**Proceedings of the 42nd IEEE
Conference on Decision and Control
Maui, Hawaii USA, December 2003**

WeM13-2

Robust Convex Programs: Randomized Solutions and Applications in Control

Giuseppe Calafiore
Dipartimento di Automatica e Informatica
Politecnico di Torino
Cso Duca degli Abruzzi, 24 – 10129 Torino, Italy
e-mail: giuseppe.calafiore@polito.it

M.C. Campi
Dipartimento di Automatica per l'Automazione
Università di Brescia
Via Branze, 38 – 25123 Brescia, Italy
e-mail: campi@ing.unibs.it

***Abstract*— Many engineering problems can be cast as optimization problems subject to convex constraints that are parameterized by an uncertainty or 'instance' term. A recently emerged successful paradigm for attacking these problems is robust optimization, where one seeks a solution which simultaneously satisfies all possible constraint instances. In practice, however, the robust approach is computationally viable only for problem families with rather simple dependence on the instance parameter (such as affine or polynomial), and leads in general to conservative answers, since the solution is computed transforming the original semi-infinite problem into a standard one, by means of relaxation techniques.**

In this paper, we take an alternative 'randomized' or 'scenario' approach: by randomly sampling the uncertainty parameter, we substitute the original infinite constraint set with a finite set of N constraints. We show that the resulting randomized solution fails to satisfy only a small portion of the original constraints, provided that a sufficient number of samples is drawn. Our key result is to provide an efficient and explicit bound on the measure (probability or volume) of the original constraints that are possibly violated by the randomized solution. This volume rapidly decreases to zero as N is increased. The proposed paradigm is here applied to the solution of a wide class of NP-hard control problems representable by means of parameter-dependent linear matrix inequalities.

I. 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 [7], [14]. 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. 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, see for instance [1].

This 'robust' convex programming paradigm has emerged around 1998 (see [4], [17]) and, besides the systems and control areas, has found applications in, to mention but a few, truss topology design, robust antenna array design, portfolio optimization, and robust estimation [16]. Unfortunately however, robust convex programs are not as easily solvable as standard ones, and are NP-hard in general, [4]. This implies 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 [6], [17]. The main drawbacks of the relaxation approach are that the extent of the introduced conservatism is in general unknown, 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.

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 uncertainties) but are instead intended in a probabilistic one (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 [12], [21], [24], [26]. However, the probabilistic approach has found to date limited application for 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, see [28] and the references therein, and the sequential methods based on stochastic gradient iterations [11], [13], [23] or ellipsoid iterations, [19].

The first approach, proposed in the seminal paper [28], is to date the most general one, since it permits to tackle non-convex and NP-hard design problems. However, it suffers from the conservatism of the Vapnik-Chervonenkis theory, 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. Moreover, the design methodology proposed in [28] does not directly aim to enforce the synthesis constraints in a (probabilistically) robust sense, but it rather aims at satisfying them *on average*.

As an alternative, when the original synthesis problem is convex (which includes many, albeit not all, relevant control problems), the sequential approaches based on stochastic gradients [11], [13], [22], [23] or ellipsoid iterations, [19], may be applied with success. However, these methods are currently limited to convex feasibility problems, and have not yet been satisfactorily 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 (i.e. the 'sample complexity' of the algorithm) cannot be fixed in advance, unless unrealistic a-priori assumptions are made.

The main contribution of the present work is to propose a general framework for solving in a probabilistic sense convex programs affected by uncertainty. Specifically, we 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. In the key result of this paper (Theorem 1) we provide an efficient bound on the sample complexity of our randomized technique, as a function of the required probabilistic robustness levels. Also, a notable improvement upon the stochastic sequential methods of [11], [13], [19], [22], [23] is that our result holds for robust optimization problems (and not only for feasibility), and that an explicit a-priori bound is given on the sample complexity of the method.

This paper is organized as follows. Section II motivates our developments by showing several relevant control problems that are naturally cast in the form of robust convex programs, and pinpoints the specific limitations of the deterministic solution methods available in the literature. Section III is the main section. There, the sampled counterpart of a robust convex program is defined and the fundamental result assessing the properties of the solution of the randomized problem is stated (Theorem 1). Section IV reports a numerical control design example, while conclusions are drawn in Section V.

