The Scenario Approach to Robust Control Design

G. Calafiore
Dipartimento di Automatica e Informatica
Politecnico di Torino
Corso Duca degli Abruzzi 24, 10129 Torino, Italy
Email: giuseppe.calafiore@polito.it

M.C. Campi
Department of Electrical Engineering and Automation
University of Brescia
Via Branze 38, 25123 Brescia, Italy
Email: campi@ing.unibs.it

March 9, 2004

Abstract. We propose a new probabilistic solution framework for robust control analysis and synthesis problems that can be expressed in the form of minimization of a linear objective subject to convex constraints parameterized by uncertainty terms. This includes for instance the wide class of NP-hard control problems representable by means of parameter-dependent linear matrix inequalities (LMIs).

It is shown in this paper that by appropriate sampling of the constraints one obtains a standard convex optimization problem (the scenario problem) whose solution is *approximately feasible* for the original (usually infinite) set of constraints, i.e. the measure of the set of original constraints that are violated by the scenario solution rapidly decreases to zero as the number of samples is increased. We provide an explicit and efficient bound on the number of samples required to attain a-priori specified levels of probabilistic guarantee of robustness. A rich family of control synthesis problems which are in general hard to solve in a deterministically robust sense is therefore amenable to polynomial-time solution, if robustness is intended in the proposed risk-adjusted sense.

Keywords and phrases: Robust control, Randomized algorithms, Probabilistic robustness, Uncertainty, Robust convex optimization.

1 Introduction

Convex optimization, and semidefinite programming in particular, has become one of the mainstream frameworks for control analysis and synthesis. It is indeed well-known that standard linear control problems such as Lyapunov stability analysis and H_2 or H_∞ synthesis may be formulated (and efficiently solved) in terms of solution of convex optimization problems with linear matrix inequality (LMI) constraints, see for instance [18, 37, 60]. More recently, research in this field has concentrated on considering problems in which the data (for instance, the matrices describing a given plant) are uncertain. A 'guaranteed' (or robust) approach in this case requires the satisfaction of the analysis or synthesis constraints *for all admissible values* of the uncertain parameters that appear in the problem data, see for instance [3]. Therefore, in the 'robustified' version of the problem one has to determine a solution that satisfies a typically infinite number of convex constraints, generated by all the instances of the original constraints, as the uncertain parameters vary over their admissible domains.

This 'robust' convex programming paradigm has emerged around 1998, [10, 40], and, besides the systems and control areas, has found applications in, to mention but a few, truss topology design [9], robust antenna array design, portfolio optimization [41], and robust estimation [39]. Unfortunately, however, robust convex programs are not as easily solvable as standard ones, and are NP-hard in general, [10]. This implies for instance that — unlike standard semidefinite programs (SDP) — simply restating a control problem in the form of a robust SDP does not mean that the problem is amenable to efficient numerical solution.

The current state of the art for attacking robust convex optimization problems is by introducing suitable *relaxations* via 'multipliers' or 'scaling' variables, [12, 40, 48]. The main drawbacks of the relaxation approach are that the extent of the introduced conservatism is in general unknown (except for particular classes of problems, see [12, 42]), and that the method itself can be applied only when the dependence of the data on the uncertainties has a particular and simple functional form, such as affine, polynomial or rational. An extensive discussion of specific robust convex problems arising in control analysis and synthesis, along with the limitations of the solution techniques currently available in the literature is presented in Section 4.

In this paper, we pursue a different 'probabilistic' approach to robustness in control problems, in which the guarantees of performance are not intended in a deterministic sense (satisfaction against all possible uncertainty outcomes) but are instead intended in a probabilistic sense (satisfaction for 'most' of the uncertainty instances, or 'in probability'). This probabilistic approach gained increasing interest in the literature in recent years, and it is now a rather established methodology for robustness *analysis*, see for instance [29, 31, 46, 56, 61]. However, the probabilistic approach has found to date limited application in robust control *synthesis*. Basically, two different methodologies are currently available for probabilistic robust control synthesis: the approach based on the Vapnik-Chervonenkis theory of learning [47, 68, 69], and the sequential methods based on stochastic gradient iterations [28, 36, 51, 54] or ellipsoid iterations, [44].

The first approach is very general but suffers from the conservatism of the Vapnik-Chervonenkis theory, [66, 67], which requires a very large number of randomly generated samples (i.e. it has high 'sample complexity') in order to achieve the desired probabilistic guarantees. Even more importantly, the design methodology proposed in the seminal paper [68] does not aim to enforce the synthesis constraints in a robust sense, but is instead directed towards minimizing the *average* cost objective.

Alternatively, when the original synthesis problem is convex (which includes many, albeit not all, relevant control problems) the sequential approaches based on stochastic gradients [28, 36, 51, 54] or ellipsoid iterations, [44], may be applied with success. However, these methods are currently limited to convex feasibility problems, and have not yet been extended to deal with optimization. More fundamentally, these algorithms have asymptotic nature, i.e. they are guaranteed to converge to a robust feasible solution (if one exists) with probability one, but the total number of uncertainty samples that need to be drawn in order to achieve the desired solution cannot be fixed in advance. We provide a more detailed discussion on these different paradigms for design under uncertainty in Section 2.

The main contribution of the present work is to propose a general framework for solving in a probabilistic sense convex programs affected by uncertainty. The fundamental idea is to consider only a finite number of sampled instances of the uncertainty (the scenarios), and to solve in one-shot the corresponding standard convex problem. We shall prove in this paper that the number of scenarios that need be considered is reasonably small and that the solution of the scenario problem has *generalization* properties, i.e. it satisfies with high probability also unseen scenarios. In other words, the scenario solution is *probabilistically robust*, that is, it satisfies the constraints for all instances, except those in an exceptional set having a known and arbitrarily low level of probability. This is fundamentally different from the average reformulation proposed in [68].

We next show that many control problems (both of analysis and synthesis) for uncertain systems that currently cannot be efficiently solved in a deterministic sense are amenable to efficient solution within the proposed probabilistic paradigm. Using an approach different from the Vapnik-Chervonenkis learning theory, in the key result of this paper (Theorem 1) we provide an efficient bound on the sample complexity of the scenario problem, as a function of the required probabilistic robustness levels. Also, a notable improvement upon the stochastic sequential methods of [28, 36, 44, 54] is that our result holds for robust optimization problems (and not only for feasibility), and that the required number of samples is explicitly bounded by a slowly increasing function of the probabilistic robustness levels. In this respect, we observe that the cited iterative stochastic methods are not guaranteed to converge, unless some strong hypotheses, such as the existence of a deterministically robust solution, are made a-priori.

This paper is organized as follows. In Section 2, we motivate the interest for probabilistic design via a comparative discussion with other design approaches. Section 3 contains the main result on scenario optimization (Theorem 1), whose proof is given in Appendix A. Section 4 presents several relevant robust control problems that are amenable to the scenario-based solution, and in Section 5 we illustrate the theory with various numerical examples. Conclusions are finally drawn in Section 6.

2 Paradigms for Design under Uncertainty

The starting premise to our approach is the observation that a wide variety of robust analysis and synthesis problems in control can be formulated as determining a vector of controller (or more generally 'design') parameters such that some performance specifications on the controlled system are satisfied, as the plant varies over a specified family of admissible plants. More precisely, many robust control problems can be expressed as optimization problems subject to closed-loop constraints that are parameterized by the uncertainties affecting the plant. In formal terms, if $\theta \in \Theta \subseteq \mathbb{R}^{n_\theta}$ is the 'design parameter' (which includes the actual controller parameters, plus possibly other additional variables such as parameters of Lyapunov functions, slack variables and scalings), and the family of admissible plants is parameterized by an 'uncertainty vector' $\delta \in \Delta \subseteq \mathbb{R}^{n_\delta}$, then the prototype control problem we refer to consists of minimizing a linear objective $c^T\theta$ (the objective to be minimized can be taken as linear, without loss of generality), subject to $f(\theta, \delta) \leq 0$, $\delta \in \Delta$, where $f(\theta, \delta) : \Theta \times \Delta \to (-\infty, \infty]$ is a function that specifies the closed-loop constraints. To make things more concrete, consider e.g. a robust H_∞ or H_2 control problem. If the closed-loop system is denoted as $G_{cl}(\xi, \delta)$ (where ξ are design parameters), then we can take $\theta = (\xi, \gamma)$ and minimize γ subject to the constraints $\psi(\xi, \delta) \leq \gamma$, where

$$\psi(\xi, \delta) = \begin{cases} \|G_{cl}(\xi, \delta)\|, & \text{if } G_{cl}(\xi, \delta) \text{ is stable} \\ \infty, & \text{otherwise,} \end{cases} \tag{1}$$

and the norm is either the H_∞ or H_2 norm. Here, $f(\theta, \delta) = \psi(\xi, \delta) - \gamma$, and $c^T\theta = \gamma$.

Notice that in the above abstract design setup we voluntarily left unspecified 'what to do' with the uncertainty, i.e. whether the constraint should hold for one specific value of δ (nominal design), or for all possible values of δ (worst-case design), or whether it should hold in some other, possibly probabilistic, sense. This choice is actually a critical one, and different choices lead to different design paradigms. We next discuss three of these paradigms.

Worst-case design. Worst-case design has been synonymous of robust design in most of the modern control literature. In worst-case design one aims at enforcing the design constraint $f(\theta, \delta) \leq 0$ for all possible values of the uncertainty $\delta \in \Delta$. The resulting closed-loop system will hence exhibit a performance level that is guaranteed for each and every plant in the uncertain family. However, a fundamental problem is encountered along this approach: obtaining worst-case solutions has been proven to be computationally hard; explicit results on the NP-hardness of several worst-case design problems are for instance found in [16, 20, 49]. In addition, a second criticism applies to a worst-case design: seeking guarantees against the worst-case can introduce undesirable *conservatism* in the design, since all the design focus is on a special 'ill' situation (the worst-case plant), which could as well be unrepresentative of the majority of admissible plants. These critical points that we recalled here are extensively documented in the recent work by Vidyasagar [68], to which we refer the reader. In [68], a new approach based on average performance is proposed to overcome the difficulties inherent in the worst-case design. This second design paradigm is discussed next.

Average design. Based on the observation that a good controller should deliver a satisfactory performance for *most* of the plants in the uncertain family, it is proposed in [68] to introduce a probability measure on the uncertainty set Δ (we denote here this probability as 'Prob'), and to seek a design parameter that minimizes the *average performance* value over the uncertainties. In e.g. the H_∞ or H_2 control problems mentioned above, this amounts to minimizing γ subject to the expected value constraint $E_\Delta[\psi(\xi, \delta)] \leq \gamma$. This solution philosophy has the advantage of alleviating the issue of conservatism. Moreover, in [68] a randomized technique is introduced, with the aim of resolving also the issue of numerical complexity. This technique is now briefly described.

We first notice that, in general, even the simple evaluation of the expectation $E_\Delta[\psi(\xi, \delta)]$ for a fixed ξ can be computationally prohibitive. Therefore, the actual expectation is replaced by its sampled empirical version: N independent identically distributed (iid) samples $\delta^{(1)}, \dots, \delta^{(N)}$ are extracted according to probability Prob, and the empirical mean

$$\hat{E}_N(\xi) \doteq \frac{1}{N} \sum_{i=1}^{N} \psi(\xi, \delta^{(i)}) \tag{2}$$

is constructed. The validity of the approximation of the true expectation by its empirical counterpart (2) hinges on a fundamental concept of learning theory (see e.g. [66]) called uniform convergence of empirical means (UCEM), which basically guarantees that, if N is sufficiently large, the error between the true and estimated mean is below a pre-specified level, for all values of ξ in its domain Ξ, provided that the function class $\{\psi(\xi, \cdot), \xi \in \Xi\}$ is not too rich; see [68]. This in turn allows to perform the minimization on $\hat{E}_N(\xi)$ instead of on the true expectation, i.e. the original problem is replaced by the randomized approximation: minimize γ, subject to $\hat{E}_N(\xi) \leq \gamma$. However, even this latter problem is in general difficult to solve, since $\hat{E}_N(\xi)$ can be a non-convex function of ξ. Therefore, a further random search on ξ is proposed in [68] to solve this optimization problem. The overall result of this randomized methodology is to return (with high probability) a so-called *probably approximate near optimal* solution of the original problem.

Besides being a seminal work that brought the learning theory methodology into the robust control realm, the approach of [68] is essentially the only systematic procedure available to tackle very general, non-convex and NP-hard robust control design problems. On the other hand, there are several issues concerning this approach that are debatable, and we discuss them in the following.

First, we observe that for general non-convex problems (i.e. problems in which $\psi(\xi, \delta)$, and hence $\hat{E}_N(\xi)$, is not convex in ξ), the pure random search on ξ is not a very sound minimization technique, and leads to rather weak statements on the quality of the resulting solution. This problem is instead alleviated in the convex case, where in principle the global solution (of the empirical mean minimization problem) can be found efficiently using deterministic methods. Another technical problem is due to the fact that the performance function must be normalized so as to return values in the interval $[0,1]$. This is needed for the applicability of the learning theory results. For example, the performance function (1) cannot be used directly within this approach, but should be replaced by a normalized version

$$\tilde{\psi}(\xi,\delta) = \begin{cases} \frac{\|G_{cl}(\xi,\delta)\|}{1+\|G_{cl}(\xi,\delta)\|}, & \text{if } G_{cl}(\xi,\delta) \text{ is stable} \\ 1, & \text{otherwise.} \end{cases}$$

Therefore, we no longer minimize the actual mean we are interested in; we instead minimize a normalized version of it, and this may carry undesirable effects such as 'levelling out' the contribution of bad events. A further issue is related to the number of samples required to obtain the desired probabilistic levels in the solution (sample complexity). The sample complexity is dictated by the Vapnik-Chervonenkis bound, which involves the determination of an upper bound on the VC-dimension of the family of performance functions. It is well known that the resulting sample complexity can be way too large to be of practical use, see [68].

While the previous issues are somehow technical, the following one pertains instead to the 'philosophical' motivation of the average paradigm: The decision of selecting the average approach leads to a concept of robustness that is inherently different from the one in the worst-case approach. As a matter of fact, the averaging operation returns an index representing a mix of performance evaluations and this is conceptually different from guaranteeing a performance level for each single situation.

Probabilistic robust design. Similar to the average approach, in the probabilistic design paradigm we assume a probability measure Prob over the uncertainty set Δ. Then, for a given probability level $\epsilon \in (0,1)$, we seek a design parameter ξ that minimizes γ while satisfying all constraints but a small fraction of them whose probability is no larger than the prespecified level ϵ. Specifically, we require that

$$\text{Prob}\{\delta \in \Delta : \psi(\xi,\delta) \leq \gamma\} > 1-\epsilon.$$

It should be noted that this approach can be seen as a relaxation of the worst-case paradigm where one allows a risk level ϵ and looks for a design parameter such that the performance specification $\psi(\xi,\delta) \leq \gamma$ is violated by at most a fraction ϵ of the plants in the uncertainty family.

2.1 Previous work along the probabilistic approach and contribution of the present paper.

This paper focuses on the probabilistic design paradigm outlined in the previous section.

While probabilistic techniques have a long history in the optimization area since the early work of Charnes and Cooper for linear programs in 1959, [30], the idea of pursuing probabilistic robustness in control systems is relatively new, but it is gaining increasing interest among control researchers. We direct the reader to the recent monograph [62] for an historical perspective on the topic and for a thorough survey of currently available randomized algorithms for approximately solving probabilistically constrained design problems in control.

