Automatica 47 (2011) 1279–1293

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Survey paper

Research on probabilistic methods for control system design☆

Giuseppe C. Calafiore [a], Fabrizio Dabbene [b], Roberto Tempo [b,*]

[a] Dipartimento di Automatica e Informatica, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy

[b] IEIIT-CNR, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy

ARTICLE INFO

Article history:
Received 29 April 2009
Received in revised form
6 December 2010
Accepted 21 January 2011
Available online 25 March 2011

Keywords:
Randomized algorithms
Control of uncertain systems
Optimization

ABSTRACT

A novel approach based on probability and randomization has emerged to synergize with the standard deterministic methods for control of systems with uncertainty. The main objective of this paper is to provide a broad perspective on this area of research known as "probabilistic robust control", and to address in a systematic manner recent advances. The focal point is on design methods, based on the interplay between uncertainty randomization and convex optimization, and on the illustration of specific control applications.

1. Introduction

The study of robustness of complex systems began in the eighties based on a deterministic description of the uncertainty acting on the system to be controlled; see the books Barmish, 1994; Zhou, Doyle, & Glover, 1996 for historical perspectives. In this direction, significant progresses have been achieved. In particular, we recall the fundamental results related to the computation of the robustness margin. In this context, the study of parametric uncertainty (Barmish, 1994) played a key role, also due to its links with the theory of uncertain polynomials that originated from Kharitonov's Theorem. In subsequent years researchers realized that these deterministic methodologies were affected by serious computational problems, especially for uncertainty entering in a nonlinear fashion into the control system. For this reason, approximation and relaxation techniques have been proposed, so that these methods have been generally well accepted by the control community, but it is clear that they necessarily entail some conservatism. The outcome is that one can usually determine only upper and lower bounds of the robustness margin.

Motivated by these considerations, probabilistic and randomized methods have been developed as effective tools to deal with uncertain complex systems. For an in-depth treatment of this research area, historical perspectives and pointers to the body of related literature, the reader may refer to the books Calafiore & Dabbene, 2006; Tempo, Calafiore, & Dabbene, 2005 and to the papers (Calafiore, Dabbene, & Tempo, 2007; Dabbene & Tempo, 2010; Tempo & Ishii, 2007). The starting point of these methods is to assume that the uncertainty affecting the system has a probabilistic nature, an assumption that appears to be natural in many applications. The objective is then to provide (probabilistic) assessments on the system characteristics. More precisely, we say that a given performance level is robustly satisfied (in a probabilistic sense) if it is guaranteed against *most*, albeit not *all*, possible uncertainty outcomes. In other words, one accepts the risk of a system property being violated for a set of uncertainties having "small" probability measure. Such systems may be viewed as being "practically robust" from an engineering point of view. The probabilistic approach is of course not limited to control problems, but appears to be useful in a wider range of related areas, such as robust optimization and general engineering design, where decisions that work satisfactorily well in an uncertain or adversarial environment should be devised.

One of the advantages of the probabilistic approach for control is to provide a reapproachement between the stochastic and the robust paradigm, utilizing classical worst-case bounds of robust control together with probabilistic information, which is often neglected in a deterministic context. The probabilistic approach has also connections with adaptive control methods. In a sense, these methodologies deal with uncertainty, but instead of covering all plants within the allowed uncertainty, they try to identify which specific plant is involved. The interplay of probability and robustness leads to innovative concepts such as the probabilistic robustness margin and the probability degradation function. However, it should be noted that moving on from deterministic to probabilistic robustness does not imply a simplification of the

☆ This work has been funded in part by PRIN 20087W5P2K from Italian Ministry of University and Research. This paper was not presented at any IFAC meeting. This paper was recommended for publication in revised form under the direction of Editor Manfred Morari.

* Corresponding author.
E-mail addresses: giuseppe.calafiore@polito.it (G.C. Calafiore), fabrizio.dabbene@polito.it (F. Dabbene), tempo@polito.it (R. Tempo).