II. Robust Convex Programs in Control

By a robust convex program (RCP) is here meant a convex optimization problem of the form (see e.g. [4])

$$\text{RCP}: \quad \min_{x\in\mathcal{X}} \; c^T x \quad \text{subject to:} \tag{1}$$

$$f(x,\delta) \le 0, \forall \delta \in \mathbf{\Delta}, \tag{2}$$

where $x \in \mathcal{X}$ is the optimization variable, $\mathcal{X} \subseteq \mathbb{R}^{n_x}$ is convex and closed, and $\mathbf{\Delta} \subset \mathbb{R}^d$ is a closed set. The function $f(x,\delta): \mathcal{X} \times \mathbf{\Delta} \to \mathbb{R}^{n_f}$ is continuous and convex in x for any fixed $\delta \in \mathbf{\Delta}$, and the inequality $f(x,\delta) \le 0$ is intended element-wise. Important special cases of the above problem are robust linear programs [5], for which $f(x,\delta)$ is affine in x, and robust semidefinite programs [6], [17], for which $f(x,\delta) = \lambda_{\max}[F(x,\delta)]$, where

$$F(x,\delta) = F_0(\delta) + \sum_{i=1}^{n} x_i F_i(\delta), \quad F_i(\delta) = F_i^T(\delta).$$

Similarly, one can consider the related problem of robust feasibility (RFP), i.e. determine (if one exists) $x \in \mathcal{X}$ such that $f(x,\delta) \le 0, \forall \delta \in \mathbf{\Delta}$. This feasibility problem can be cast in an equivalent minimization form in the augmented variables x, η

$$\min_{x\in\mathcal{X},\eta\in\mathbb{R}} \; \eta \quad \text{subject to:} \tag{3}$$

$$f(x,\delta) - \eta \mathbf{1} \le 0, \forall \delta \in \mathbf{\Delta}, \tag{4}$$

where $\mathbf{1}$ is a vector of ones. Provided that the optimum in the above problem is attained, the original problem (RFP) is feasible if and only if the RCP (3)-(4) has a non-positive optimal objective, and it is strictly feasible if the optimal objective is negative. Hence, the class (1)-(2) actually encompasses both feasibility and optimization problems.

In this section, we survey some relevant control analysis and synthesis problems that can be naturally cast in the above form. This list of problems is by no means complete or representative of all the interesting problems that can be encountered in the literature. However, our aim is here to motivate the introduction of the probabilistic approach by showing a selection of problems for which no deterministic polynomial-time algorithm is known that computes an exact solution. We refer the reader to [1] for further examples along this line.

A. Analysis via parameter-dependent Lyapunov functions

Consider the finite-dimensional linear uncertain system described in state-space form as

$$\dot{\xi} = A(\delta)\xi, \tag{5}$$

where $\xi \in \mathbb{R}^n$ is the state variable. Assume further that the system matrix is a generic function of a vector of uncertain parameters $\delta \in \mathbb{R}^d$, which belongs to a closed set $\mathbf{\Delta} \subset \mathbb{R}^d$. The uncertain parameter is unknown, but constant in time.

Let a symmetric matrix function $P(\theta,\delta)$ be chosen in a family parameterized by $\theta \in \mathbb{R}^p$, and assume that $P(\theta,\delta)$ is affine in θ, for all $\delta \in \mathbf{\Delta}$. The dependence of $P(\theta,\delta)$ 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 1 (Generalized quadratic stability – GQS): Given a symmetric matrix function $P(\theta,\delta)$, affine in $\theta \in \mathbb{R}^p$ for all $\delta \in \mathbf{\Delta}$, the system (5) is said quadratically stable with respect to $P(\theta,\delta)$ if there exist $\theta \in \mathbb{R}^p$ such that

$$P(\theta,\delta) \succ 0, \; A^T(\delta)P(\theta,\delta) + P(\theta,\delta)A(\delta) \prec 0, \; \forall \delta \in \mathbf{\Delta}. \tag{6}$$