However, the randomized approach that we propose in this paper is distinctively different from those discussed in [62] and in other related works such as [28, 36, 44, 51, 54]. These latter references propose sequential stochastic algorithms for determining an approximately feasible design, based on random gradient descent or ellipsoidal iterations. While successful in many practical cases, these techniques present some limitations that are next summarized.

First, the cited iterative methods are all focused on feasibility problems, and have not yet been convincingly extended to deal with constrained minimization. Secondly, for the earliest versions of these algorithms (exposed in [28] for feasibility of uncertain LMIs, and in [54] for guaranteed cost LQR design), convergence to a robustly feasible solution was guaranteed with probability one in a finite number of updates, only if an a-priori hypothesis is made on the distance of the starting solution from a robustly feasible one (which was supposed to exist). This distance is however typically unknown to the designer. Moreover, these algorithms have an 'inner loop' where a randomized test of feasibility is performed on the current solution. Even under the mentioned hypothesis, the number of these inner iterations can be unbounded, see [51]. More recently, a slight modification has been introduced in these algorithms in order to force an exit from the inner loop, see [26, 62, 50]. However, it is necessary to assume deterministic robust feasibility of the problem in order to guarantee that the iterative algorithm returns a probabilistically robust solution, and the existence of such a solution is again typically unknown to the designer. To summarize, the results of these papers are asymptotic in nature, as the total number of iterations is not in general a-priori computable with the available information.

The key to the randomized approach that we propose in this paper is instead to consider a particular optimization problem (the scenario problem) obtained by sampling a suitable number of constraints of the original problem, and to solve it in one-shot via standard interior-point methods for convex programming. The fundamental result given in Theorem 1 provides an explicit and efficient a-priori bound on the sample complexity of the scenario approach, thus showing that this method overcomes the NP-hardness barrier of worst-case design techniques, as well as the mentioned limitations of the iterative stochastic methods for probabilistic design.

The same authors of the present paper have published a previous contribution in [24], dealing with convex optimization within a probabilistic framework. Unlike [24], the present paper focuses on robust control and shows the usefulness of probabilistic optimization in this context. Moreover, the probabilistic analysis of the present paper departs significantly from that of [24] and provides a new and tight sample complexity bound that ouperforms the one in [24].

We conclude this section by stating in formal terms the typical control design problem which is the object of our study.

2.2 A prototype robust control design problem

The design constraints are here expressed by the condition $f(\theta, \delta) \leq 0$, where f is a scalar-valued function. Note that considering scalar-valued constraint functions is without loss of generality, since multiple constraints $f_1(\theta, \delta) \leq 0, \ldots, f_{n_f}(\theta, \delta) \leq 0$ can be reduced to a single scalar-valued constraint by the position $f(\theta, \delta) \doteq \max_{i=1,\ldots,n_f} f_i(\theta, \delta)$. The convexity of f is formally stated in the next assumption.

Assumption 1 (convexity) *Let $\Theta \subseteq \mathbb{R}^{n_\theta}$ be a convex and closed set, and let $\Delta \subseteq \mathbb{R}^{n_\delta}$. We assume that $f(\theta, \delta) : \Theta \times \Delta \to (-\infty, \infty]$ is continuous and convex in θ, for any fixed value of $\delta \in \Delta$.* ⋆

Under Assumption 1, we consider worst-case (or 'robust') control design problems which can be expressed in the form

$$\text{RCP}: \quad \min_{\theta \in \Theta} c^T \theta \quad \text{subject to:} \qquad f(\theta, \delta) \leq 0, \quad \forall \delta \in \Delta. \tag{3}$$

Remark 1 (difficulties in solving RCP) Notice that in the robust convex program RCP, a possibly infinite number of constraints are considered. Despite the convexity assumption, these problems are in general NP-hard, see [10, 40, 12]. ⋆

Remark 2 (RCP structure) Assumption 1 only requires convexity with respect to the design variable θ, while generic non-linear dependence with respect to δ is allowed. Important special cases of robust convex programs are robust linear programs, [11], for which $f(\theta, \delta) = \max_{i=1,\ldots,n_f} f_i(\theta, \delta)$, where each $f_i(\theta, \delta)$ is affine in θ, and robust semidefinite programs, [3, 12, 40], for which $f(\theta, \delta) = \lambda_{\max}[F(\theta, \delta)]$, where

$$F(\theta, \delta) = F_0(\delta) + \sum_{i=1}^{n_\theta} \theta_i F_i(\delta), \quad F_i(\delta) = F_i^T(\delta),$$

and $\lambda_{\max}[\cdot]$ denotes the largest eigenvalue. ⋆

Remark 3 (RCPs in control) A question that arises naturally is how many robust control problems can be expressed in the RCP format we deal with in this paper. Clearly, not all control problems can be expressed in this format. In particular, problems that are nominally non-convex (here 'nominally' means in absence of uncertainty), such as static output feedback, cannot be formulated as RCPs. However, it is well-known that large families of control design problems can be nominally cast in the form of a convex optimization problem, often as semidefinite programs. In this case, adding uncertainty to the nominal problem data usually yields a robust convex problem of the type considered in this paper.

We present in Section 4 a selection of robust analysis and synthesis problems that can be cast in the RCP form and for which no exact deterministic solution algorithm is to date available. These problems include robust analysis and synthesis using parameter-dependent Lyapunov functions, as well as Linear Parameter-Varying (LPV) synthesis and SISO fixed order controller design. Other problems that are readily amenable to our solution framework are mentioned in Section 4.5. ⋆

3 Scenario Optimization and Probabilistic Robustness

In this section, we concentrate on solving the probabilistic robust design problem. The reason for considering this probabilistic relaxation instead of the original worst-case design problem (3) is that problem (3) does not admit a numerically viable solution methodology in general. The main result of this section (Theorem 1) shows instead that a solution to the probabilistic design problem can be found at low computational effort, with complete generality.

One aspect that needs be remarked is that the probabilistic level ϵ is chosen by the designer. By selecting ϵ to be small, a design with characteristics similar to the worst-case design is obtained, while, by increasing ϵ, a lower degree of robustness is accepted. The actual choice of the risk level ϵ would then depend on the application at hand, and could for instance be 10^{-6} if we are dealing with an airliner, or 0.01 if we are dealing with a washing machine.

Let us first specify more precisely our probabilistic setup. We assume that the support Δ for δ is endowed with a σ-algebra $\mathcal{D}$ and that Prob is defined over $\mathcal{D}$. Moreover, we assume that $\{\delta \in \Delta : f(\theta, \delta) \leq 0\} \in \mathcal{D}$, $\forall \theta \in \Theta$. Depending on the situation at hand, Prob can have different interpretations. Sometimes, it is the actual probability with which the uncertainty parameter δ takes on value in Δ. Other times, Prob simply describes the relative importance we attribute to different uncertainty instances. We have the following definition.

Definition 1 (violation probability) *Let $\theta \in \Theta$ be given. The* probability of violation *of θ is defined as*

$$V(\theta) \doteq \text{Prob}\{\delta \in \Delta : \ f(\theta, \delta) > 0\}.$$

$\star$

For example, if a uniform (with respect to Lebesgue measure) probability density is assumed, then $V(\theta)$ measures the volume of 'bad' parameters δ such that the constraint $f(\theta, \delta) \leq 0$ is violated. Clearly, a solution θ with small associated $V(\theta)$ is feasible for most of the problem instances, i.e. it is *approximately feasible* for the robust problem. This concept of approximate feasibility has been introduced in the context of robust control in [7]. Any solution θ such that $V(\theta) \leq \epsilon$ is here named an 'ϵ-level' solution:

Definition 2 (ϵ-level solution) *Let $\epsilon \in (0, 1)$. We say that $\theta \in \Theta$ is an ϵ-level robustly feasible (or, more simply, an ϵ-level) solution, if $V(\theta) \leq \epsilon$.* $\star$

Our goal is to devise an algorithm that returns a ϵ-level solution, where ϵ is any fixed small level, that is worst-case optimal over the set of satisfied constraints. To this purpose, we now introduce the 'scenario' version of the robust design problem. By scenario it is here meant any realization or instance of the uncertainty parameter δ. In the scenario design we optimize the objective subject to a finite number of randomly selected scenarios.

Definition 3 (scenario design) *Assume that N independent identically distributed samples $\delta^{(1)}, \ldots, \delta^{(N)}$ are drawn according to probability* Prob. *A scenario design problem is given by the convex program*

$$\text{RCP}_N: \quad \min_{\theta \in \Theta} c^T\theta \quad \text{subject to:} \qquad (4)$$
$$f(\theta, \delta^{(i)}) \leq 0, \; i = 1, \ldots, N.$$

⋆

We make the following technical assumption on the scenario optimization problem.

Assumption 2 *For all possible extractions $\delta^{(1)}, \ldots, \delta^{(N)}$, the optimization problem (4) is either unfeasible, or, if feasible, it attains a unique optimal solution.* ⋆

This assumption is made to avoid mathematical cluttering. In fact, assuming the most general situation allowing for possible non-uniqueness and/or non-existence of the solution (i.e. the problem is feasible, but the solution escapes to infinity) would lead to mathematical complications that would obscure the clarity of presentation. We remark however that the results in this section extend in a natural way to the most general situation, and the interested reader can find these extended results in Appendix C.

Remark 4 Contrary to problem RCP, the scenario problem RCP_N is a standard convex optimization problem with a finite number N of constraints, and hence its optimal solution $\hat{\theta}_N$ is in general efficiently computable by means of numerical algorithms. Moreover, since only N constraints are imposed in RCP_N, it is clear that the optimal solution of RCP_N is superoptimal for RCP, i.e. the objective corresponding to $\hat{\theta}_N$ outperforms the one achieved with the solution of RCP. In this way, the scenario approach alleviates the conservativeness of the worst-case approach. The price we pay is that the obtained solution can possibly be unfeasible for a small fraction of the plants, see Theorem 1 below. ⋆

The fundamental question that need now be addressed is: what guarantee can be provided on the level of feasibility of the solution $\hat{\theta}_N$ of RCP_N? The following key Theorem 1 answers this question.

Before stating the theorem, we note that, since the constraints $f(\theta, \delta^{(i)}) \leq 0$ are randomly selected, the resulting optimal solution $\hat{\theta}_N$ is a random variable that depends on the multi-sample extraction $(\delta^{(1)}, \ldots, \delta^{(N)})$. Therefore, $\hat{\theta}_N$ can be a ϵ-level solution for a given random extraction and not for another. In the theorem, the parameter β bounds the probability that $\hat{\theta}_N$ is not a ϵ-level solution. Thus, β is the *risk of failure*, or *confidence*, associated to the randomized solution algorithm.

Theorem 1 *Fix two real numbers $\epsilon \in (0, 1)$ (level parameter) and $\beta \in (0, 1)$ (confidence parameter). If*

$$N \geq N(\epsilon, \beta) \doteq \frac{2}{\epsilon} \ln \frac{1}{\beta} + 2n_\theta + \frac{2n_\theta}{\epsilon} \ln \frac{2}{\epsilon}, \qquad (5)$$

then, with probability no smaller than $1 - \beta$, one of the following two facts happen: i) the scenario problem RCP_N *is unfeasible, and then RCP is unfeasible; or, ii)* RCP_N *is feasible, and then its optimal solution $\hat{\theta}_N$ is ϵ-level robustly feasible.* ⋆

In the theorem, probability $1-\beta$ refers to the probability Prob^N (= Prob $\times \cdots \times$ Prob, N times) of extracting a 'bad' multisample, i.e. a multisample $\delta^{(1)}, \ldots, \delta^{(N)}$ such that $\hat{\theta}_N$ is found (i.e. the problem is feasible) and it does not meet the ϵ-level feasibility property.

The proof of Theorem 1 is postponed to Appendix A to avoid breaking the continuity of discourse.

A number of remarks are in order.

Remark 5 (the role of convexity) Theorem 1 says that if we extract a *finite* number N of constraints, then the solution of the randomized problem – if feasible – satisfies most of the other unseen constraints. This is a *generalization* property in the learning theoretic sense: the explicit satisfaction of some 'training' scenarios generalizes automatically to the satisfaction of other unseen scenarios.

It is interesting to note that generalization calls for some kind of structure, and *the only structure used here is convexity.* So, convexity in the scenario approach is fundamental in two different respects: on the computational side, it allows for an efficient solution of the ensuing optimization problem; on the theoretical side, it allows for generalization. ⋆

Remark 6 (sample complexity) Formula (5) provides a ready-to-use 'sample complexity' (i.e. the number N of random scenarios that need to be drawn in order to achieve the desired probabilistic level in the solution) of the scenario problem. In fact $N(\epsilon,\beta)$ in (5) only depends on the number n_θ of optimization variables, besides the probabilistic levels ϵ and β, and its evaluation does not involve computing complicated complexity measures such as the VC-dimension.

$N(\epsilon,\beta)$ exhibits a substantially linear scaling with $\frac{1}{\epsilon}$ and a logarithmic scaling with $\frac{1}{\beta}$. If e.g. $\epsilon = 0.01$, $\beta = 0.0001$ and $n_\theta = 10$, we have $N \geq 12459$. While theory requires the introduction of parameter β, for practical use β plays a very marginal role as, due to the logarithmic dependence, it can be selected to be very small without significantly increasing N.

In general, for reasonable probabilistic levels, the required number of scenarios appears to be manageable by current convex optimization numerical solvers. We also mention that the reader can find a better bound than (5) in the Appendix (equation (22)): in Theorem 1 we have been well advised to provide bound (5) – derived from the bound in the Appendix – to improve readability.

Notice further that extracting δ samples according to a given probability measure Prob is not always a simple task to accomplish, see [29, 25] for a discussion of this topic and polynomial-time algorithms for the sample generation in some matrix norm-bounded sets of common use in robust control. ⋆

Remark 7 (VC-dimension) Bound (5) depends on the problem structure through n_θ, the number of optimization variables. It is not difficult to conceive situations where the class of sets $\{\delta \in \Delta : f(\theta,\delta) > 0\} \subseteq \Delta$, parameterized in θ, has infinite VC-dimension (see e.g. [66] for a definition of VC-dimension), even for small n_θ. Then, estimating $\text{Prob}\{\delta \in \Delta : f(\theta,\delta) > 0\} = V(\theta)$ uniformly with respect to θ is impossible and the VC-theory is of no use. Theorem 1 says that, if attention is restricted to $\hat{\theta}_N$, then estimating $V(\hat{\theta}_N)$ is indeed possible and this can be done at a low computational cost. ⋆

Remark 8 (Prob-independent bound) In some applications, probability Prob is not explicitly known, and the scenarios are directly made available as 'observations'. This could for example be the case when the instances of δ are actually related to various measurements or identification experiments made on a plant at different times and/or different operating conditions, see e.g. [22, 23]. In this connection, notice that the bound (5) is probability independent, i.e. it holds irrespective of the underlying probability Prob, and can therefore be applied even when Prob is unknown.

In other cases, one needs to assume a probability measure in order to perform a scenario design. The choice of this measure is typically dictated by a-priori knowledge about the nature of the uncertainties acting on the plant. Otherwise, if little knowledge is available on the probability distribution of the uncertain parameters, a 'distributionally robust' approach can be taken, as suggested in [6]. In this case, the uniform distribution is typically chosen, due to its worst-case properties, see again [6]. ⋆

Remark 9 (feasibility vs. performance) Solution methodologies for the RCP problem are known only for certain simple dependencies of f on δ, such as affine, polynomial or rational. In all other cases, the scenario approach offers a viable route to find a solution.