0005-1098/$ – see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2011.02.029

problem. Indeed, assessing probabilistic robustness may be even computationally harder than establishing robustness in the usual deterministic sense, since it requires the computation of multi-dimensional probability integrals. These integrals can be evaluated exactly only in very special cases of limited practical interest.

This computational problem is resolved by means of randomized techniques, which have been used extensively in various branches of science and engineering to tackle difficult problems that are too hard to be treated via exact deterministic methods. Specific examples include the Monte Carlo methods in computational physics, simulations, financial risk analysis, and the Las Vegas techniques in computer science. In the context of systems and control, the key idea is to play with "uncertainty randomization", and this requires the development of specific techniques for generating random samples of the structured uncertainty acting on the system. The probability is estimated using a finite number of random samples, and tail inequalities are used to bound the estimation error. Since the estimated probability is itself a random quantity, this method always entails a risk of failure, i.e. there exists a nonzero probability of making an erroneous estimation. The resulting algorithms are called *randomized algorithms* (RAs), i.e. algorithms that make random choices during execution to produce a result. It has been demonstrated that, in the context of systems and control, RAs have low complexity and are associated with robustness bounds which are less conservative than the classical ones, obviously at the expense of a probabilistic risk.

Besides analysis problems, the probabilistic approach unveils its full potential in the context of control systems design. In this endeavor, unlike analysis problems, the system is not fixed a priori but depends on some parameters (for instance, parameters defining the controller), that need to be determined in order to make the system behave as desired. In this respect, a randomized algorithm for design should be able to determine these parameters to guarantee the desired system specifications up to a given level of probability. The recently developed MATLAB toolbox RACT, see Tremba et al. (2008), may facilitate the control engineer to perform this task.

This paper is structured as follows: In Section 2 we introduce the main concepts through two illustrative examples dealing with probabilistic analysis and design, respectively. In Section 3 we discuss randomized estimates of probabilities and extrema, and we discuss the related sample size bounds and probability tail inequalities. In Section 4 we move on to the core of the paper, which is probabilistic design. Subsequently, in Sections 5–7 we study sequential methods, the scenario approach and statistical learning techniques, respectively. In Section 8, we describe a number of control applications. Conclusions are drawn in Section 9.

2. Preliminary concepts through examples

To motivate subsequent developments, we discuss two illustrative examples dealing, respectively, with $\mathcal{H}_\infty$ performance of a system affected by parametric uncertainty and with input design in uncertain model predictive control.

2.1. Probabilistic measures of performance

Consider the linear system

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w; \qquad z = \begin{bmatrix} 1 & 0 \end{bmatrix} x \tag{1}$$

with parameters $a_0 = \bar{a}_0 + q_0$, $a_1 = \bar{a}_1 + q_1$, and uncertainty vector $q \doteq [q_0\ q_1]^\top$ that belongs to the set

$$\mathbb{Q} = \{q \in \mathbb{R}^2 : \|q\|_\infty \le \rho\}$$

for some positive ρ (i.e. $|q_0| \le \rho$, $|q_1| \le \rho$), and nominal values $\bar{a}_0 = 1$, $\bar{a}_1 = 0.8$.

Suppose that we are interested in computing the peak of the modulus of the frequency response on the w–z channel. When the system is stable, this peak is given by the $\mathcal{H}_\infty$ norm of the transfer function $G_q(s)$ of this channel, which is denoted as $\|G_q\|_\infty = \sup_\omega |G_q(j\omega)|$; see Doyle, Francis, and Tannenbaum (1992). Given a level $\gamma \ge 1$, a (deterministic) robustness analysis problem requires, for instance, to verify whether the system is stable and the performance level $\|G_q\|_\infty \le \gamma$ is guaranteed for all $q \in \mathbb{Q}$. That is, we need to verify if the specification

$$G_q(s) \text{ is stable and } \|G_q\|_\infty \le \gamma \tag{2}$$

is robustly satisfied for all $q \in \mathbb{Q}$.