For specific choices of the parameterization $P(\theta,\delta)$, the above GQS criterion clearly encompasses the popular quadratic stability (QS, [7]) and affine quadratic stability (AQS, [15]) criteria, as well as the biquadratic stability condition of [27]. For instance, the quadratic stability condition is recovered by choosing $P(\theta,\delta) = P$ (i.e. θ 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 (6). The AQS condition is instead obtained by choosing

$$P(\theta,\delta) = P_0 + \delta_1 P_1 + \cdots + \delta_d P_d, \tag{7}$$

where θ represents the free elements in the matrices $P_i = P_i^T$, $i = 0,\ldots,d$. 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, except for the 'polytopic' case, see [7]. The AQS condition is computationally hard even in the polytopic case with fixed number of vertices, and therefore convex relaxations that lead to numerically tractable sufficient conditions for AQS have been proposed in the literature, see for instance [15]. Notice also that the generic parameter dependent Lyapunov functions introduced in Definition 1 may also be used to assess other Lyapunov-based performance measures, such as the H_2 and H_∞ norms.

This first problem instance serves to highlight the fact that (as it is well-known) no efficient deterministic algorithm exists for (6) in the case of generic dependence on δ. However, a key feature of the conditions (6) is that for any *fixed* $\delta \in \mathbf{\Delta}$ they represent a convex LMI condition in θ, and therefore this problem is amenable to the probabilistic solution technique proposed in Section III.

B. Robust synthesis for LPV systems

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

$$\begin{bmatrix} \dot{\xi} \\ z \\ y \end{bmatrix} \begin{bmatrix} A(\delta(t)) & B_1(\delta(t)) & B_2(\delta(t)) \\ C_1(\delta(t)) & D_{11}(\delta(t)) & D_{12}(\delta(t)) \\ C_2(\delta(t)) & D_{21}(\delta(t)) & 0 \end{bmatrix} \begin{bmatrix} \xi \\ w \\ u \end{bmatrix} \tag{8}$$

where $\xi \in \mathbb{R}^n$ 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 $\delta(t) \in \mathbb{R}^d$ 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 $\mathbf{\Delta}$, whereas its actual time value $\delta(t)$ 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 [3] and the survey [25].

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

$$\begin{bmatrix} \dot{\xi}_k \\ u \end{bmatrix} = \begin{bmatrix} A_k(\delta(t)) & B_k(\delta(t)) \\ C_k(\delta(t)) & 0 \end{bmatrix} \begin{bmatrix} \xi_k \\ y \end{bmatrix}. \tag{9}$$

This controller should exponentially stabilize the system (8), and achieve an L_2 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 data on the scheduling parameter is non-linear. We refer the reader to [13] for a further discussion on the difficulties involved in the solution of the LPV design problem. This latter paper also proposes a new randomization-based solution technique, which motivated the introduction of this section in the present paper. Indeed, the parameter-dependent inequalities derived in [13] are there solved using sequential stochastic gradient methods (see also [11]), while the same inequalities are here viewed as an instance of a robust convex feasibility problem, and hence directly amenable to the randomized solution proposed in Section III.

Under standard regularity assumptions (see [13]), the L_2 quadratic LPV control problem is formalized 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 has L_2 gain smaller than γ, for all $\delta(t) \in \mathbf{\Delta}$. The solvability conditions for this problem are directly stated in terms of robust feasibility of three LMIs in [3] (Theorem 4.2), or in an equivalent quadratic matrix inequality form in [13] (Lemma 1). We state these conditions in the following lemma.

Lemma 1 (Robust quadratic LPV synthesis): The L_2 quadratic LPV control problem is solvable if and only if there exist $0 \prec P = P^T \in \mathbb{R}^{n,n}$ and $0 \prec Q = Q^T \in \mathbb{R}^{n,n}$ such that, for all $\delta \in \mathbf{\Delta}$,

$$\begin{bmatrix} A(\delta)P + PA(\delta) + \gamma^{-2}B_1(\delta)B_1^T(\delta) - B_2(\delta)B_2^T(\delta) & PC_1^T(\delta) \\ * & -I \end{bmatrix} \prec 0$$

$$\begin{bmatrix} A^T(\delta)Q + QA(\delta) + \gamma^{-2}C_1^T(\delta)C_1(\delta) - C_2^T(\delta)C_2(\delta) & QB_1(\delta) \\ * & -I \end{bmatrix} \prec 0$$

$$\begin{bmatrix} P & \gamma^{-1}I \\ * & Q \end{bmatrix} \succ 0.$$

The sampling technique presented in Section III can then be used to determine an approximately feasible solution $P \succ 0, Q \succ 0$ to these parameterized LMIs. The LPV controller matrices can be subsequently recovered using standard formulas.

