper and lower bounds is reduced in the refined case (Figure 12). Moreover, a convergence towards the bounds presented in Table 4 (obtained with uniform distributions and visualized by the black solid and dashed lines) is observed when $\sigma \rightarrow \infty$. Conversely, for small values of σ , the probability levels become unsurprisingly so small that their evaluation through Monte-Carlo strategies would require a prohibitive number of simulations. In this respect, the proposed approach clearly bridges a gap between classical methods and μ -analysis. Moreover, the obtained results permit to establish an explicit and constructive relationship between the probability to be demonstrated and the characteristics of the uncertainties. As an example, if the targeted probability of failure is 10^{-6} , then the controller is validated as long as $\sigma \leq 0.44$.

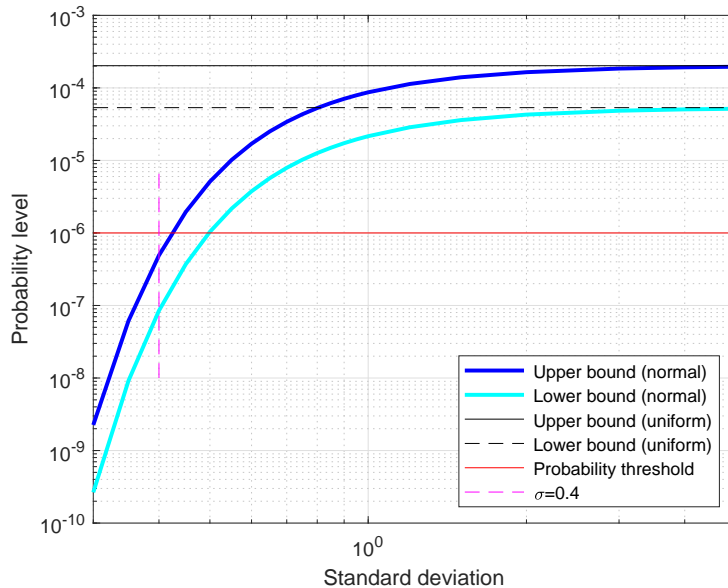


Figure 11: Bounds on $\mathcal{P}(\gamma_{\mathbf{ape}} > \mathbf{1})$ *w.r.t* standard deviation based on the initial set of performance regions (default settings)

Finally, a comparison of the upper bounds (see Figure 13) between the default and refined settings reveals a reasonably small gap – especially for low σ – which might not justify the additional CPU time. The refined analysis has essentially permitted to increase the lower bound on the probability of failure. As a result, for the given probability level to be demonstrated the

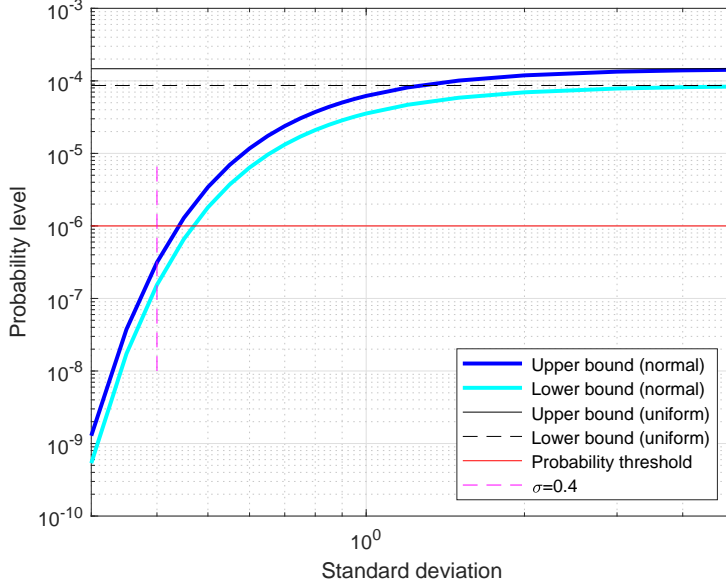


Figure 12: Bounds on $\mathcal{P}(\gamma_{\mathbf{ape}} > \mathbf{1})$ w.r.t standard deviation based on the refined set of performance regions (refined settings)

controller is validated by the default analysis as long as $\sigma \leq 0.42$. The difference between the two results based on default or refined settings is then not really significant. Relevant conclusions can thus be obtained here at a very low cost (a few seconds).