Let for instance $\gamma = \sqrt{2}$. Since (1) is a simple second order system, it is straightforward to verify that (2) is satisfied for all $q \in \mathbb{Q}$ if and only if $\rho < 0.8$ and

$$\frac{(0.8-\rho)^2}{2-\sqrt{2}} > 1 + \rho,$$

i.e., for $\rho < \bar{\rho} \doteq 0.025$; see Fig. 1(a). The limit value $\bar{\rho}$ for the uncertainty radius is called the *deterministic robustness radius* for the performance specification (2); see Barmish (1994).

Note that, from the point of view of the worst-case approach to robustness, the system ceases to satisfy the required robust performance specification for $\rho > \bar{\rho}$. In this sense, worst-case robustness analysis gives a yes/no answer, and provides no information on the system behavior for uncertainty levels beyond $\bar{\rho}$. Probabilistic analysis can nicely complement the information provided by the deterministic approach, or can be a valid alternative when the deterministic approach cannot be applied. Assume for instance that q_0, q_1 are random variables, independent and uniformly distributed in the interval $[-\rho, \rho]$. Then, for $\rho > \bar{\rho}$, some uncertainty realizations violate the given specifications. However, in practice, a violation might be tolerated, if it happens "rarely". To quantify how rare is the violation, we use probabilities. This concept of *approximate* constraint satisfaction has been studied in Barmish and Shcherbakov (2002). More precisely, with reference to Fig. 1(b), we define the *violation set*

$$\mathbb{Q}_{\text{viol}} = \{q \in \mathbb{Q} : (2) \text{ is violated}\}.$$

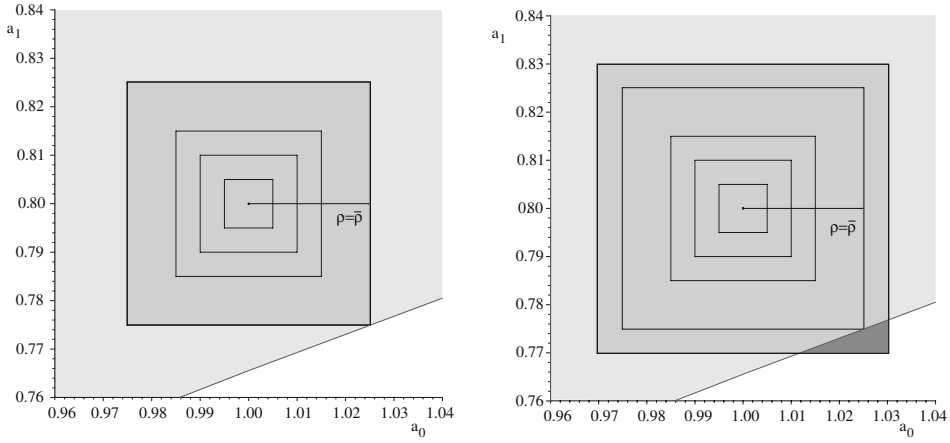
The *violation probability* V is the probability measure of $\mathbb{Q}_{\text{viol}}$. In the case of uniform measure in the set $\mathbb{Q}$, this probability is expressed as the ratio of areas

$$V = \frac{\text{area of } \mathbb{Q}_{\text{viol}}}{\text{area of } \mathbb{Q}}.$$

The probabilistic robustness level, or *reliability*, of the uncertain system performance is hence given by $R = 1 - V$.

In this example, the reliability can be computed explicitly for various values of ρ, and the curve shown in Fig. 2 can be traced. We can infer useful information from this curve, which is referred to as the *probability degradation function*, (Tempo et al., 2005). For instance, we see that if a 5% probability of violation is allowed, then the tolerable uncertainty radius can be increased by about 54% with respect to the deterministic robustness radius $\bar{\rho}$.