C. LP-based robust controller design

We consider next a robust controller design technique based on (robust) linear programming, proposed in [20] for SISO continuous-time uncertain plants.

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 $\delta \in \mathbf{\Delta} \subset \mathbb{R}^d$, and are regrouped in vectors $a(\delta), b(\delta)$. 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 $d_{cl}(f, g, \delta)$ which is affine in the controller parameters f, g. The robust control problem discussed in [20] is then of the following type: given a target stable interval polynomial family

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

determine if there exist f, g such that $d_{cl}(f, g, \delta) \in \mathcal{F}$, for all instances of $\delta \in \mathbf{\Delta}$. 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 [20] by reducing it to a standard LP, using a vertexization argument. In the generic non-linear case discussed here this approach is no longer viable, and therefore we look at the problem as a robust convex (and actually linear) program, which is amenable to the randomized solution proposed in the next section.

III. Sampled Convex Programs and Probabilistic Robustness

We show in the sequel that the difficulty in the solution of (1)-(2) is mainly due to the fact that one insists on satisfying the problem constraints for *all* admissible instances of the uncertainty, and that this difficulty is released if a certain risk of constraint violation is tolerated.

Consider (2), and assume that the support $\mathbf{\Delta}$ for δ is endowed with a σ-algebra $\mathcal{D}$ and that a probability measure $\mathbf{P}$ over $\mathcal{D}$ is also assigned.

Definition 2 (Violation probability): Let $x \in \mathcal{X}$ be a candidate solution for (1)-(2). The *probability of violation* of x is defined as

$$V(x) \doteq \mathbf{P}\{\delta \in \Delta : \ f(x, \delta) \not\leq 0\}$$

(here, it is assumed that $\{\delta \in \Delta : \ f(x, \delta) \not\leq 0\}$ is an element of the σ-algebra $\mathcal{D}$). ⋆

For example, if a uniform (with respect to Lebesgue measure) probability density is assumed, then $V(x)$ measures the volume of 'bad' parameters δ such that the constraint $f(x, \delta) \leq 0$ is violated. Clearly, a solution x with small associated $V(x)$ is feasible for 'most' of the problem instances, i.e. it is *approximately feasible* for the robust problem. This concept of approximate feasibility seems to have been first introduced in the context of robust control in [2]. We have the following definition.

Definition 3 (ϵ-level solution): Let $\epsilon \in [0, 1]$. We say that $x \in \mathcal{X}$ is an ϵ-level robustly feasible solution if $V(x) \leq \epsilon$. ⋆

Our goal is to devise an algorithm that returns a ϵ-level solution, where ϵ is any fixed small level. To this purpose, we now introduce the sampled counterpart of the robust problem (1)-(2).

Definition 4 (Sampled convex program): Assume that N independent identically distributed samples $\delta^{(1)}, \ldots, \delta^{(N)}$ are drawn according to probability $\mathbf{P}$. The sampled counterpart of RCP is given by the convex optimization problem

$$\mathrm{RCP}_N : \quad \min_{x \in \mathcal{X}} c^T x \quad \text{subject to:} \qquad f(x, \delta^{(i)}) \leq 0, \ i = 1, \ldots, N. \tag{10}$$

Notice that RCP_N is a standard convex program with a finite number of constraints, and therefore it is usually solvable efficiently by means, for instance, of interior point techniques.

In the sequel, we shall make the assumption that RCP_N admits a unique optimal solution. Clearly, should RCP_N be unfeasible (i.e. $\cap_{i=1,\ldots,N} \left\{x : f(x, \delta^{(i)}) \leq 0\right\} \cap \mathcal{X} = \emptyset$), then RCP would be unfeasible too. Removing the uniqueness assumption entails some technical difficulties that would lead us beyond the scope of this presentation, but does not affect the main result. For the interested reader, these details as well as the proof of Theorem 1 below, are reported in the technical paper [9].

Let then $\hat{x}_N$ be the unique solution of problem RCP_N. Since the constraints $f(x, \delta^{(i)}) \leq 0$ are randomly selected, $\hat{x}_N$ is a random variable. The following fundamental theorem pinpoints the properties of $\hat{x}_N$.