Even when solving RCP is possible, resorting to the scenario approach can be advantageous because it alleviates the conservativeness inherent in RCP. In fact, solving RCP gives a 100% deterministic guarantee that the constraints are satisfied. On the other hand, accepting a small risk of constraint violation can result in a (sometimes significant) performance improvement for all plants whose constraints are satisfied. ⋆

Remark 10 (a measurability issue) If RCP_N is always (i.e. for all probabilistic outcomes) feasible, Theorem 1 states that $\mathrm{Prob}^N\{V(\hat{\theta}_N) \leq \epsilon\} \geq 1-\beta$. However, without any further assumption, there is no guarantee that $V(\hat{\theta}_N)$ is measurable, so that $\mathrm{Prob}^N\{V(\hat{\theta}_N) \leq \epsilon\}$ may not be well-defined. Similar subtle measurability issues are often encountered in learning theory, see e.g. [17]. Here and elsewhere, the measurability of $V(\hat{\theta}_N)$, as well as that of other variables defined over Δ^N, is taken as an assumption. ⋆

3.1 The chance-constrained problem

Consider the following problem:

$$\mathrm{CCP}(\epsilon): \quad \min_{\theta\in\Theta} c^T\theta \quad \text{subject to:} \qquad \qquad (6)$$
$$\mathrm{Prob}\{\delta \in \Delta : f(\theta,\delta) \leq 0\} > 1-\epsilon.$$

The distinctive feature of $\mathrm{CCP}(\epsilon)$ is that it is required that the neglected constraint set is chosen in an optimal way, i.e. among all sets of constraints with probability no larger than

ϵ, the removed one is the one that allows for the greatest reduction in the design objective. In the optimization literature, this problem is called 'chance-constrained', see e.g. [55, 65].

It should readily be remarked that an exact numerical solution of CCP(ϵ) is in general hopeless, see [55, 65]. Moreover, CCP(ϵ) is in general non-convex, even when the function $f(\theta, \delta)$ is convex in θ for all $\delta \in \Delta$.

As we have seen, RCP_N returns with high probability a feasible solution of CCP(ϵ). In the next theorem, a comparison between RCP_N and CCP(ϵ) is provided.

Theorem 2 *Let $\epsilon, \beta \in (0, 1)$ be given probability levels. Let $J_{\mathrm{CCP}(\epsilon)}$ denote the optimal objective value of the chance-constrained problem CCP(ϵ) in (6) when it is feasible (i.e. $J_{\mathrm{CCP}(\epsilon)} \doteq \inf_{\theta\in\Theta} c^T\theta$ subject to $V(\theta) \leq \epsilon$) and let $J_{\mathrm{RCP_N}}$ be the optimal objective value of the scenario problem RCP_N in (4) when it is feasible (notice that $J_{\mathrm{RCP_N}}$ is a random variable, while $J_{\mathrm{CCP}(\epsilon)}$ is a deterministic value), with N any number satisfying (5). Then,*

1. *with probability at least $1 - \beta$, if RCP_N is feasible it holds that*

$$J_{\mathrm{RCP_N}} \geq J_{\mathrm{CCP}(\epsilon)}$$

2. *assume CCP(ϵ_1) is feasible, where $\epsilon_1 = 1-(1-\beta)^{1/N}$. With probability at least $1-\beta$, it holds that*

$$J_{\mathrm{RCP_N}} \leq J_{\mathrm{CCP}(\epsilon_1)}.$$

⋆

A proof of this theorem is given in Appendix B.

A few words help clarify result 2 in the theorem. First notice that $J_{\mathrm{CCP}(\epsilon)}$ is a non-increasing function of ϵ. Result 2 states that the optimal value $J_{\mathrm{RCP_N}}$ (where N has been selected so that the optimal solution is ϵ-level feasible with probability $1 - \beta$) is, with probability at least $1 - \beta$, no worse than $J_{\mathrm{CCP}(\epsilon_1)}$, for a certain $\epsilon_1 \leq \epsilon$ explicitly given. For a ready comparison between ϵ and ϵ_1, observe that relation $a^s \leq sa + (1 - s)$ holds for any $a \geq 0$ and $0 \leq s \leq 1$ (as it easily follows by observing that the two sides coincide for $s = 0$ and $s = 1$ and that a^s is convex in s). Then, with the position $a \doteq 1 - \beta; s \doteq 1/N$, we have

$$\epsilon_1 = 1 - (1 - \beta)^{1/N} \geq 1 - \left[\frac{1}{N}(1 - \beta) + \left(1 - \frac{1}{N}\right)\right] = \frac{\beta}{N},$$

which, used in result 2 of the theorem, gives $J_{\mathrm{RCP_N}} \leq J_{\mathrm{CCP}(\beta/N)}$, with N any number satisfying (5). For a crude evaluation, note that if $n_\theta > 1$ and β is not taken to be very small as compared to ϵ, then the dominant term in (5) is $\frac{2n_\theta}{\epsilon} \ln \frac{2}{\epsilon}$, leading to $\epsilon_1 \approx \frac{\beta}{N} \approx \frac{\beta}{2n_\theta \ln \frac{2}{\epsilon}}\epsilon$, where $\frac{\beta}{2n_\theta \ln \frac{2}{\epsilon}}$ is the rescaling factor between ϵ and ϵ_1.

3.2 A-priori and a-posteriori assessments

It is worth noticing that a distinction should be made between the a-priori and a-posteriori assessments that one can make regarding the probability of constraint violation. Indeed, *before* running the optimization, it is guaranteed by Theorem 1 that if $N \geq N(\epsilon, \beta)$ samples are drawn, the solution of the randomized program will be ϵ-level robustly feasible, with probability no smaller than $1 - \beta$. However, the a-priori parameters ϵ, β are generally chosen to be not too small, due to technological limitations on the number of constraints that one specific optimization software can deal with.

On the other hand, once a solution has been computed (and hence $\theta = \hat{\theta}_N$ has been fixed), one can make an a-posteriori assessment of the level of feasibility using standard Monte-Carlo techniques. In this case, a new batch of $\tilde{N}$ independent random samples of $\delta \in \Delta$ is generated, and the *empirical probability* of constraint violation, say $\hat{V}_{\tilde{N}}(\hat{\theta}_N)$, is computed according to the formula $\hat{V}_{\tilde{N}}(\hat{\theta}_N) = \frac{1}{\tilde{N}} \sum_{i=1}^{\tilde{N}} 1(f(\hat{\theta}_N, \delta^{(i)}) > 0)$, where $1(\cdot)$ is the indicator function. Then, the classical Hoeffding's inequality, [43], guarantees that

$$|\hat{V}_{\tilde{N}}(\hat{\theta}_N) - V(\hat{\theta}_N)| \leq \tilde{\epsilon}$$

holds with confidence greater than $1 - \tilde{\beta}$, provided that

$$\tilde{N} \geq \frac{\ln 2/\tilde{\beta}}{2\tilde{\epsilon}^2} \tag{7}$$

test samples are drawn. This latter a-posteriori verification can be easily performed using a very large sample size $\tilde{N}$, because no optimization problem is involved in such an evaluation.

4 Robust Convex Programs in Control

In this section, we discuss several relevant control analysis and synthesis problems that can be naturally cast in the RCP format, and for which no deterministic polynomial-time algorithm is known that computes an exact solution. For these problems, the solution approach that we propose is to first relax the problems in a probabilistic sense and then solve the probabilistic problem via the randomized scenario approach presented in the previous section.

4.1 Stability analysis using parameter-dependent Lyapunov functions

Consider the family of linear systems described in state-space form as

$$\{\dot{x} = A(\delta)x, \quad \delta \in \Delta\}, \tag{8}$$

where $x \in \mathbb{R}^{n_x}$ is the state variable, and the parameter $\delta \in \Delta \subseteq \mathbb{R}^{n_\delta}$ parameterizing the system family is unknown, but constant in time. In the sequel, we shall refer to system families of the type (8) simply as 'uncertain systems'.

Let a symmetric matrix function $P(\xi, \delta)$ be chosen in a family parameterized by a vector of parameters $\xi \in \mathbb{R}^{n_\xi}$, and assume that $P(\xi, \delta)$ is linear in ξ, for all $\delta \in \Delta$. The dependence of $P(\xi, \delta)$ on the uncertainty δ, as well as the dependence of $A(\delta)$ on δ, are otherwise left generic. We introduce the following sufficient condition for robust stability, which follows directly from the standard Lyapunov theory.

Definition 4 (generalized quadratic stability – GQS) *Given a symmetric matrix function $P(\xi, \delta)$, linear in $\xi \in \mathbb{R}^{n_\xi}$ for all $\delta \in \Delta$, the uncertain system (8) is said to be quadratically stable with respect to $P(\xi, \delta)$ if there exists $\xi \in \mathbb{R}^{n_\xi}$ such that*

$$\begin{bmatrix} -P(\xi,\delta) & 0 \\ 0 & A^T(\delta)P(\xi,\delta) + P(\xi,\delta)A(\delta) \end{bmatrix} \prec 0, \quad \forall \delta \in \Delta \tag{9}$$

($\prec$ means negative definite). Such a $P(\xi, \delta)$ is called a Lyapunov matrix for the uncertain system (8). ⋆

For specific choices of the parameterization $P(\xi, \delta)$, the above GQS criterion clearly encompasses the popular quadratic stability (QS, [18, 19]) and affine quadratic stability (AQS, [38]) criteria, as well as the biquadratic stability condition of [63]. For instance, the quadratic stability condition is recovered by choosing $P(\xi, \delta) = P$ (i.e. ξ contains the free elements of $P = P^T$, and there is no dependence on δ), which amounts to determining a *single* Lyapunov matrix P that simultaneously satisfies (9). The AQS condition is instead obtained by choosing

$$P(\xi, \delta) = P_0 + \delta_1 P_1 + \cdots + \delta_{n_\delta} P_{n_\delta}, \tag{10}$$

where ξ represents the free elements in the matrices $P_i = P_i^T$, $i = 0, \ldots, n_\delta$. Notice that QS, AQS and GQS constitute a hierarchy of sufficient conditions for robust stability having decreasing conservatism. However, even the simplest (and most conservative) QS condition is hard to check numerically. Only in the case when the set $\{A(\delta),\ \delta \in \Delta\}$ is a polytope, the QS condition is exactly checkable numerically via convex optimization, [18, 19]. As a matter of fact, in this case a classical vertex result holds which permits to convert the infinite number of constraints entailed by (9) into a finite number of LMIs involving the vertices of the polytope. Notice however that in the classical case when $A(\delta)$ is an interval matrix, the number of vertices of the polytope grows as $2^{n_x^2}$, which means that QS cannot be checked with a computational effort that is polynomial in the problem size n_x.

The AQS condition is computationally hard even in the polytopic case with a fixed number of vertices, and therefore convex relaxations that lead to numerically tractable sufficient conditions for AQS have been proposed in the literature. For instance, in [38] a further multiconvexity requirement is imposed in order to obtain LMI sufficient conditions when Δ is a hypercube and $A(\delta)$ is affine, while in [34] the so-called $\mathcal{S}$-procedure is used for the same purpose. More recently, a generalization of the method, based on a class of Lyapunov functions that depend quadratically (instead of affinely) on δ has been proposed in [63], while the case of linear-fractional (LFT) dependency in $A(\delta)$ is studied in [52]. All these extensions are again particular cases of the GQS criterion defined above.

Now, notice that a key feature of the condition (9) is that, for any *fixed* $\delta \in \Delta$ it represents a convex LMI condition in ξ, and therefore finding a feasible parameter ξ amounts indeed to solving a robust convex program. This is the key observation that makes the scenario paradigm well-suited for probabilistic analysis within the context of generalized quadratic stability. With pre-specified confidence, a matrix $P(\xi, \delta)$ generated by a scenario solution would be a Lyapunov matrix for all but a small fraction of the systems in the family (8).

4.1.1 Formalization as RCP$_N$

Notice that condition (9) is a feasibility condition expressed by a strict matrix inequality, while both problems RCP and RCP$_N$ are minimization problems subject to a non-strict inequality condition (in (3) and (6) we have $f(\theta, \delta) \leq 0$ as opposed to $f(\theta, \delta) < 0$). The precise formalization of the GQS problem within the scenario setting can be done in more than one way and it is to a certain extent a matter of taste. Here, as an illustration, we further develop this first example to indicate a possible way to cast it within the setup of Section 3. It is tacitly understood that similar formalizations apply to all other examples.

First, set an optimization program with the format of (3) as follows:

$$\begin{aligned} \text{RCP}: \quad & \min \alpha \quad \text{subject to:} \\ & -I \preceq \begin{bmatrix} -P(\xi, \delta) & 0 \\ 0 & A^T(\delta)P(\xi, \delta) + P(\xi, \delta)A(\delta) \end{bmatrix} \preceq \alpha I, \quad \forall \delta \in \Delta. \end{aligned}$$

Then, assume a probability measure Prob over the uncertainties is given, and build the scenario counterpart of the problem

$$\begin{aligned} \text{RCP}_N: \quad & \min \alpha \quad \text{subject to:} \\ & -I \preceq \begin{bmatrix} -P(\xi, \delta^{(i)}) & 0 \\ 0 & A^T(\delta^{(i)})P(\xi, \delta^{(i)}) + P(\xi, \delta^{(i)})A(\delta^{(i)}) \end{bmatrix} \preceq \alpha I, \\ & \qquad i = 1, \ldots, N, \end{aligned}$$

where the scenarios $\delta^{(i)}$ are independently extracted at random according to Prob. Here, the optimization variable is $\theta \doteq (\xi, \alpha)$. Note also that the lower bound $-I$ has been introduced without loss of generality since, otherwise, the solution may escape to infinity due to homogeneity of the constraint.

Applying Theorem 1 we can then conclude that, with probability at least $1 - \beta$, either RCP$_N$ is unfeasible, so that RCP and the original GQS is unfeasible, or the solution $(\bar{\xi}, \bar{\alpha})$ of RCP$_N$ is a ϵ-level solution for RCP. In the latter case, if $\bar{\alpha} \geq 0$, it is easily seen that GQS is again unfeasible. Finally, if $\bar{\alpha} < 0$, then $P(\bar{\xi}, \delta)$ is a ϵ-level solution for GQS. Applicability of Theorem 1 subsumes that Assumption 2 holds. If not, the extended results in Appendix C can be applied.

4.2 Generalized quadratic synthesis for uncertain systems

Consider the uncertain system

$$\dot{x} = A(\delta)x + B_1(\delta)w + B_2(\delta)u \quad (11)$$
$$z = C(\delta)x, \quad (12)$$

where $x \in \mathbb{R}^{n_x}$ is the state variable, $w \in \mathbb{R}^{n_w}$ is the exogenous input, $u \in \mathbb{R}^{n_u}$ is the control input, $z \in \mathbb{R}^{n_z}$ is the performance output, and all matrices are generic functions of $\delta \in \Delta$.