6. CONCLUSIONS AND PERSPECTIVES

Based on recent algorithmic advances in probabilistic μ -analysis and a new Matlab implementation, promising performance assessment results have been obtained on a realistic AOCS benchmark. This opens the way towards the integration of probabilistic μ in a more efficient design-and-validation process. A proposed integration scheme to be further evaluated through future studies is exposed in Figure 14. As emphasized in Section 5.2, a very interesting feature of the proposed approach is to enable very fast iterations between the probability level and the uncertainties characterization. At a medium cost, uncertainty resizing is also possible. Future work will be devoted to more advanced performance evaluation involving several criteria simultaneously such as H_∞ performance on multiple channels but also ro-

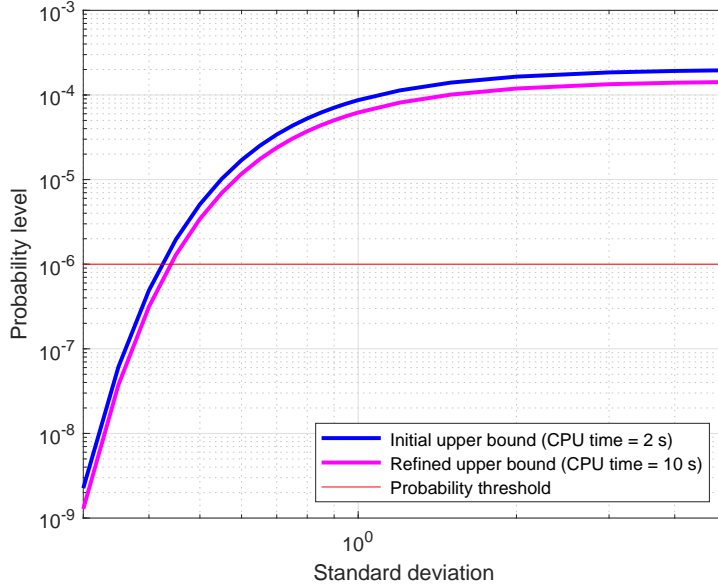


Figure 13: Comparison of upper-bounds on $\mathcal{P}(\gamma_{\mathbf{a}\mathbf{p}\mathbf{e}} > \mathbf{1})$ *w.r.t* standard deviation

bust gain and phase margins. In the longer term, extensions to uncertain time-varying parameters should finally be considered.

References

- [1] J. Helton, J. Johnson, C. Sallaberry, C. Storlie, Survey of sampling-based methods for uncertainty and sensitivity analysis, *Reliability Engineering and System Safety* 91 (2006) 1175–1209.
- [2] D. Landau, K. Binder, *A guide to Monte Carlo simulations in statistical physics*, Cambridge University Press, 2005.
- [3] X. Zhu, Y. Huang, J. Doyle, Soft vs. hard bounds in probabilistic robustness analysis, in: *Proceedings of the IEEE CDC*, 1996, pp. 3412–3417.
- [4] K. Zhou, J. Doyle, K. Glover, *Robust and optimal control*, Prentice-Hall, New Jersey, 1996.
- [5] G. Ferreres, *A practical approach to robustness analysis with aeronautical applications*, Kluwer Academic, 1999.

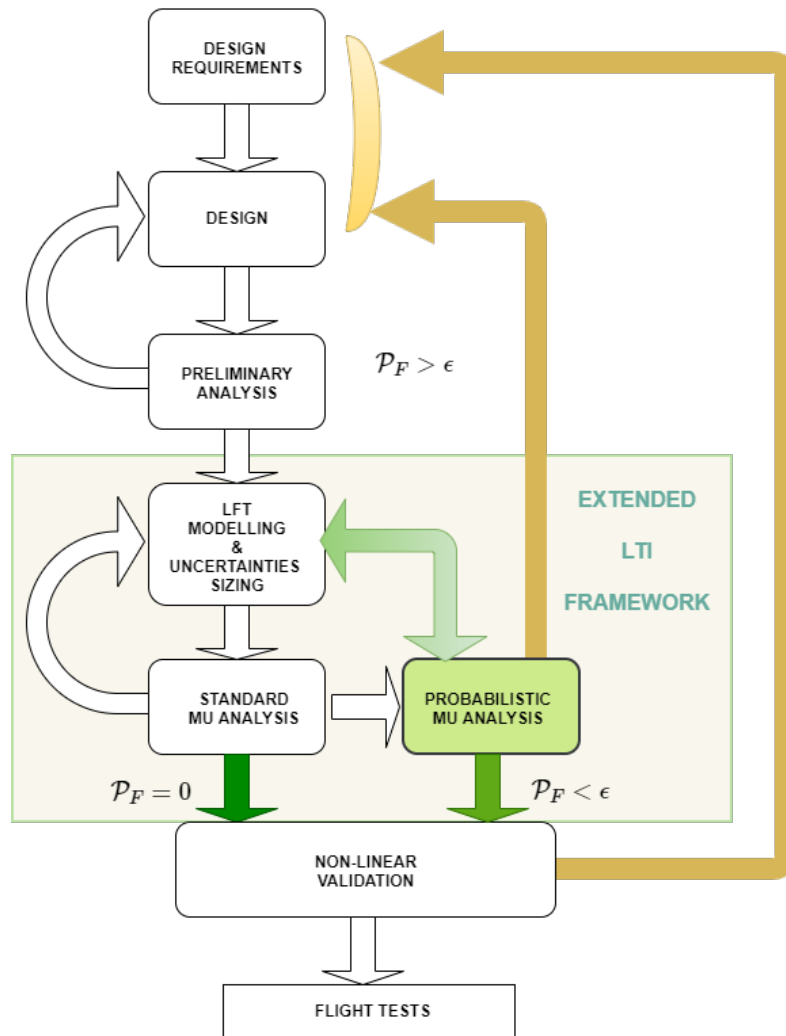


Figure 14: Integration of probabilistic μ -analysis in a design & validation process