Theorem 1: Fix two real numbers $\epsilon \in [0, 1]$ (level parameter) and $\beta \in [0, 1]$ (confidence parameter) and let

$$N \geq \frac{n_x}{\epsilon\beta} - 1. \tag{11}$$

Then, with probability not smaller than $1-\beta$, the randomized optimization problem RCP_N returns an optimal solution $\hat{x}_N$ which is ϵ-level robustly feasible. ⋆

In the theorem, probability $1-\beta$ refers to the probability $\mathbf{P}^N$ ($= \mathbf{P} \times \cdots \times \mathbf{P}$, n times) of extracting a 'bad' multisample, i.e. a multisample $\delta^{(1)}, \ldots \delta^{(N)}$ such that $\hat{x}_N$ does not meet the ϵ-level feasibility property. We here remark that the 'sample complexity' of the algorithm (i.e. the number N of random samples that need to be drawn in order to achieve the desired probabilistic level in the solution) scales linearly with respect to $1/\epsilon\beta$, and with respect to the number n_x of decision variables. The original semi-infinite problem is therefore replaced by a standard convex problem with many constraints. For reasonable probabilistic levels, the required number of these constraints appears to be manageable by current convex optimization numerical solvers.

Remark 1 (Role of probability $\mathbf{P}$): Probability $\mathbf{P}$ plays a double role in our approach: on one hand, it is the probability according to which the uncertainty is sampled; on the other hand, it is the probabilistic measure according to which the probabilistic levels of quality mentioned in the above theorem are assessed. In certain problems, $\mathbf{P}$ is the probability of occurrence of the different instances of the uncertain parameter δ. In other cases, it more simply represents the different importance we place on different instances. Extracting δ samples according to a given probability measure is not always a simple task to accomplish, see [10] for a discussion of this topic and polynomial-time algorithms for the sample generation in some matrix norm-bounded sets.

In some applications (see e.g. [8]), probability $\mathbf{P}$ is not explicitly known and the sampled constraints are directly made available as observations. In this connection, it is important to note that the bound (11) is probability independent (i.e. it holds irrespective of the underlying probability $\mathbf{P}$) and can therefore be applied even when $\mathbf{P}$ is unknown. □

Remark 2 (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 other cases, the randomized approach offers a practicable way of proceeding in order to compute a solution.

Even when solving the RCP problem is possible, the randomized approach can offer advantages that should be considered when choosing a solution methodology. In fact, solving RPC gives 100% deterministic guarantee that the constraints are satisfied, no matter what $\delta \in \mathbf{\Delta}$ is. Solving RCP_N leaves instead a chance to the occurrence of δ's which are violated by the solution. On the other hand, RCP_N provides a solution (for the satisfied constraints) that outperforms the solution obtained via RCP in terms of achieved optimal objective value. In this context, fixing a suitable level ϵ is sometimes a matter of trading probability of unfeasibility against performance. $\square$

Remark 3 (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 $N \geq n/\epsilon\beta - 1$ samples are drawn, then (with probability not smaller than $1-\beta$) the solution of the randomized program will be ϵ-level robustly feasible. However, the a-priori parameters ϵ, β are generally chosen 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 $x = \hat{x}_N$ is fixed), one can make an a-posteriori assessment of the level of feasibility using 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 $\hat{V}_{\tilde{N}}(\hat{x}_N)$, is computed according to the formula $\hat{V}_{\tilde{N}}(\hat{x}_N) = \frac{1}{\tilde{N}}\sum_{i=1}^{\tilde{N}} 1(f(\hat{x}_N, \delta^{(i)})) \leq 0)$, where $1(\cdot)$ is the indicator function. Then, the classical Hoeffding's inequality, [18], guarantees that $|\hat{V}_{\tilde{N}}(\hat{x}_N) - V(\hat{x}_N)| \leq \tilde{\epsilon}$ holds with confidence greater than $1-\tilde{\beta}$, provided that

$$\tilde{N} \geq \frac{\log 2/\tilde{\beta}}{2\tilde{\epsilon}^2} \tag{12}$$

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

IV. Numerical Example: Robust pole assignment

We next consider a modification of an example originally proposed in [20] concerning a fixed-order robust controller design for a SISO plant. Consider the setup introduced in Section II-C, 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 acting respectively on the dc-gain, the numerator damping, and the pole locations of the plant. In this example, we assume