4.2.1 State-feedback stabilization

Suppose we want to stabilize (11) by means of a state-feedback control law $u = Kx$, where $K \in \mathbb{R}^{n_u,n_x}$ is a static feedback gain. The resulting closed-loop system is robustly stable if and only if $A_{cl}(\delta) \doteq A(\delta) + B_2(\delta)K$ is Hurwitz for all $\delta \in \Delta$. Using the enhanced LMI characterization proposed in [4] (Theorem 3.1), robust stabilizability of (11) is equivalent to the existence of matrices $V \in \mathbb{R}^{n_x,n_x}$, $R \in \mathbb{R}^{n_u,n_x}$, and a Lyapunov symmetric matrix function $P(\delta) \in \mathbb{R}^{n_x,n_x}$ such that

$$\begin{bmatrix} -(V+V^T) & V^TA^T(\delta) + R^TB_2^T(\delta) + P(\delta) & V^T \\ * & -P(\delta) & 0 \\ * & * & -P(\delta) \end{bmatrix} \prec 0, \quad \forall \delta \in \Delta \quad (13)$$

(asterisks denote entries that are easily inferred from symmetry). If a feasible solution is found, the robustly stabilizing feedback gain is recovered as $K = RV^{-1}$. A sufficient condition for robust stabilizability is hence readily obtained by considering a specific parameterized matrix function family $P(\xi, \delta)$ (linear in the parameter ξ, for any fixed $\delta \in \Delta$) in the above condition. The resulting problem is convex in the decision variable $\theta \doteq (\xi, V, R)$, for any fixed $\delta \in \Delta$, and it is therefore a robust convex problem. Notice again that this robust problem is hard to solve in general. As an exception, in the special case when $[A(\delta)\ B_2(\delta)]$ is affine in δ, Δ is a hypercube, and $P(\delta)$ is chosen in the affine form (10), the above robust condition can be transformed by a standard 'vertexization' argument into a finite set of LMIs involving the vertex matrices, and hence solved exactly (this latter special case is indeed the one presented in reference [4]). We remark however again that the number of vertices (and hence of LMI constraints) grows exponentially with the number of uncertain parameters n_δ, which makes this standard approach practically unviable in cases when n_δ is large.

This robust state-feedback stabilization problem is amenable to the scenario randomization approach similarly to the problem in Section 4.1. A numerical example is presented in Section 5.2.

4.2.2 State-feedback robust H_2 synthesis

For system (11)-(12), consider the problem of designing a state-feedback law $u = Kx$ such that the closed loop is robustly stable, and has guaranteed H_2 performance level γ on the $w - z$ channel.

Adopting the LMI characterization of H_2 performance proposed in [4] (Theorem 3.3), we have that the closed-loop system with controller $K = RV^{-1}$ is robustly stable and has guaranteed H_2 performance less than γ, if there exist $Z = Z^T \in \mathbb{R}^{n_w,n_w}$, $R \in \mathbb{R}^{n_u,n_x}$ and $V \in \mathbb{R}^{n_x,n_x}$ and a Lyapunov symmetric matrix function $P(\delta) \in \mathbb{R}^{n_x,n_x}$ such that

$$\begin{bmatrix} -(V+V^T) & V^TA^T(\delta)+R^TB_2^T(\delta)+P(\delta) & V^TC(\delta) & V^T \\ * & -P(\delta) & 0 & 0 \\ * & * & -\gamma I & 0 \\ * & * & * & -P(\delta) \end{bmatrix} \prec 0, \quad \forall \delta \in \Delta$$

$$\begin{bmatrix} P(\delta) & B_1(\delta) \\ * & Z \end{bmatrix} \succ 0, \quad \mathrm{Tr}\, Z < 1, \quad \forall \delta \in \Delta.$$

Again, we can recast the problem within the randomized setting by considering symmetric parameter-dependent Lyapunov matrix $P(\xi, \delta)$ linear in $\xi \in \mathbb{R}^{n_\xi}$. Notice also that the above matrix inequalities are linear in γ, and therefore the H_2 level can also be minimized subject to these constraints.

4.3 Controller synthesis for LPV systems

Consider a linear parameter-varying (LPV) system of the form

$$\begin{bmatrix} \dot{x} \\ z \\ y \end{bmatrix} \begin{bmatrix} A(\delta(t)) & B_1(\delta(t)) & B_2(\delta(t)) \\ C_1(\delta(t)) & 0 & D_{12}(\delta(t)) \\ C_2(\delta(t)) & D_{21}(\delta(t)) & 0 \end{bmatrix} = \begin{bmatrix} x \\ w \\ u \end{bmatrix},$$

where $x \in \mathbb{R}^{n_x}$ is the state, $w \in \mathbb{R}^{n_w}$ is the exogenous input, $u \in \mathbb{R}^{n_u}$ is the control input, $z \in \mathbb{R}^{n_z}$ is the performance output, $y \in \mathbb{R}^{n_y}$ is the measured output, and $\delta(t) \in \mathbb{R}^{n_\delta}$ is a time-varying parameter, usually referred to as the *scheduling parameter*. In the LPV setting, the parameter $\delta(t)$ is known to be contained in a set Δ, whereas its actual value at time t $\delta(t)$ is a-priori unknown but can be measured online. The LPV formulation has recently received considerable attention, since it forms the basis of systematic gain-scheduling approaches to non-linear control design, see for instance [8, 13, 59, 14] and the survey [58].

The design objective is to determine a controller that processes at time t not only the measured output $y(t)$ but also the measured scheduling parameter $\delta(t)$, in order to determine the control input $u(t)$ for the system.

4.3.1 Quadratic control of LPV systems

Here, we consider a controller of the form

$$\begin{bmatrix} \dot{x}_k \\ u \end{bmatrix} = \begin{bmatrix} A_k(\delta(t)) & B_k(\delta(t)) \\ C_k(\delta(t)) & 0 \end{bmatrix} \begin{bmatrix} x_k \\ y \end{bmatrix}.$$

Suppose the controller has to be designed so that exponential stability is enforced while achieving a quadratic performance specification on the $w - z$ channel. The main difficulty of the problem resides in the fact that in natural applications of the LPV methodology the dependence of the matrices on the scheduling parameter is non-linear. To address this issue, two main approaches have been proposed in the literature. One approach is based on embedding the non-linear dependence into a simpler one (such as affine or linear-fractional), and then reduce the problem to some tractable finite-dimensional convex optimization problem, see for instance [5] and the references therein. Of course, this approach generally involves conservatism in the approximation. A second methodology is instead based on 'gridding' the parameter set, and hence transforming the solvability conditions of the original problem into a finite set of convex constraints, see for instance [2, 8, 70]. The problem with this approach is that the number of grid points (and of constraints, consequently) increases exponentially with the number of scheduling parameters, and may lead to numerically critical implementations. Recently, an alternative randomization-based technique for LPV design has been proposed in [36]. The motivation for this section comes from this latter approach. Indeed, the parameter-dependent inequalities derived in [36] are there solved using sequential stochastic gradient methods, while the same inequalities are here viewed as an instance of a robust convex feasibility problem, and hence directly amenable to the scenario solution proposed in Section 3.

To be specific, let the following (rather standard) assumptions (see [8, 36]) hold:
(i) $\begin{bmatrix} D_{12}^T(\delta(t))C_1(\delta(t)) & D_{12}^T(\delta(t))D_{12}(\delta(t)) \end{bmatrix} = \begin{bmatrix} 0 & I \end{bmatrix}$, $\begin{bmatrix} B_1(\delta(t)) \\ D_{21}(\delta(t))D_{21}^T(\delta(t)) \end{bmatrix} = \begin{bmatrix} 0 \\ I \end{bmatrix}$, hold for all $\delta(t) \in \Delta$;
(ii) $\delta(t)$ is a piecewise continuous function of t, with a finite number of discontinuities in any interval.
Then, formalize the quadratic LPV control problem as follows: Given $\gamma > 0$, find matrices $A_k(\delta(t))$, $B_k(\delta(t))$, $C_k(\delta(t))$ such that the closed-loop system is exponentially stable, and

$$\sup_{w \in L_2 \setminus 0} \frac{\int_0^\infty z^T(t)z(t)dt}{\int_0^\infty w^T(t)w(t)dt} < \gamma,$$

for all $\delta(\cdot)$ such that $\delta(t) \in \Delta$, $\forall t$. The solvability conditions for this problem are directly stated in terms of robust feasibility of three LMIs in [8] (Theorem 4.2). The synthesis LMIs reported below are an equivalent modification of those in [8].

The quadratic LPV L_2 control problem is solvable if and only if there exist $0 \prec P = P^T \in \mathbb{R}^{n_x, n_x}$ and $0 \prec Q = Q^T \in \mathbb{R}^{n_x, n_x}$ such that

$$\begin{bmatrix} A(\delta)P + PA^T(\delta) - \gamma B_2(\delta)B_2^T(\delta) & PC_1^T(\delta) & B_1(\delta) \\ * & -\gamma I & 0 \\ * & * & -I \end{bmatrix} \prec 0, \quad \forall \delta \in \Delta$$

$$\begin{bmatrix} A^T(\delta)Q + QA(\delta) - C_2^T(\delta)C_2(\delta) & QB_1(\delta) & C_1^T(\delta) \\ * & -I & 0 \\ * & * & -\gamma I \end{bmatrix} \prec 0, \quad \forall \delta \in \Delta$$

$$\begin{bmatrix} P & I \\ * & Q \end{bmatrix} \succ 0.$$

Moreover, if feasible $P \succ 0, Q \succ 0$ exist, then the LPV controller matrices are recovered as

$$\begin{aligned} A_k(\delta) &= A(\delta) - Q^{-1}C_2^T(\delta)C_2(\delta) - B_2(\delta)B_2^T(\delta)Z^{-1} + \gamma^{-1}Q^{-1}C_1^T(\delta)C_1(\delta) + \\ &\quad +\gamma^{-1}Q^{-1}(A^T(\delta)Q + QA(\delta) + \gamma^{-1}C_1^T(\delta)C_1(\delta) - C_2^T(\delta)C_2(\delta) + \\ &\quad +QB_1(\delta)B_1^T(\delta)Q)Q^{-1}Z^{-1}, \\ B_k(\delta) &= Q^{-1}C_2^T(\delta), \\ C_k(\delta) &= -B_2^T(\delta)Z^{-1}, \end{aligned}$$

where $Z \doteq (P - Q^{-1})/\gamma$.

Again, this LPV design problem (either finding a feasible design for fixed level γ, or minimizing γ subject to the above constraints) is stated in the form of a RCP, and it is hence amenable to the randomized scenario solution. In this specific context, the scenario approach can be viewed as a kind of gridding technique, where the grid points are randomly selected. The advantage resides in the fact that bound (5) can be used to determine the number of grid points, and this number is independent of the dimension of δ.

4.3.2 State-feedback synthesis for LPV systems

Similar to the approach in the previous section, we here consider a state-feedback design problem for a LPV system with guaranteed decay rate. Consider the LPV system

$$\dot{x} = A(\delta(t))x + B(\delta(t))u$$

and assume that the state is measured, and that the controller is of the form

$$u = K(\delta(t))x,$$

where

$$K(\delta(t)) = K_0 + \sum_{i=1}^{n_\delta} K_i\delta_i(t).$$

The control objective is to determine the matrices K_i, $i = 0, \ldots, n_\delta$, such that the controlled system has a guaranteed exponential decay rate $\nu > 0$. Specifically, defining the closed loop matrix $A_{cl}(\delta(t)) = A(\delta(t)) + B(\delta(t))K(\delta(t))$, the control objective is met if there exists a symmetric matrix $P \succ 0$ such that the matrix inequality

$$A_{cl}(\delta)P + PA_{cl}^T(\delta) + 2\nu P \prec 0 \tag{14}$$

holds for all $\delta \in \Delta$. Introducing the new variables $Y_i \doteq K_iP$, $i = 0, \ldots, n_\delta$, the design requirements are satisfied if

$$\begin{bmatrix} A(\delta)P + B(\delta)Y_0 + \sum_{i=1}^{n_\delta} B(\delta)Y_i\delta_i \\ +PA^T(\delta) + Y_0^TB^T(\delta) + \sum_{i=1}^{n_\delta} Y_i^TB^T(\delta)\delta_i + 2\nu P & 0 \\ 0 & -P \end{bmatrix} \prec 0, \quad \forall\delta \in \Delta.$$

We can apply the scenario solution approach to these conditions, as shown with a numerical example in Section 5.3.

4.4 LP-based robust controller design

Two SISO robust controller design techniques based on (robust) linear programming are reviewed in this section. The first one is a fixed-order controller design technique for discrete-time uncertain systems based on the super-stability and equalized performance concepts introduced in [15] and [53]. The second one is the robust pole assignment method proposed in [45] for SISO continuous-time uncertain plants.

4.4.1 Super-stability and robust ℓ_1 controller design

Let a SISO discrete-time plant be described by the transfer function

$$G(q,\delta) \doteq \frac{b(q,\delta)}{1+a(q,\delta)},$$

where $a(q,\delta), b(q,\delta)$ are polynomials in the unit delay operator q:

$$a(q,\delta) \doteq a_1(\delta)q + \cdots + a_n(\delta)q^n, \quad b(q,\delta) \doteq b_1(\delta)q + \cdots + b_m(\delta)q^m,$$

whose coefficients depend on the uncertain parameter $\delta \in \Delta$ in a generic way. Let $a(\delta), b(\delta)$ denote the vectors of coefficients $a(\delta) \doteq [a_1(\delta) \ \cdots \ a_n(\delta)]^T$, $b(\delta) \doteq [b_1(\delta) \ \cdots \ b_m(\delta)]^T$, and let $\|x\|_1$ denote the usual ℓ_1-norm of vector x (the sum of absolute values of the elements of x). Then, the system is said to be robustly super-stable if $\|a(\delta)\|_1 < 1$, for all $\delta \in \Delta$, see [53]. Super-stability implies BIBO stability, and for super-stable systems the ℓ_1-norm of the system is bounded by (see [15])

$$\|G(q,\delta)\|_1 \leq \frac{\|b(\delta)\|_1}{1-\|a(\delta)\|_1}. \tag{15}$$

Consider now a controller of fixed structure

$$C(q) \doteq \frac{f(q)}{1+g(q)}, \tag{16}$$

with $f(q) \doteq f_0 + f_1 q + \cdots + f_\ell q^\ell$, $g(q) \doteq g_1 q + \cdots + g_p q^p$ to be connected to the plant $G(q,\delta)$ in negative feedback, and let $f \doteq [f_0 \ \cdots \ f_l]^T$, $g \doteq [g_1 \ \cdots \ g_p]^T$. The resulting closed-loop transfer function is

$$G_{cl}(q,\delta) = \frac{h_{cl}(q,\delta)}{1+r_{cl}(q,\delta)},$$

where $h_{cl}(q,\delta) = f(q)b(q,\delta)$, $r_{cl}(q,\delta) = a(q,\delta) + g(q)(1+a(q,\delta)) + f(q)b(q,\delta)$. Notice that the coefficients of the polynomials $h_{cl}(q,\delta)$ and $r_{cl}(q,\delta)$ depend *affinely* on the controller coefficients.

Consider now the following robust ℓ_1-performance problem: determine a controller of structure (16) such that the closed-loop system is robustly BIBO stable, and the ℓ_1-norm of G_{cl} is less than a given $\gamma > 0$, for all $\delta \in \Delta$. Using super-stability as a sufficient condition for BIBO stability, and applying the bound (15) to the closed-loop, we have that the previous synthesis problem is solvable if there exist vectors f, g such that

$$\gamma \|r_{cl}(\delta)\|_1 + \|h_{cl}(\delta)\|_1 \leq \gamma, \quad \forall \delta \in \Delta,$$

where $r_{cl}(\delta)$ and $h_{cl}(\delta)$ are the vectors containing the coefficients of $r_{cl}(q,\delta)$ and $h_{cl}(q,\delta)$. We notice again that it is not known how to solve exactly and efficiently the above problem, except for particularly simple cases such as affine dependence on δ (and Δ being a hypercube), which is the case treated in the cited references. We remark instead that the scenario counterpart of this problem in the general case simply involves a linear program with many constraints, which can be solved with extreme numerical efficiency.