This elementary example highlights all the relevant ingredients of probabilistic methods, that is (i) an uncertainty set $\mathbb{Q}$; (ii) a probability measure Prob defined over $\mathbb{Q}$; (iii) an index of performance defined on the uncertain system. The focal point in probabilistic robustness analysis is indeed the evaluation of the violation probability V (or, equivalently, of the reliability R). This quantity is in general very difficult to compute exactly (except for very simple cases such as the one in the previous example). However, V can be *estimated* by means of sampling techniques, as discussed in Section 3. Besides probabilistic analysis, however, a problem of major interest from an engineering viewpoint is to *design* a system so that its performance is robustly satisfied, at least in a probabilistic sense.



(a) The square of radius $\bar{\rho}$ is the maximal admissible uncertainty set around the nominal parameters values $\bar{a}_0, \bar{a}_1$.

(b) If the uncertainty radius is extended beyond $\bar{\rho}$, a subset of $\mathbb{Q}$ violates the performance specifications (darker area).

Fig. 1. Worst case vs. probabilistic robustness. The light shaded region contains coefficient values for which the system is stable and $\|G_q\|_\infty \leq \sqrt{2}$.

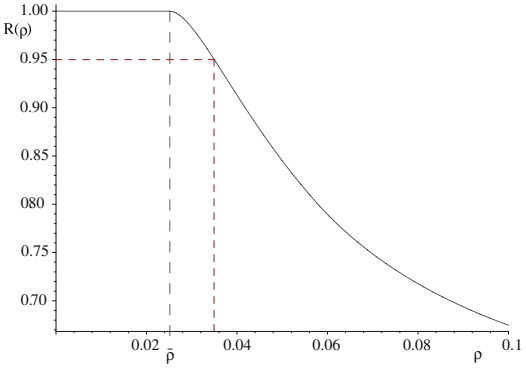


Fig. 2. Degradation of the reliability level as a function of the uncertainty radius ρ.

2.2. Probabilistic design

Consider the linear system

$$x(k+1) = Ax(k) + Bu(k), \quad x(0) = x_0$$
$$y(k) = Cx(k)$$

where $x \in \mathbb{R}^n$ is the system state, $u \in \mathbb{R}^{n_u}$ is the control input, and $y \in \mathbb{R}^{n_y}$ is the system output. A typical problem that is addressed in the model predictive control (MPC) literature (see, e.g., Camacho & Bordons, 2003) is to determine a sequence of control actions $u(0), u(1), \ldots, u(T-1)$, such that a suitable performance index is minimized over a finite horizon T, while satisfying a given set of constraints on the input and output signals. This problem may take for instance the following form

$$\begin{array}{ll} \min \eta & \text{subject to :} \\ & J(u(0), u(1), \ldots, u(T-1)) \leq \eta \\ & y_{\min} \leq y(k) \leq y_{\max}, \qquad k = 1, \ldots, T \\ & u_{\min} \leq u(k) \leq u_{\max}, \qquad k = 0, \ldots, T-1, \end{array} \tag{3}$$

where J is a quadratic cost function

$$J(u(0), u(1), \ldots, u(T-1)) = \sum_{k=0}^{T-1} \left(x^\top(k) Q x(k) + u^\top(k) R u(k)\right) + x^\top(T) P x(T),$$

with Q, R, P given positive definite matrices. Problem (3) is a convex quadratic programming problem, hence the optimal input sequence can be efficiently computed by numerical methods.