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

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 $b_0(\delta) = 2(1+\delta_1)$, $b_1(\delta) = 3(1+\delta_1)(1+\delta_2)$, $b_2(\delta) = 2(1+\delta_1)$, $a_0(\delta) = -0.236(2+\delta_3)(1+\delta_4)$, $a_1(\delta) = -(2+\delta_3)(1+\delta_4) + 0.236(\delta_4 - \delta_3) - 0.236$, $a_2(\delta) = \delta_4 - \delta_3 - 0.764$. Define now the following target stable interval polynomial family

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

with

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

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

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

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

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

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

$$c^- \leq A(\delta)x + q(\delta) \leq c^+, \quad \forall \delta \in \boldsymbol{\Delta}. \tag{13}$$

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 robust linear program

$$\min_x\ c^T x, \quad \text{subject to } (13). \tag{14}$$

We remark that the solution approach of [20] cannot be applied in this case, since the coefficients $a_i(\delta), b_i(\delta)$ do not lie in independent intervals. We therefore apply the proposed probabilistic solution method: Assuming a uniform density over $\boldsymbol{\Delta}$ and fixing the risk level parameter ϵ to $\epsilon = 0.01$ and the confidence parameter to $\beta = 0.01$, we determine the sample bound according to Theorem 1:

$$N \geq \frac{3}{0.01 \times 0.01} - 1 = 29,999.$$

Then, N independent samples $\delta^{(1)}, \ldots, \delta^{(N)}$ are generated, and the robust problem (14) is substituted by its sampled counterpart

$$\min_x \quad c^T x, \quad \text{subject to}$$
$$c^- \leq A(\delta^{(i)})x + q(\delta^{(i)}) \leq c^+,\ i = 1, \ldots, N.$$

The numerical solution of the above sampled linear program yielded the solution $\hat{x}_N = [9.0993\ \ 19.1832\ \ 11.7309]^T$, and hence the controller

$$C(s) = \frac{9.0993 + 19.1832s}{11.7309 + s}.$$

Once we solved the synthesis problem, we can proceed to an a-posteriori Monte-Carlo test, in order to obtain a more refined estimate of the probability of constraint violation for the computed solution. As discussed in Remark 3, 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 (12) we obtain that the test should be run using at least $\tilde{N} = 6.103 \times 10^6$ samples. This yielded an estimated violation probability $\hat{V}_{\tilde{N}}(\hat{x}_N) = 0.00074$, and, from Hoeffding inequality, we have that $|\hat{V}_{\tilde{N}}(\hat{x}_N) - V(\hat{x}_N)| \leq 0.001$ holds with confidence greater than 99.999%. From a practical point of view, we can hence claim that the above robust controller has violation probability which is at most 0.00174, i.e. it satisfies more than 99.8% of the design constraints (15). We also notice that the absolute value of the largest violation of these constraints in our Monte-Carlo analysis was about 0.7.

V. Conclusions

This paper presented a novel approach to robust control design. 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 samples required to determine a solution that guarantees a-priori probabilistic robustness level ϵ, with confidence $1 - \beta$. This bound holds for generic convex problems, it is distribution-independent, and requires virtually no assumption on the 'structure' or functional dependence of the data on the uncertainty.

VI. Acknowledgments

This work was supported in part by FIRB funds from MIUR, Italy.

VII. References

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

[2] 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.

[3] 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.

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

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

[6] 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.

[7] 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.

[8] 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.

[9] G. Calafiore and M.C. Campi. Uncertain convex programs: randomized solutions and confidence levels. *To appear in Mathematical Programming*, 2004.

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

[11] 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.

[12] 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.

[13] Y. Fujisaki, F. Dabbene, and R. Tempo. Probabilistic robust design of LPV control systems. In *Proc. IEEE Conf. on Decision and Control*, pages 2019–2024, Orlando, FL, December 2001.

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

[15] 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.

[16] 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.

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

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

[19] S. Kanev, B. De Schutter, and M. Verhaegen. The ellipsoid algorithm for probabilistic robust controller design. In *Proc. IEEE Conf. on Decision and Control*, Las Vegas, December 2002.

[20] 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.

[21] 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.

[22] 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.

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

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

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

[26] 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.

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

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