4.4.2 Robust pole assignment

Let a SISO continuous-time plant be described by the proper transfer function

$$G(s,\delta) \doteq \frac{b(s,\delta)}{a(s,\delta)} = \frac{b_0(\delta) + b_1(\delta)s + \cdots + b_m(\delta)s^m}{a_0(\delta) + a_1(\delta)s + \cdots + a_n(\delta)s^n},$$

where the polynomial coefficients depend in a generic non-linear way on the uncertain parameter $\delta \in \Delta$, and are grouped in the vectors $a(\delta), b(\delta)$. Similar to the previous section, consider a control setup with a fixed-structure negative feedback proper controller of degree r

$$C(s) \doteq \frac{f(s)}{g(s)},$$

with numerator and denominator coefficient vectors f, g. Clearly, the closed-loop denominator $d_{cl}(s,\delta) = a(s,\delta)g(s) + b(s,\delta)f(s)$ has a coefficient vector $d_{cl}(\delta)$ which is affine in the controller parameters. The robust control problem discussed in [45] is then of the following type: given a target stable interval polynomial family

$$\mathcal{F} \doteq \{p(s) : p(s) = c_0 + c_1 s + \cdots + c_{n+r}s^{n+r}, \quad c_i \in [c_i^-, c_i^+], \forall i\},$$

determine, if there exist, f, g such that $d_{cl}(f,g,\delta) \in \mathcal{F}$, for all instances of $\delta \in \Delta$. It is then straightforward to see that this problem amounts to checking robust feasibility of a set of linear inequalities in f, g. A specific case of this problem, where the numerator and denominator of G are assumed to be affected by affine interval uncertainty is solved in [45] by reducing it to a standard LP, using again the vertexization argument. In the generic non-linear case, this approach is no longer viable, but we can again solve very efficiently the scenario version of the problem, by means of a standard linear program. A numerical example is given in Section 5.1.

4.5 Other problems

In the previous sections, a few control problems amenable to the scenario reformulation have been illustrated. This is just a sample of possible problems and many more can be considered, such as those presented in [3], as well as problems in the robust model predictive control setup. For discrete-time systems, the robust analysis and design criteria proposed in [32, 33] are also directly suitable for the scenario technique. Also, problems of set-membership state reachability and filtering [21, 27] may be efficiently solved in a probabilistic sense by the proposed methodology.

In the next section, we present various numerical examples of the scenario design approach.

5 Numerical Examples

5.1 Robust pole assignment

As a first numerical test, we adapt from [45] a SISO example of fixed-order robust controller design. Consider the setup introduced in Section 4.4.2, with the plant described by the uncertain transfer function

$$G(s,\delta) = 2(1+\delta_1)\frac{s^2 + 1.5(1+\delta_2)s + 1}{(s-(2+\delta_3))(s+(1+\delta_4))(s+0.236)},$$

where $\delta = [\delta_1\ \delta_2\ \delta_3\ \delta_4]^T$ collects the uncertainty terms affecting the dc-gain, the numerator damping, and the locations of the first two poles of the plant. In this example, we assume

$$\Delta = \{\delta:\ |\delta_1| \leq 0.05, |\delta_2| \leq 0.05, |\delta_3| \leq 0.1, |\delta_4| \leq 0.05\}.$$

The above uncertain plant can be rewritten in the form

$$G(s,\delta) \doteq \frac{b(s,\delta)}{a(s,\delta)} = \frac{b_0(\delta) + b_1(\delta)s + b_2(\delta)s^2}{a_0(\delta) + a_1(\delta)s + a_2(\delta)s^2 + s^3},$$

where

$$\begin{aligned}
b_0(\delta) &= 2(1+\delta_1),\\
b_1(\delta) &= 3(1+\delta_1)(1+\delta_2),\\
b_2(\delta) &= 2(1+\delta_1),\\
a_0(\delta) &= -0.236(2+\delta_3)(1+\delta_4),\\
a_1(\delta) &= -(2+\delta_3)(1+\delta_4) + 0.236(\delta_4-\delta_3) - 0.236,\\
a_2(\delta) &= \delta_4 - \delta_3 - 0.764.
\end{aligned}$$

Define now the following target stable interval polynomial family

$$\mathcal{F} = \{p(s) : p(s) = c_0 + c_1 s + c_2 s^2 + c_3 s^3 + s^4, \quad c_i \in [c_i^-, c_i^+], i = 0,\dots,3\},$$

with

$$c^- \doteq \begin{bmatrix} c_0^- \\ c_1^- \\ c_2^- \\ c_3^- \end{bmatrix} = \begin{bmatrix} 38.25 \\ 57 \\ 31.25 \\ 6 \end{bmatrix}, \quad c^+ \doteq \begin{bmatrix} c_0^+ \\ c_1^+ \\ c_2^+ \\ c_3^+ \end{bmatrix} = \begin{bmatrix} 54.25 \\ 77 \\ 45.25 \\ 14 \end{bmatrix}.$$

The robust synthesis problem we consider is to determine (if one exists) a first order controller

$$C(s) \doteq \frac{f(s)}{g(s)} = \frac{f_0 + f_1 s}{g_0 + s}$$

such that the closed-loop polynomial of the system

$$\begin{aligned} d_{cl}(s,\delta) &= a(s,\delta)g(s) + b(s,\delta)f(s) \\ &= (b_0(\delta)f_0 + a_0(\delta)g_0) + (b_1(\delta)f_0 + b_0(\delta)f_1 + a_1(\delta)g_0 + a_0(\delta))s \\ &\quad +(b_2(\delta)f_0 + b_1(\delta)f_1 + a_2(\delta)g_0 + a_1(\delta))s^2 + (b_2(\delta)f_1 + g_0 + a_2(\delta))s^3 + s^4 \end{aligned}$$

belongs to $\mathcal{F}$, for all $\delta \in \Delta$. Let $\theta \doteq [f_0\ f_1\ g_0]^T \in \mathbb{R}^3$ be the design vector of controller parameters, and define

$$M(\delta) \doteq \begin{bmatrix} b_0(\delta) & 0 & a_0(\delta) \\ b_1(\delta) & b_0(\delta) & a_1(\delta) \\ b_2(\delta) & b_1(\delta) & a_2(\delta) \\ 0 & b_2(\delta) & 1 \end{bmatrix}, \quad p(\delta) \doteq \begin{bmatrix} 0 \\ a_0(\delta) \\ a_1(\delta) \\ a_2(\delta) \end{bmatrix}.$$

Then, the robust synthesis conditions are satisfied if and only if

$$c^- \le M(\delta)\theta + p(\delta) \le c^+, \quad \forall \delta \in \Delta. \tag{17}$$

To the the above robust linear constraints, we also associate a linear objective vector $c^T \doteq [0\ 1\ 0]$ (this amounts to seeking the robustly stabilizing controller having the smallest high-frequency gain), thus obtaining the following robust linear program:

$$\min_\theta\ c^T\theta \quad \text{subject to: (17).} \tag{18}$$

We remark that the solution approach of [45] cannot be directly applied in this case, since the coefficients $a_i(\delta), b_i(\delta)$ do not lie in independent intervals; application of the results in [45] would require overbounding the parameter uncertainty leading to conservative results. We therefore apply the proposed probabilistic solution method: we assume a uniform density over Δ; fixing the risk level parameter to $\epsilon = 0.01$ and the confidence parameter to $\beta = 0.0001$, since $n_\theta = 3$, the sample bound computed according to Theorem 1 (or Theorem 4) is

$$N \ge 5028.$$

Then, the robust problem (18) is substituted by its scenario counterpart

$$\begin{aligned} \min_\theta\ & c^T\theta \quad \text{subject to:} \\ & c^- \le M(\delta^{(i)})\theta + p(\delta^{(i)}) \le c^+, \quad i = 1,\dots,N, \end{aligned} \tag{19}$$

where $\delta^{(1)}, \ldots, \delta^{(N)}$ are independent random samples.

The numerical solution of one instance of the above sampled linear program yielded the solution

$$\hat{\theta}_N = [9.0620 \;\; 19.0908 \;\; 11.6034]^T,$$

and, hence, the controller

$$C(s) = \frac{9.0620 + 19.0908s}{11.6034 + s}.$$

Once we have solved the synthesis problem, we can proceed to an a-posteriori Monte-Carlo test, in order to obtain a more refined estimate $\hat{V}_{\tilde{N}}(\hat{\theta}_N)$ of the probability of constraint violation for the computed solution. As discussed in Section 3.2, we can use a much larger sample size for this a-posteriori analysis, since no numerical optimization is involved in the process. Setting for instance $\tilde{\epsilon} = 0.001$, and $\tilde{\beta} = 0.00001$, from the Chernoff bound (7) we obtain that the test should be run using at least $\tilde{N} = 6.103 \times 10^6$ samples. From Hoeffding inequality, we have that

$$|\hat{V}_{\tilde{N}}(\hat{\theta}_N) - V(\hat{\theta}_N)| \leq 0.001$$

holds with confidence greater than 99.999%.

One run of the Monte Carlo test yielded the estimate

$$\hat{V}_{\tilde{N}}(\hat{\theta}_N) = 0.00076.$$

From a practical point of view, we can thus claim that the above robust controller has violation probability which is at most 0.00176, i.e. it satisfies more than 99.8% of the design constraints (19).

5.2 Robust state-feedback stabilization

We next consider a robust state-feedback stabilization problem of the form presented in Section 4.2.1. In particular, let the uncertain system be given by

$$\dot{x} = A(\delta)x + B_2 u,$$

where $B_2 = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}^T$, and

$$A(\delta) = \begin{bmatrix} 2(1+\delta_3)\Omega & -2-2\delta_1-2(1+\delta_3)\Omega & 2(1+\delta_3)\sin(0.785+\delta_4) & 2+2\delta_1+2(1+\delta_3)\Omega \\ 0 & -2-2\delta_1 & 0 & 4+2\delta_1+2\delta_2 \\ 4(1+\delta_3)\cos(0.785+\delta_4) & -4(1+\delta_3)\cos(0.785+\delta_4) & -2(1+\delta_3)\Omega & 4(1+\delta_3)\Omega \\ 0 & 0 & 0 & 2+2\delta_2 \end{bmatrix},$$

with $\Omega \doteq \cos(0.785 + \delta_4) - \sin(0.785 + \delta_4)$, $\delta \doteq [\delta_1 \; \delta_2 \; \delta_3 \; \delta_4]^T$, and

$$\Delta = \{\delta : \; |\delta_1| \leq 0.2, |\delta_2| \leq 0.2, |\delta_3| \leq 0.2, |\delta_4| \leq 0.2\}.$$

The objective is to determine a state-feedback control law $u = Kx$, such that the resulting closed-loop system is robustly stable.

Using quadratic stability as a sufficient condition for robust stability, from (13) we have that such controller exists if there exist $V \in \mathbb{R}^{4,4}$, $R \in \mathbb{R}^{2,4}$, and a Lyapunov matrix $P = P^T$ such that

$$\begin{bmatrix} -(V+V^T) & V^T A^T(\delta) + R^T B_2^T + P & V^T \\ * & -P & 0 \\ * & * & -P \end{bmatrix} \prec 0$$

is satisfied for all $\delta \in \Delta$. If a feasible solution is found, the feedback gain is recovered as $K = RV^{-1}$.

Similarly to the discussion in Section 4.1.1, we consider the following RCP:

$$\begin{aligned} \min \quad & \alpha \quad \text{subject to:} \\ -I \preceq & \begin{bmatrix} -(V+V^T) & V^T A^T(\delta) + R^T B_2^T + P & V^T \\ * & -P & 0 \\ * & * & -P \end{bmatrix} \preceq \alpha I, \quad \forall \delta \in \Delta, \end{aligned} \tag{20}$$

and, assuming uniform distribution over Δ, we write the scenario counterpart of (20) as

$$\begin{aligned} \min \quad & \alpha \quad \text{subject to:} \\ -I \preceq & \begin{bmatrix} -(V+V^T) & V^T A^T(\delta^{(i)}) + R^T B_2^T + P & V^T \\ * & -P & 0 \\ * & * & -P \end{bmatrix} \preceq \alpha I, \quad i = 1, \ldots, N, \end{aligned}$$

where $\delta^{(i)}$ are independent samples uniformly extracted from Δ. The design variable in this problem is $\theta \doteq (V, R, P, \alpha)$, which contains $n_\theta = 16 + 8 + 10 + 1 = 35$ free variables. Hence, fixing a-priori probabilistic levels $\epsilon = 0.1$ and $\beta = 0.01$, we determine the sample bound according to Theorem 1 (or Theorem 4):

$$N \geq 2260.$$

Notice that, due to current limitations in available numerical solvers for semidefinite programs, we cannot in this case insist on very low a-priori levels. We remark however that this limitation is only of 'technological' nature: one can indeed expect that in few years commercial SDP solvers will be able to deal with a number of constraints in the order of the millions, much like some linear programming solvers can deal with today. Moreover, we will again observe later in this example that the a-posteriori probability levels of the computed solution are significantly better than those guaranteed a-priori.

The numerical solution of one instance of the above scenario problem (using the `mincx` routine in Matlab's LMI-Lab) yielded optimal objective $\alpha = -0.0635$, corresponding to the controller

$$K = \begin{bmatrix} -3.8877 & 1.8740 & -2.6492 & -1.6638 \\ 2.5376 & -1.5973 & 2.4038 & -4.0906 \end{bmatrix}.$$

Next, we run an a-posteriori Monte-Carlo analysis of robust stability with $\tilde{\epsilon} = 0.001$ and $\tilde{\beta} = 0.00001$ ($\tilde{N} = 6.103 \times 10^6$). This a-posteriori test is conducted by testing directly whether $A(\delta) + B_2K$ is Hurwitz for the extracted δ's. Interestingly, for the computed controller K, the a-posteriori estimated probability of instability of $A(\delta) + B_2K$ resulted to be equal to zero, which means for instance that we can claim with confidence greater than 99.999% that our controller may fail at most on a set of δ's having volume 0.001.

5.3 Synthesis of LPV controller

We next present a numerical example of LPV state-feedback stabilization with guaranteed decay rate (see Section 4.3.2).

We consider a multivariable model given in [1] (see also the original paper [64] for a slightly different model and set of data) of the dynamics of the lateral motion of an aircraft. The state space equation is given by

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & L_p & L_\beta & L_r \\ g/V & 0 & Y_\beta & -1 \\ N_{\dot\beta}(g/V) & N_p & N_\beta + N_{\dot\beta} Y_\beta & N_r - N_{\dot\beta} \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 0 & -3.91 \\ 0.035 & 0 \\ -2.53 & 0.31 \end{bmatrix} u,$$

where x_1 is the bank angle, x_2 its derivative, x_3 the sideslip angle, x_4 the yaw rate, u_1 the rudder deflection, u_2 the aileron deflection, and the coefficients in the A matrix have a physical interpretation as discussed in [1] and are subject to time variability.

The following nominal values for the parameters are taken: $L_p = -2.93$, $L_\beta = -4.75$, $L_r = 0.78$, $g/V = 0.086$, $Y_\beta = -0.11$, $N_{\dot\beta} = 0.1$, $N_p = -0.042$, $N_\beta = 2.601$ and $N_r = -0.29$. The actual values of the parameters fluctuate in time with a maximum variation of 15% from the nominal (central) values and are measured on-line.