- [6] C. Roos, Systems Modeling, Analysis and Control (SMAC) toolbox: an insight into the robustness analysis library, in: Proceedings of the IEEE CACSD Conference, 2013, pp. 176–181, available with the SMAC Toolbox at w3.onera.fr/smac/smart.
- [7] G. Balas, P. Seiler, A. Packard, Analysis of an UAV flight control system using probabilistic μ , in: Proceedings of the AIAA GNC Conference, 2012.

- [8] A. Marcos, S. Bennani, C. Roux, Stochastic μ -analysis for launcher thrust vector control systems, in: Proceedings of the EuroGNC Conference, 2015.
- [9] S. Thai, C. Roos, J.-M. Biannic, Probabilistic μ -analysis for stability and H_∞ performance verification, in: Proceedings of the ACC, 2019, p. 30993104.
- [10] S. Khatri, P. Parrilo, Guaranteed bounds for probabilistic μ , in: Proceedings of the IEEE CDC, 1998, pp. 3349 – 3354. doi:10.1109/CDC.1998.758218.
- [11] A. Falcoz, D. Alazard, C. Pittet, Probabilistic μ -analysis for system performances assessment, IFAC-PapersOnLine 50 (1) (2017) 399 – 404, 20th IFAC World Congress.
- [12] X. Zhu, Improved bounds computation for probabilistic μ , in: Proceedings of the ACC, 2000, pp. 4336–4340.
- [13] C. Roos, J.-M. Biannic, A detailed comparative analysis of all practical algorithms to compute lower bounds on the structured singular value, Control Engineering Practice 44 (2015) 219–230.
- [14] M. Newlin, P. Young, Mixed μ problems and branch and bound techniques, International Journal of Robust and Nonlinear Control 7 (2) (1997) 145–164.
- [15] R. Braatz, M. Morari, μ -sensitivities as an aid for robust identification, in: Proceedings of the ACC, 1991, pp. 231–236.
- [16] J. Lesprier, C. Roos, J.-M. Biannic, Improved μ upper bound computation using the μ -sensitivities, IFAC-PapersOnLine 48 (14) (2015) 215 – 220, 8th IFAC ROCOND Symposium.
- [17] A. Packard, J. Doyle, The complex structured singular value, Automatica 29 (1) (1993) 71–109.
- [18] J.-M. Biannic, C. Roos, Generalized State Space: a new Matlab class to model uncertain and nonlinear systems as LFR, available with the SMAC Toolbox at w3.onera.fr/smac/gss (2012-2020).

- [19] F. Sanfedino, D. Alazard, V. Pommier-Budinger, F. Boquet, A. Falcoz, Dynamic modeling and analysis of micro-vibration jitter of a spacecraft with solar arrays drive mechanism for control purposes, in: Proceedings of the ESA GNC Conference, 2017.
- [20] V. Preda, J. Cieslak, D. Henry, S. Bennani, A. Falcoz, Robust microvibration mitigation and pointing performance analysis for high stability spacecraft, *International Journal of Robust and Nonlinear Control* 28 (18) (2018) 5688–5716. doi:10.1002/rnc.4338.
- [21] F. Boquet, Active solutions to the spacecraft jitter problem, in: *ESA Workshop on Jitter*, 2019.
- [22] M. Pittelkau, Definition, metrics and algorithms for displacement, jitter and stability, Tech. Rep. CP-2003-212246, NASA (2004).
- [23] T. Ott, A. Benoit, P. Van den Braembussche, W. Fichter, ESA pointing error engineering handbook, in: *Proceedings of the ESA GNC Conference*, 2011.
- [24] M. Pittelkau, W. McKinley, Optical transfer functions, weighting functions, and metrics for images with two-dimensional line-of-sight motion, *Optical Engineering* 55 (6) (2016) 1 – 17.
- [25] P. Apkarian, P. Gahinet, C. Buhr, Multi-model, multi-objective tuning of fixed-structure controllers, in: *Proceedings of the ECC*, 2014, pp. 856–861.
- [26] R. Tempo, E. Bai, F. Dabbene, Probabilistic robustness analysis: explicit bounds for the minimum number of samples, in: *Proceedings of the IEEE CDC*, 1996, pp. 3424–3428. doi:10.1109/CDC.1996.573690.