Setting the desired decay rate to $\nu = 0.5$, and assuming uniform probability distribution over Δ, we applied the proposed scenario approach for the solution of this design problem. Similarly to Section 4.1.1, we introduced the RCP:

$$\min \ \alpha \ \text{ subject to:} \tag{21}$$

$$-I \preceq \begin{bmatrix} \begin{array}{l} A(\delta)P + B(\delta)Y_0 + \sum_{i=1}^{n_\delta} B(\delta)Y_i\delta_i \\ +PA^T(\delta) + Y_0^T B^T(\delta) + \sum_{i=1}^{n_\delta} Y_i^T B^T(\delta)\delta_i + 2\nu P \end{array} & 0 \\ 0 & -P \end{bmatrix} \preceq \alpha I, \quad \forall \delta \in \Delta.$$

and then solved its scenario counterpart with N computed as follows: The number of uncertainty terms is $n_\delta = 9$, so that $n_\theta = 83$. The probability levels were selected to be $\epsilon = 0.1$ and $\beta = 0.01$, yielding $N = 5232$ according to Theorem 1.

The solution of one instance of the scenario problem yielded optimal values $\alpha = -0.2294$, and

$$P = \begin{bmatrix} 0.4345 & -0.3404 & -0.0014 & -0.0061 \\ -0.3404 & 0.7950 & -0.0053 & -0.0007 \\ -0.0014 & -0.0053 & 0.4787 & 0.3604 \\ -0.0061 & -0.0007 & 0.3604 & 0.7507 \end{bmatrix}$$

$$\begin{aligned}
K_0 &= \begin{bmatrix} 2.3689 & 3.0267 & 0.2346 & -0.2593 \\ -0.0268 & -0.0028 & 1.6702 & -2.1804 \end{bmatrix} \times 10^3 \\
K_1 &= \begin{bmatrix} -1.9052 & -2.4343 & -0.1896 & 0.2091 \\ 0.0221 & 0.0021 & -1.3443 & 1.7538 \end{bmatrix} \times 10^4 \\
K_2 &= \begin{bmatrix} 1.5467 & 1.9763 & 0.1539 & -0.1698 \\ -0.0179 & -0.0017 & 1.0914 & -1.4238 \end{bmatrix} \times 10^4 \\
K_3 &= \begin{bmatrix} -1.8403 & -2.3515 & -0.1831 & 0.2020 \\ 0.0213 & 0.0020 & -1.2985 & 1.6944 \end{bmatrix} \times 10^3 \\
K_4 &= \begin{bmatrix} -269.0519 & -343.8210 & -26.8210 & 29.5593 \\ 3.1197 & 0.2849 & -190.1478 & 247.8405 \end{bmatrix} \\
K_5 &= \begin{bmatrix} -325.6902 & -416.1430 & -31.9206 & 35.5400 \\ 3.7771 & 0.3475 & -229.8091 & 299.8153 \end{bmatrix} \\
K_6 &= \begin{bmatrix} -0.7610 & -1.0314 & -0.3023 & 0.4185 \\ -2.1024 & -2.6955 & -0.5855 & 0.7303 \end{bmatrix} \times 10^4
\end{aligned}$$

$$\begin{aligned}
K_7 &= \begin{bmatrix} 8.4788 & 11.2324 & 1.2490 & -0.9159 \\ -0.0983 & -0.0090 & 5.9940 & -7.8125 \end{bmatrix} \\
K_8 &= \begin{bmatrix} -0.8506 & -1.0279 & 0.1419 & -0.2416 \\ 2.1211 & 2.6972 & -0.5517 & 0.7533 \end{bmatrix} \times 10^4 \\
K_9 &= \begin{bmatrix} -1.7472 & -2.2325 & -0.1738 & 0.1922 \\ 0.0203 & 0.0019 & -1.2328 & 1.6084 \end{bmatrix} \times 10^3.
\end{aligned}$$

Different a-posteriori tests can be conducted on the computed solution. For instance, we may estimate by Monte Carlo the probability of violation of the constraint used in problem (21). Using $\tilde{N} = 6.103 \times 10^6$ ($\tilde{\epsilon} = 0.001$, $\tilde{\beta} = 0.00001$) parameter samples, this estimated probability resulted to be equal to 8.65×10^{-5}.

Alternatively (and perhaps more meaningfully), we may test the solution against the original design inequality (14). In this case, using again $\tilde{N} = 6.103 \times 10^6$ parameter samples, we obtained an estimated probability of violating (14) equal to zero, i.e. our design satisfied all the $\tilde{N}$ a-posteriori random tests.

6 Conclusions

This paper presented a novel approach to robust control design, based on the concept of uncertainty scenarios. Within this framework, if the robustness requirements are imposed in a probabilistic sense, then a wide class of control analysis and synthesis problems are amenable to efficient numerical solution. This solution is computed solving a convex optimization problem having a finite number N of sampled constraints. The main contribution of the paper is to provide an explicit and efficient bound on the number of scenarios required to obtain a design that guarantees an a-priori specified probabilistic robustness level.

This methodology is illustrated by several control design examples that present difficulties when tackled by means of standard worst-case techniques. We believe that, due to its intrinsic simplicity, the scenario approach will be an appealing solution technique for many practical engineering design problems, also beyond the control applications mentioned in this paper.

A Technical preliminaries and proof of Theorem 1

A.1 Preliminaries

We first recall a classical result due to Helly, see [57].

Lemma 1 (Helly) *Let $\{\mathcal{X}_i\}_{i=1,\ldots,p}$ be a finite collection of convex sets in $\mathbb{R}^n$. If every subcollection consisting of $n+1$ sets has a non-empty intersection, then the entire collection has a non-empty intersection.* ⋆

Next, we prove a key instrumental result. Consider the convex optimization program

$$\mathcal{P}: \quad \min_{x\in\mathbb{R}^n} c^T x \quad \text{subject to:} \quad x \in \bigcap_{i\in\{1,\ldots,m\}} \mathcal{X}_i,$$

where $\mathcal{X}_i$, $i = 1,\ldots,m$, are closed convex sets, and define the convex programs $\mathcal{P}_k$, $k = 1,\ldots,m$, obtained from $\mathcal{P}$ by removing the k-th constraint:

$$\mathcal{P}_k: \quad \min_{x\in\mathbb{R}^n} c^T x \quad \text{subject to:} \quad x \in \bigcap_{i\in\{1,\ldots,m\}\setminus k} \mathcal{X}_i.$$

Let x^* be any optimal solution of $\mathcal{P}$ (assuming it exists), and let x_k^* be any optimal solution of $\mathcal{P}_k$ (again, assuming it exists). We have the following definition.

Definition 5 (support constraint) *The k-th constraint $\mathcal{X}_k$ is a* support constraint *for $\mathcal{P}$ if problem $\mathcal{P}_k$ has an optimal solution x_k^* such that $c^T x_k^* < c^T x^*$.* ⋆

The following theorem holds.

Theorem 3 *The number of support constraints for problem $\mathcal{P}$ is at most n.* ⋆

A proof of this result was first given by the authors of the present paper in [24]. We here report an alternative and more compact proof based on an idea suggested to us by professor A. Nemirovski in a personal communication.

Proof. Let problem $\mathcal{P}$ have q support constraints $\mathcal{X}_{s_1}, \ldots, \mathcal{X}_{s_q}$, where $\mathcal{S} \doteq \{s_1, \ldots, s_q\}$ is a subset of q indices from $\{1, \ldots, m\}$. We next prove (by contradiction) that $q \leq n$.

Let x^* be any optimal solution of $\mathcal{P}$, with corresponding optimal objective value $J^* = c^T x^*$, and let x_k^* be any optimal solution of $\mathcal{P}_k$, $k = 1, \ldots, m$, with corresponding optimal objective value $J_k^* = c^T x_k^*$. Consider the smallest objective improvement obtained by removing a support constraint

$$\eta_{\min} \doteq \min_{k \in \mathcal{S}} (J^* - J_k^*)$$

and, for some η with $0 < \eta < \eta_{\min}$, define the hyperplane

$$\mathcal{H} \doteq \{x : c^T x = J^* - \eta\}.$$

By construction, the q points x_k^*, $k \in \mathcal{S}$, lie in the half-space $\{x : c^T x < J^* - \eta\}$, while x^* lies in the half-space $\{x : c^T x > J^* - \eta\}$, and therefore $\mathcal{H}$ separates x_k^*, $k \in \mathcal{S}$, from x^*. Next, for all indices $k \in \mathcal{S}$, we denote with $\bar{x}_k^*$ the point of intersection between the line segment $\overline{x_k^* x^*}$ and $\mathcal{H}$.

Since $x_k^* \in \bigcap_{i \in \{1,\ldots,m\} \setminus k} \mathcal{X}_i$, $k \in \mathcal{S}$, and $x^* \in \bigcap_{i \in \{1,\ldots,m\}} \mathcal{X}_i$, then by convexity we have that $\bar{x}_k^* \in \bigcap_{i \in \{1,\ldots,m\} \setminus k} \mathcal{X}_i$, $k \in \mathcal{S}$, and therefore (since, by construction, $\bar{x}_k^* \in \mathcal{H}$)

$$\bar{x}_k^* \in \left(\bigcap_{i \in \{1,\ldots,m\} \setminus k} \mathcal{X}_i \right) \bigcap \mathcal{H}, \quad k \in \mathcal{S}.$$

For $i = 1, \ldots, m$, define the convex sets $\Omega_i \doteq \mathcal{X}_i \bigcap \mathcal{H}$, and consider any collection $\{\Omega_{i_1}, \ldots, \Omega_{i_n}\}$ of n of these sets.

Suppose now (for the purpose of contradiction) that $q > n$. Then, there must exist an index $j \notin \{i_1, \ldots, i_n\}$ such that $\mathcal{X}_j$ is a support constraint, and by the previous reasoning, this means that there exists a point $\bar{x}_j^*$ such that $\bar{x}_j^* \in \left(\bigcap_{i \in \{1,\ldots,m\} \setminus j} \mathcal{X}_i \right) \bigcap \mathcal{H}$. Thus, $\bar{x}_j^* \in \Omega_{i_1} \cap \cdots \cap \Omega_{i_n}$, that is the collection of convex sets $\{\Omega_{i_1}, \ldots, \Omega_{i_n}\}$ has at least a point in common. Now, since the sets Ω_i, $i = 1, \ldots, m$, belong to the hyperplane $\mathcal{H}$ (i.e. to $\mathbb{R}^{n-1}$, modulo a fixed translation) and all collections composed of n of these sets have a point in common, by Helly's lemma (Lemma 1) there exists a point $\tilde{x}$ such that $\tilde{x} \in \bigcap_{i \in \{1,\ldots,m\}} \Omega_i$. Such a $\tilde{x}$ would therefore be feasible for problem $\mathcal{P}$; moreover, it would yield an objective value $\tilde{J} = c^T \tilde{x} < c^T x^* = J^*$ (since $\tilde{x} \in \mathcal{H}$). This is a contradiction, because x^* would no longer be an optimal solution for $\mathcal{P}$, and hence we conclude that $q \leq n$. ⋆

We are now ready to present a proof of Theorem 1.

A.2 Proof of Theorem 1

We prove that the conclusion in Theorem 1 holds with

$$N \geq \frac{1}{1-\nu} \left(\frac{1}{\epsilon} \ln \frac{1}{\beta} + n_\theta + \frac{n_\theta}{\epsilon} \ln \frac{1}{\nu \epsilon} + \frac{1}{\epsilon} \ln \left(\left(\frac{n_\theta}{e} \right)^{n_\theta} \frac{1}{n_\theta !} \right) \right), \tag{22}$$

where ν is a parameter that can be freely selected in $(0, 1)$. To prove that bound (5) follows from (22), proceed as follows. Since $n_\theta! \geq (n_\theta/e)^{n_\theta}$, the last term in (22) is not positive and can be dropped, leading to the bound

$$N \geq \frac{1}{1-\nu}\left(\frac{1}{\epsilon}\ln\frac{1}{\beta} + n_\theta + \frac{n_\theta}{\epsilon}\ln\frac{1}{\nu\epsilon}\right), \tag{23}$$

Bound (5) is then obtained from (23) by taking $\nu = 1/2$. We also note that further optimizing (23) with respect to ν always leads to a $\nu \leq 1/2$ with a corresponding improvement by at most of a factor 2.

Proof of Theorem 1 with (22) in place of (5).

The fact that, if RCP_N is unfeasible, then RCP is unfeasible too is trivially true, since RCP exhibits more constraints than RCP_N. Thus, we have to prove that, with probability $1-\beta$, either RCP_N is unfeasible or, if feasible, its solution is ϵ-level robustly feasible. This part of the proof is inspired by a similar proof given in a different context in [35].

For clarity of exposition, we first assume that problem RCP_N is feasible for any selection of $\delta^{(1)}, \ldots, \delta^{(N)}$. The case where infeasibility can occur is obtained as an easy extension as indicated at the end of the proof.

Given N scenarios $\delta^{(1)}, \ldots, \delta^{(N)}$, select a subset $I = \{i_1, \ldots, i_{n_\theta}\}$ of n_θ indices from $\{1, \ldots, N\}$ and let $\hat{\theta}_I$ be the optimal solution of the program

$$\begin{aligned} \min_{\theta\in\Theta} c^T\theta \quad & \text{subject to:} \\ & f(\theta, \delta^{(i_j)}) \leq 0, \; j = 1, \ldots, n_\theta. \end{aligned}$$

Based on $\hat{\theta}_I$ we next introduce a subset Δ_I^N of the set Δ^N defined as

$$\Delta_I^N \doteq \{(\delta^{(1)}, \ldots, \delta^{(N)}) : \quad \hat{\theta}_I = \hat{\theta}_N\} \tag{24}$$

($\hat{\theta}_N$ is the optimal solution with all N constraints $\delta^{(1)}, \ldots, \delta^{(N)}$ in place).

Let now I range over the collection $\mathcal{I}$ of all possible choices of n_θ indices from $\{1, \ldots, N\}$ ($\mathcal{I}$ contains $\binom{N}{n_\theta}$ sets). We want to prove that

$$\Delta^N = \bigcup_{I\in\mathcal{I}} \Delta_I^N. \tag{25}$$

To show (25), take any $(\delta^{(1)}, \ldots, \delta^{(N)}) \in \Delta^N$. From the set of constraint $\delta^{(1)}, \ldots, \delta^{(N)}$ eliminate a constraint which is not a support constraint (this is possible in view of Theorem 3, since $N > n_\theta$). The resulting optimization problem with $N-1$ constraints admits the same optimal solution $\hat{\theta}_N$ as the original problem with N constraints. Consider now the set of the remaining $N-1$ constraints and, among these, remove a constraint which is not a support constraint for the problem with $N-1$ constraints. Again, the optimal solution does not change. If we keep going this way until we are left with n_θ constraints, in the end we still have $\hat{\theta}_N$ as optimal solution, showing that $(\delta^{(1)}, \ldots, \delta^{(N)}) \in \Delta_I^N$, where I is the set containing the n_θ constraints remaining at the end of the process. Since this is true for any choice of $(\delta^{(1)}, \ldots, \delta^{(N)}) \in \Delta^N$, (25) is proven.

Next, let

$$B \doteq \{(\delta^{(1)}, \ldots, \delta^{(N)}) : \quad V(\hat{\theta}_N) > \epsilon\}$$

and

$$B_I \doteq \{(\delta^{(1)}, \ldots, \delta^{(N)}) : \quad V(\hat{\theta}_I) > \epsilon\}.$$

We now have:

$$\begin{aligned} B &= B \cap \Delta^N \\ &= B \cap (\cup_{I \in \mathcal{I}} \Delta_I^N) \quad \text{(apply (25))} \\ &= \cup_{I \in \mathcal{I}} (B \cap \Delta_I^N) \\ &= \cup_{I \in \mathcal{I}} (B_I \cap \Delta_I^N). \quad \text{(because of (24))} \end{aligned} \tag{26}$$

A bound for $\text{Prob}^N(B)$ is now obtained by bounding $\text{Prob}(B_I \cap \Delta_I^N)$ and then summing over $I \in \mathcal{I}$.

Fix any I, e.g. $I = \{1, \ldots, n_\theta\}$ to be more explicit. The set $B_I = B_{\{1,\ldots,n_\theta\}}$ is in fact a cylinder with base in the cartesian product of the first n_θ constraint domains (this follows from the fact that condition $V(\hat{\theta}_{\{1,\ldots,n_\theta\}}) > \epsilon$ only involves the first n_θ constraints). Fix $(\bar{\delta}^{(1)}, \ldots, \bar{\delta}^{(n_\theta)}) \in$ base of the cylinder. For a point $(\bar{\delta}^{(1)}, \ldots, \bar{\delta}^{(n_\theta)}, \delta^{(n_\theta+1)}, \ldots, \delta^{(N)})$ to be in $B_{\{1,\ldots,n_\theta\}} \cap \Delta_{\{1,\ldots,n_\theta\}}^N$, constraints $\delta^{(n_\theta+1)}, \ldots, \delta^{(N)}$ must be satisfied by $\hat{\theta}_{\{1,\ldots,n_\theta\}}$, for, otherwise, we would not have $\hat{\theta}_{\{1,\ldots,n_\theta\}} = \hat{\theta}_N$, as it is required in $\Delta_{\{1,\ldots,n_\theta\}}^N$. But, $V(\hat{\theta}_{\{1,\ldots,n_\theta\}}) > \epsilon$ in $B_{\{1,\ldots,n_\theta\}}$. Thus, by the fact that the extractions are independent, we conclude that

$$\begin{aligned} \text{Prob}^{N-n_\theta}\{(\delta^{(n_\theta+1)}, \ldots, \delta^{(N)}) : \quad & (\bar{\delta}^{(1)}, \ldots, \bar{\delta}^{(n_\theta)}, \delta^{(n_\theta+1)}, \ldots, \delta^{(N)}) \\ & \in B_{\{1,\ldots,n_\theta\}} \cap \Delta_{\{1,\ldots,n_\theta\}}^N\} < (1-\epsilon)^{N-n_\theta}. \end{aligned}$$

The probability on the left hand side is nothing but the conditional probability that $(\delta^{(1)}, \ldots, \delta^{(N)}) \in B_{\{1,\ldots,n_\theta\}} \cap \Delta_{\{1,\ldots,n_\theta\}}^N$ given $\delta^{(1)} = \bar{\delta}^{(1)}, \ldots, \delta^{(n_\theta)} = \bar{\delta}^{(n_\theta)}$. Integrating over the base of the cylinder $B_{\{1,\ldots,n_\theta\}}$, we then obtain

$$\text{Prob}^N(B_{\{1,\ldots,n_\theta\}} \cap \Delta_{\{1,\ldots,n_\theta\}}^N) < (1-\epsilon)^{N-n_\theta} \cdot \text{Prob}^{n_\theta}(\text{base of } B_{\{1,\ldots,n_\theta\}}) \le (1-\epsilon)^{N-n_\theta}. \tag{27}$$

From (26), we finally arrive to the desired bound for $\text{Prob}^N(B)$

$$\text{Prob}^N(B) \le \sum_{I \in \mathcal{I}} \text{Prob}^N(B_I \cap \Delta_I^N) < \binom{N}{n_\theta} (1-\epsilon)^{N-n_\theta}. \tag{28}$$

The last part of the proof is nothing but algebraic manipulations on bound (28) to show that, if N is chosen according to (22), then

$$\binom{N}{n_\theta} (1-\epsilon)^{N-n_\theta} \le \beta, \tag{29}$$

so concluding the proof. These manipulations are reported next.
Any of the following inequality implies the next in a top-down fashion, where the first one is (22):

$$\begin{aligned}
N &\geq \frac{1}{1-\nu}\left(\frac{1}{\epsilon}\ln\frac{1}{\beta}+n_\theta+\frac{n_\theta}{\epsilon}\ln\frac{1}{\nu\epsilon}+\frac{1}{\epsilon}\ln\left(\left(\frac{n_\theta}{e}\right)^{n_\theta}\frac{1}{n_\theta!}\right)\right)\\
(1-\nu)N &\geq \frac{1}{\epsilon}\ln\frac{1}{\beta}+n_\theta+\frac{n_\theta}{\epsilon}\ln\frac{1}{\nu\epsilon}+\frac{1}{\epsilon}\ln\left(\left(\frac{n_\theta}{e}\right)^{n_\theta}\frac{1}{n_\theta!}\right)\\
(1-\nu)N &\geq \frac{1}{\epsilon}\ln\frac{1}{\beta}+n_\theta+\frac{n_\theta}{\epsilon}\left(\ln\frac{n_\theta}{\nu\epsilon}-1\right)-\frac{1}{\epsilon}\ln(n_\theta!)\\
N &\geq \frac{1}{\epsilon}\ln\frac{1}{\beta}+n_\theta+\frac{n_\theta}{\epsilon}\left(\ln\frac{n_\theta}{\nu\epsilon}-1+\frac{\nu N\epsilon}{n_\theta}\right)-\frac{1}{\epsilon}\ln(n_\theta!)\\
N &\geq \frac{1}{\epsilon}\ln\frac{1}{\beta}+n_\theta+\frac{n_\theta}{\epsilon}\ln N-\frac{1}{\epsilon}\ln(n_\theta!), \qquad (30)
\end{aligned}$$

where the last implication can be justified by observing that $\ln x \geq 1-\frac{1}{x}$, for $x>0$, and applying this inequality with $x=\frac{n_\theta}{\nu N\epsilon}$. Proceeding from (30), the next inequalities in the chain are

$$\begin{aligned}
\ln\beta &\geq -\epsilon N+\epsilon n_\theta+n_\theta\ln N-\ln(n_\theta!)\\
\beta &\geq \frac{N^{n_\theta}}{n_\theta!}e^{-\epsilon(N-n_\theta)}\\
\beta &\geq \frac{N(N-1)\cdots(N-n_\theta+1)}{n_\theta!}(1-\epsilon)^{N-n_\theta},
\end{aligned}$$

where, in the last implication, we have used the fact that $e^{-\epsilon(N-n_\theta)} \geq (1-\epsilon)^{N-n_\theta}$, as it follows by taking logarithm of the two sides and further noting that $-\epsilon \geq \ln(1-\epsilon)$. The last inequality can be rewritten as

$$\beta \geq \binom{N}{n_\theta}(1-\epsilon)^{N-n_\theta},$$

which is (29).

So far, we have assumed that RCP_N is feasible for any selection of $\delta^{(1)},\dots,\delta^{(N)}$. Relax now this assumption and call $F \subseteq \Delta^N$ the set where RCP_N is indeed feasible. The same derivation can then be worked out in the domain F, instead of Δ^N, leading to the conclusion that (28) holds with $B \doteq \left\{(\delta^{(1)},\dots,\delta^{(N)}) \in F : V(\hat{\theta}_N) > \epsilon\right\}$. This concludes the proof. ⋆

B Proof of Theorem 2

The first claim is immediate, since from Theorem 1, with probability at least $1-\beta$, if RCP_N is feasible, then its optimal solution $\hat{\theta}_N$ satisfies $V(\hat{\theta}_N) \leq \epsilon$, i.e. it is a feasible, albeit possibly not optimal, solution for problem $\text{CCP}(\epsilon)$, and hence $J_{\text{RCP}_\text{N}} \geq J_{\text{CCP}(\epsilon)}$.

To prove the second claim, notice that if θ is feasible for problem $\text{CCP}(\epsilon_1)$ with $\epsilon_1 = 1-(1-\beta)^{1/N}$, i.e.

$$\text{Prob}\{\delta \in \Delta : \ f(\theta, \delta) > 0\} \leq 1-(1-\beta)^{1/N},$$

then for each of N independent extractions $\delta^{(1)}, \dots, \delta^{(N)}$ of δ it holds that

$$\text{Prob}\{\delta^{(i)} \in \Delta : \ f(\theta, \delta^{(i)}) \leq 0\} \geq (1-\beta)^{1/N}, \quad i = 1, \dots, N,$$

and hence, by independence, the joint event $\{(\delta^{(1)}, \dots, \delta^{(N)}) \in \Delta^N : f(\theta, \delta^{(i)}) \leq 0, \ i = 1, \dots, N\}$ holds with probability at least $1-\beta$. This means that, with probability at least $1-\beta$, a feasible point for $\text{CCP}(\epsilon_1)$ is also a feasible point for RCP_N. We now have two possibilities, depending on whether $\text{CCP}(\epsilon_1)$ attains an optimal solution (i.e. a $\hat{\theta}$ feasible for $\text{CCP}(\epsilon_1)$ exists such that $c^T\hat{\theta} = J_{\text{CCP}(\epsilon_1)}$) or not. In the first situation ($\hat{\theta}$ exists), taking $\theta = \hat{\theta}$ in the previous reasoning immediately implies that $J_{\text{RCP}_\text{N}} \leq J_{\text{CCP}}(\epsilon_1)$, as desired.

In the second situation ($\hat{\theta}$ does not exist), consider a point $\bar{\theta}$ which is feasible for $\text{CCP}(\epsilon_1)$ and such that $c^T\bar{\theta} \leq J_{\text{CCP}(\epsilon_1)} + \rho$, for some $\rho > 0$ (such a $\bar{\theta}$ exists since $J_{\text{CCP}(\epsilon_1)} = \inf c^T\theta$ over θ's that are feasible for $\text{CCP}(\epsilon_1)$). By the previous reasoning, this implies that, with probability at least $1-\beta$, the point $\bar{\theta}$ is also feasible for problem RCP_N, entailing

$$\text{Prob}\left\{(\delta^{(1)}, \dots, \delta^{(N)}) \in \Delta^N : J_{\text{RCP}_\text{N}} \leq J_{\text{CCP}(\epsilon_1)} + \rho\right\} \geq 1-\beta. \tag{31}$$

For the purpose of contradiction, suppose now that result 2 in the theorem is violated so that $J_{\text{RCP}_\text{N}} > J_{\text{CCP}(\epsilon_1)}$ with probability larger than β. Since

$$\left\{(\delta^{(1)}, \dots, \delta^{(N)}) \in \Delta^N : J_{\text{RCP}_\text{N}} > J_{\text{CCP}(\epsilon_1)}\right\} = \bigcup_{\nu > 0} \left\{(\delta^{(1)}, \dots, \delta^{(N)}) \in \Delta^N : J_{\text{CCP}_\text{N}} > J_{\text{CCP}(\epsilon_1)} + \frac{1}{\nu}\right\},$$

then

$$\beta < \text{Prob}^N\left\{(\delta^{(1)}, \dots, \delta^{(N)}) \in \Delta^N : J_{\text{RCP}_\text{N}} > J_{\text{CCP}(\epsilon_1)}\right\} =$$
$$\lim_{\nu \to \infty} \text{Prob}^N\left\{(\delta^{(1)}, \dots, \delta^{(N)}) \in \Delta^N : J_{\text{RCP}_\text{N}} > J_{\text{CCP}(\epsilon_1)} + \frac{1}{\nu}\right\}$$

and we conclude that there exists a $\bar{\nu}$ such that $J_{\text{RCP}_\text{N}} > J_{\text{CCP}(\epsilon_1)} + \frac{1}{\bar{\nu}}$ with probability larger than β. But this contradicts (31) for $\rho = \frac{1}{\bar{\nu}}$, so concluding the proof. ⋆

C Relaxing the assumption that RCP_N has a unique solution

In Section 3, the theory has been developed under Assumption 2 requiring that RCP_N is either unfeasible or, if feasible, it admits a unique optimal solution. Here, we drop Assumption 2 and consider the general case allowing for non-uniqueness of the solution or non-existence of the solution even when RCP_N is feasible (i.e. the solution escapes to infinity).

C.1 Non-uniqueness of the solution

We follow the same approach as in [24], Section 4.1. Suppose that when problem RCP_N admits more than one optimal solution we break the tie by a tie-break rule as follows:

Tie-break rule: *Let $t_i(\theta)$, $i = 1, \dots, p$, be given convex and continuous functions. Among the optimal solutions for RCP_N, select the one that minimizes $t_1(\theta)$. If indetermination still occurs, select among the θ that minimize $t_1(\theta)$ the solution that minimizes $t_2(\theta)$, and so on with $t_3(\theta), t_4(\theta), \dots$. We assume that functions $t_i(\theta)$, $i = 1, \dots, p$, are selected so that the tie is broken within p steps at most. As a simple example of a tie-break rule, one can consider $t_1(\theta) = \theta_1, t_2(\theta) = \theta_2, \dots$.* $\star$

From now on, by 'optimal solution' we mean either the unique optimal solution, or the solution selected according to the Tie-break rule, if multiple optimal solutions occur.

Theorem 1 hold unchanged if we drop the uniqueness requirement in Assumption 2, provided that 'optimal solution' is intended in the indicated sense.

To see this, generalize Definition 5 of support constraints to: *The k-th constraint $\mathcal{X}_k$ is a support constraint for $\mathcal{P}$ if problem $\mathcal{P}_k$ has an optimal solution x_k^* such that $x_k^* \neq x^*$.* Indeed this definition generalizes Definition 5 since, in case of a single optimal solution, $x_k^* \neq x^*$ is equivalent to $c^T x_k^* < c^T x^*$. In [24], Section 4.1, it is proven that Theorem 3 holds true with this extended definition of support constraint (i.e. the number of support constraints is at most n_θ), and then an inspection of the proof of Theorem 1 in Appendix A.2 reveals that this proof goes through unaltered in the present setting, so concluding that Theorem 1 still holds.

C.2 Non-existence of the solution

Even when RCP_N is feasible, it may happen that no optimal solution exists since the set for θ allowed by the extracted constraints is unbounded in such a way that the optimal solution escapes to infinity. In this section, we further generalize Theorem 1 so as to cope with this situation too and then provide a reformulation of Theorem 1 (Theorem 4 below) that covers all possible situations.

Suppose that a random extraction of a multisample $\delta^{(1)}, \dots, \delta^{(N)}$ is rejected when the problem is feasible but no optimal solution exists, and another extraction is performed in this case. Then, the result of Theorem 1 holds if attention is restricted to the accepted multi-samples. This idea is now formalized.

Let $D \subseteq \Delta^N$ be the set where RCP_N is feasible but an optimal solution does not exist (it escapes to infinity) and assume that its complement $A = D^c$ has positive probability: $\mathrm{Prob}^N(A) > 0$. Moreover, let Prob_A^N be the probability Prob^N restricted to A: $\mathrm{Prob}_A^N(\cdot) \doteq \mathrm{Prob}^N(\cdot \cap A)/\mathrm{Prob}^N(A)$. In addition, assume that if a problem with, say, m constraints is feasible and admits optimal solution, then, after adding an extra $(m+1)$-th constraint, if the problem remains feasible, then an optimal solution continues to exists (this rules out the possibility of pathological situations where adding a constraint forces the solution to drift away to infinity).

Going trough the proof of Theorem 1, we can readily see that it remains valid if attention is restricted to A. Precisely, the following theorem holds:

Theorem 4 *Fix two real numbers $\epsilon \in (0,1)$ (level parameter) and $\beta \in (0,1)$ (confidence parameter). If*

$$N \geq N(\epsilon, \beta) \doteq \frac{2}{\epsilon} \ln \frac{1}{\beta} + 2n_\theta + \frac{2n_\theta}{\epsilon} \ln \frac{2}{\epsilon},$$

then, with probability Prob_A^N *no smaller than* $1-\beta$*, one of the following two facts happen: i) the randomized optimization problem* RCP_N *is unfeasible, and then* RCP *is unfeasible; or, ii)* RCP_N *is feasible, and then its optimal solution* $\hat{\theta}_N$ *(unique after the Tie-break rule has been applied) is ϵ-level robustly feasible.* ⋆

The proof of this theorem is obtained by following the same steps as in the proof of Theorem 1 in Appendix A.2 with a few simple amendments, as sketched in the following.

Similarly to the proof of Theorem 1, forget for the time being that RCP_N can be unfeasible and assume $F = \Delta^N$. In the present context, interpret all subset of Δ^N (e.g. Δ_I^N, B, B_I) as subsets of A, so e.g. $\Delta_I^N \doteq \{(\delta^{(1)}, \ldots, \delta^{(N)}) \in A : \hat{\theta}_I = \hat{\theta}_N\}$. Everything in the proof goes through unaltered till equation (27). In (27), drop the last step and consider the inequality

$$\mathrm{Prob}^N(B_{\{1,\ldots,n_\theta\}} \cap \Delta^N_{\{1,\ldots,n_\theta\}}) < (1-\epsilon)^{N-n_\theta} \cdot \mathrm{Prob}^{n_\theta}(\text{base of } B_{\{1,\ldots,n_\theta\}}). \qquad (32)$$

Now, the cylinder $B_{\{1,\ldots,n_\theta\}}$ does not intersect D since any multi-sample in the cylinder is formed by the first n_θ samples that generate the optimal solution $\hat{\theta}_{\{1,\ldots,n_\theta\}}$, plus the remaining $N - n_\theta$ samples that, in conjunction with the first n_θ, either make the problem unfeasible, or, if feasible, add constraints so still preventing escape to infinity. So, $\mathrm{Prob}^{n_\theta}(\text{base of } B_{\{1,\ldots,n_\theta\}}) = \mathrm{Prob}^N(B_{\{1,\ldots,n_\theta\}}) \leq 1 - \mathrm{Prob}^N(D) = \mathrm{Prob}^N(A)$, which, used in (32) gives:

$$\mathrm{Prob}^N(B_{\{1,\ldots,n_\theta\}} \cap \Delta^N_{\{1,\ldots,n_\theta\}}) < (1-\epsilon)^{N-n_\theta} \cdot \mathrm{Prob}^N(A),$$

and, after substitution in (28), we obtain

$$\mathrm{Prob}^N(B) \leq \sum_{I \in \mathcal{I}} \mathrm{Prob}^N(B_I \cap \Delta_I) < \binom{N}{n_\theta} (1-\epsilon)^{N-n_\theta} \cdot \mathrm{Prob}^N(A) \leq \beta \cdot \mathrm{Prob}^N(A),$$

or, equivalently,

$$\mathrm{Prob}^N(B)/\mathrm{Prob}^N(A) \leq \beta.$$

Since the left hand side is $\mathrm{Prob}_A^N(B)$, the desired result remains proven. The case when F is a strict subset of Δ^N can be dealt with without any additional complication. ⋆

References

[1] B.D.O. Anderson and J.B. Moore. *Optimal Control: Linear Quadratic Methods.* Prentice, Englewood Cliffs, 1990.

[2] P. Apkarian and R.J. Adams. Advanced gain-scheduling techniques for uncertain systems. *IEEE Trans. Control Sys. Tech.*, 6:21–32, 1998.

[3] P. Apkarian and H.D. Tuan. Parameterized LMIs in control theory. *SIAM Journal on Control and Optimization*, 38(4):1241–1264, 2000.

[4] P. Apkarian, H.D. Tuan, and J. Bernussou. Continuous-time analysis, eigenstructure assignment, and H_2 synthesis with enhanced linear matrix inequalities (LMI) characterizations. *IEEE Trans. Aut. Control*, 46(12):1941–1946, 2001.

[5] T. Asai and S. Hara. A unified approach to LMI-based reduced order self-scheduling control synthesis. *Systems and Control Letters*, 36:75–86, 1999.

[6] B.R. Barmish and C.M. Lagoa. The uniform distribution: A rigorous justification for its use in robustness analysis. *Mathematics of Control, Signals, and Systems*, 10:203–222, 1997.

[7] B.R. Barmish and P.S. Scherbakov. On avoiding vertexization of robustness problems: the approximate feasibility concept. In *Proc. IEEE Conf. on Decision and Control*, volume 2, pages 1031–1036, Sydney, Australia, December 2000.

[8] G. Becker and A. Packard. Robust performance of linear parametrically varying systems using parametrically-dependent linear feedback. *Systems and Control Letters*, 23:205–215, 1994.

[9] A. Ben-Tal and A. Nemirovski. Robust truss topology design via semidefinite programming. *SIAM J. on Optimization*, 7(4):991–1016, 1997.

[10] A. Ben-Tal and A. Nemirovski. Robust convex optimization. *Mathematics of Operations Research*, 23(4):769–805, 1998.

[11] A. Ben-Tal and A. Nemirovski. Robust solutions of uncertain linear programs. *Operations Research Letters*, 25(1):1–13, 1999.

[12] A. Ben-Tal and A. Nemirovski. On tractable approximations of uncertain linear matrix inequalities affected by interval uncertainty. *SIAM Journal on Optimization*, 12(3):811–833, 2002.

[13] F. Blanchini. The gain scheduling and the robust state feedback stabilization problems. *IEEE Trans. Aut. Control*, 45(11):2061–2070, November 2000.

[14] F. Blanchini and S. Miani. Stabilization of LPV systems: state feedback, state estimation and duality. *SIAM Journal on Control and Optimization*, 32(1):76–97, 2003.

[15] F. Blanchini and M. Sznaier. A convex optimization approach to fixed-order controller design for disturbance rejection in SISO systems. *IEEE Trans. Aut. Control*, 45(4):784–789, April 2000.

[16] V.D. Blondel and J.N. Tsitsiklis. A survey of computational complexity results in systems and control. *Automatica*, 36:1249–1274, 2000.

[17] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. Warmuth. Learnability and the Vapnik-Chervonenkis dimension. *Journal of the ACM*, 36:929–965, 1989.

[18] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA, June 1994.

[19] S. Boyd and Q. Yang. Structured and simultaneous Lyapunov functions for system stability problems. *Int. J. Control*, 49:2215–2240, 1989.

[20] R.P. Braatz, P.M. Young, J.C. Doyle, and M. Morari. Computational complexity of μ calculation. *IEEE Trans. Aut. Control*, AC-39:1000–1002, 1994.

[21] G. Calafiore. Set simulations for quadratic systems. *IEEE Trans. Aut. Control*, 48(5):800–805, 2003.

[22] G. Calafiore and M.C. Campi. Interval predictors for unknown dynamical systems: an assessment of reliability. In *Proc. IEEE Conf. on Decision and Control*, Las Vegas, USA, December 2002.

[23] G. Calafiore and M.C. Campi. A learning theory approach to the construction of predictor models. *AIMS J. Discrete and Continuous Dynamical Systems*, pages 156–166, 2003.

[24] G. Calafiore and M.C. Campi. Uncertain convex programs: randomized solutions and confidence levels. *Mathematical Programming – Online First*, DOI 10.1007/s10107-003-0499-y, 2004.

[25] G. Calafiore and F. Dabbene. A probabilistic framework for problems with real structured uncertainty in systems and control. *Automatica*, 38(8):1265–1276, 2002.

[26] G. Calafiore, F. Dabbene, and R. Tempo. Randomized algorithms in robust control. In *Proc. IEEE Conf. on Decision and Control*, Maui, USA, December 2003.

[27] G. Calafiore and L. El Ghaoui. Ellipsoidal bounds for uncertain linear equations and dynamical systems. *Automatica*, 50(5):773–787, 2004.

[28] G. Calafiore and B.T. Polyak. Fast algorithms for exact and approximate feasibility of robust LMIs. *IEEE Trans. Aut. Control*, 46(11):1755–1759, November 2001.

[29] G. Calafiore, F. Dabbene, and R. Tempo. Randomized algorithms for probabilistic robustness with real and complex structured uncertainty. *IEEE Trans. Aut. Control*, 45(12):2218–2235, December 2000.

[30] A. Charnes and W.W. Cooper. Chance constrained programming. *Management Sci.*, 6:73–79, 1959.

[31] X. Chen and K. Zhou. Order statistics and probabilistic robust control. *Systems and Control Letters*, 35:175–182, 1998.

[32] M.C. de Oliveira, J. Bernussou, and J.C. Geromel. A new discrete-time robust stability condition. *Systems and Control Letters*, 37:261–265, 1999.

[33] M.C. de Oliveira, J.C. Geromel, and J. Bernussou. Extended H_2 and H_∞ norm characterization and controller parameterization for discrete-time systems. *Int. J. Control*, 75(9):666–679, 2002.

[34] E. Feron, P. Apkarian, and P. Gahinet. Analysis and synthesis of robust control systems via parameter-dependent Lyapunov functions. *IEEE Trans. Aut. Control*, 41(7):1041–1046, 1996.

[35] S. Floyd and M. Warmuth. Sample compression, learnability, and the Vapnik-Chervonenkis dimension. *Machine Learning*, pages 1–36, 1995.

[36] Y. Fujisaki, F. Dabbene, and R. Tempo. Probabilistic design of LPV control systems. *Automatica*, 39(8):1323–1337, 2003.

[37] P. Gahinet. Explicit controller formulas for LMI-based H_∞ synthesis. *Automatica*, 32(7):1007–1014, 1996.

[38] P. Gahinet, P. Apkarian, and M. Chilali. Affine parameter-dependent Lyapunov functions and real parametric uncertainty. *IEEE Trans. Aut. Control*, 41(3):436–442, 1996.

[39] L. El Ghaoui and H. Lebret. Robust solutions to least-squares problems with uncertain data. *SIAM J. on Matrix Analysis and Applications*, 18(4):1035–1064, 1997.

[40] L. El Ghaoui and H. Lebret. Robust solutions to uncertain semidefinite programs. *SIAM J. Optim.*, 9(1):33–52, 1998.

[41] L. El Ghaoui, M. Oks, and F. Oustry. Worst-case value-at-risk and robust portfolio optimization: A conic programming approach. *Operations Research*, 51(4):543–556, July 2003.

[42] M.X. Goemans and D.P. Williamson. .878-approximation for MAX CUT and MAX 2SAT. *Proc. 26th ACM Sym. Theor. Computing*, pages 422–431, 1994.

[43] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58:13–30, 1963.

[44] S. Kanev, B. De Schutter, and M. Verhaegen. An ellipsoid algorithm for probabilistic robust controller design. *Systems and Control Letters*, 49(5):365–375, 2003.

[45] L.H. Keel and S.P. Bhattacharyya. A linear programming approach to controller design. In *Proc. IEEE Conf. on Decision and Control*, pages 2139–2148, San Diego, CA, December 1997.

[46] P.P. Khargonekar and A. Tikku. Randomized algorithms for robust control analysis have polynomial time complexity. In *Proc. IEEE Conf. on Decision and Control*, Kobe, Japan, December 1996.

[47] V. Koltchinskii, C.T. Abdallah, M. Ariola, P. Dorato, and D. Panchenko. Improved sample complexity estimates for statistical learning control of uncertain systems. *IEEE Trans. Aut. Control*, 46:2383–2387, 2000.

[48] J.B. Lasserre. Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, 11(3):796–817, 2001.

[49] A. Nemirovski. Several NP-hard problems arising in robust stability analysis. *SIAM J. on Matrix Analysis and Applications*, 6:99–105, 1993.

[50] Y. Oishi. Probabilistic design of a robust state-feedback controller based on parameter-dependent lyapunov functions. In *Proc. IEEE Conf. on Decision and Control*, pages 1920–1925, December 2003.

[51] Y. Oishi and H. Kimura. Randomized algorithms to solve parameter-dependent linear matrix inequalities and their computational complexity. In *Proc. IEEE Conf. on Decision and Control*, volume 2, pages 2025–2030, December 2001.

[52] D. Peaucelle and D. Arzelier. Robust performance analysis with LMI-based methods for real parametric uncertainty via parameter-dependent Lyapunov functions. *IEEE Trans. Aut. Control*, 46(4):624–630, 2001.

[53] B.T. Polyak and M.E. Halpern. The use of a new optimization criterion for discrete-time feedback controller design. In *Proc. IEEE Conf. on Decision and Control*, pages 894–899, Phoenix, AZ, December 1999.

[54] B.T. Polyak and R. Tempo. Probabilistic robust design with linear quadratic regulators. *Systems and Control Letters*, 43:343–353, 2001.

[55] A. Prékopa. *Stochastic Programming.* Kluwer, 1995.

[56] L.R. Ray and R.F. Stengel. A Monte Carlo approach to the analysis of control system robustness. *Automatica*, 29:229–236, 1993.

[57] R.T. Rockafellar. *Convex Analysis.* Princeton University Press, Princeton, NJ, 1970.

[58] W.J. Rugh and J.S. Shamma. Research on gain scheduling. *Automatica*, 36:1401–1425, 2000.

[59] C.W. Scherer. LPV control and full block multipliers. *Automatica*, 37:361–375, 2001.

[60] R.E. Skelton, T. Iwasaki, and K. Grigoriadis. *A unified algebraic approach to linear control design.* Taylor & Francis, London, 1998.

[61] R. Tempo, E.W. Bai, and F. Dabbene. Probabilistic robustness analysis: Explicit bounds for the minimum number of samples. *Systems and Control Letters*, 30:237–242, 1997.

[62] R. Tempo, G. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems.* Communications and Control Engineering Series. Springer-Verlag, London, 2004.

[63] A. Trofino and C.E. de Souza. Biquadratic stability of uncertain linear systems. *IEEE Trans. Aut. Control*, 46(8):1303–1307, 2001.

[64] J.S. Tyler and F.B. Tuteur. The use of a quadratic performance index to design multivariable invariant plants. *IEEE Trans. Aut. Control*, AC-11:84–92, 1966.

[65] S. Vajda. *Probabilistic Programming.* Academic Press, New York, 1972.

[66] V. Vapnik. *Statistical Learning Theory.* Wiley, New York, 1996.

[67] M. Vidyasagar. *A Theory of Learning and Generalization.* Springer-Verlag, London, 1997.

[68] M. Vidyasagar. Randomized algorithms for robust controller synthesis using statistical learning theory. *Automatica*, 37(10):1515–1528, October 2001.

[69] M. Vidyasagar and V. Blondel. Probabilistic solutions to some NP-hard matrix problems. *Automatica*, 37(9):1397–1405, September 2001.

[70] F. Wu and K.M. Grigoriadis. LPV systems with parameter-varying time delays: analysis and control. *Automatica*, 37:221–229, 2001.