An interesting variation on this problem arises when the system matrices A, B, C are not known exactly. Indeed, if the entries of $A(q), B(q), C(q)$ are nonlinear functions of a vector of random parameters $q \in \mathbb{Q}$, then the constraints in problem (3) need be enforced in some "robust" sense. A sensible approach is for instance to ask whether the command and output constraints are met with high probability, that is for most (if not all) possible realizations of q. Formally, denoting the design variables with θ (here, $\theta \doteq [\eta\, u^\top(0)\, u^\top(1) \ldots u^\top(T-1)]^\top$), we rewrite the constraints in problem (3) as

$$f(\theta, q) = \max \Bigg\{ J - \eta, \max_{k=1,\ldots,T} \{y(k) - y_{\max}, y_{\min} - y(k)\}, \max_{k=0,\ldots,T-1} \{u(k) - u_{\max}, u_{\min} - u(k)\} \Bigg\}.$$

We introduce the probability of violation for the constraints at θ as

$$V(\theta) = \text{Prob}\{q \in \mathbb{Q} : f(\theta, q) > 0\}.$$

Then, fixing a probability level $\epsilon \in (0, 1)$, we say that a design θ is a probabilistically feasible design to level ϵ if it satisfies $V(\theta) \leq \epsilon$. The main objective of probabilistic design methods is to devise algorithms that are capable of returning probabilistically feasible solutions to uncertain design problems.

3. Estimation of probability and extrema

Let a performance function $f(q)$ be defined for a generic uncertain dynamic system, where $q \in \mathbb{Q}$ is a vector of random uncertain parameters, and $\mathbb{Q} \subseteq \mathbb{R}^\ell$ is a given uncertainty domain. For instance, for the example in Section 2.1, we take

$$f(q) = \begin{cases} +\infty & \text{if } G_q \text{ is unstable} \\ \|G_q\|_\infty & \text{otherwise.} \end{cases}$$

We define two probabilistic analysis problems.

Problem 1 (*Reliability Estimation*)**.** Given $\gamma > 0$, estimate the reliability of the specification $f(q) \leq \gamma$, that is evaluate

$$R = \text{Prob}\{f(q) \leq \gamma\}. \tag{4}$$

Problem 2 (*Performance Level Estimation*)**.** Given $\epsilon \in (0, 1)$, estimate a performance level γ such that $f(q) \leq \gamma$ holds with reliability at least $1 - \epsilon$, that is find γ such that

$$\text{Prob}\{f(q) \leq \gamma\} \geq 1 - \epsilon. \tag{5}$$

An approximate solution to Problem 1 can be easily determined by means of a Monte Carlo approach as follows: Extract N independent identically distributed (i.i.d.) samples $q^{(1)}, \ldots, q^{(N)}$ of the random vector q, according to the assumed probability distribution Prob on $\mathbb{Q}$, and compute the *empirical reliability*

$$\hat{R}_N = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}(f(q^{(i)}) \leq \gamma), \tag{6}$$

where the indicator function $\mathbb{I}(\cdot)$ is one if the clause is true, and it is zero otherwise. Note that, from the law of large numbers, we know that $\hat{R}_N \to R$ as the number N of samples goes to infinity. However, in practice, it is important to know a priori how good (or how bad) is the estimate $\hat{R}_N$ of R when a *finite* and given number of samples is employed. Such an assessment is provided by the Hoeffding inequality (Hoeffding, 1963), which states that for given $\epsilon > 0$

$$\text{Prob}^N\{|\hat{R}_N - R| \geq \epsilon\} \leq 2e^{-2N\epsilon^2}, \tag{7}$$

where Prob^N is the product probability measure $\text{Prob} \times \text{Prob} \times \cdots \times \text{Prob}$ (N times). Hence, if we set a priori the accuracy $\epsilon \in (0, 1)$ and a confidence level $\delta \in (0, 1)$, that is we set $2e^{-2N\epsilon^2} \leq \delta$, then we can "invert" this inequality for N, and obtain the so-called (additive) Chernoff bound (Chernoff, 1952) for the sample complexity

$$N \geq \frac{1}{2\epsilon^2}\log\frac{2}{\delta}. \tag{8}$$

In other words, this means that if the number of samples used in (6) satisfies (8), then $\hat{R}_N$ will be ϵ-close to R, except for very unfortunate experiments, that may happen with probability smaller than δ.

Similarly, the following Monte Carlo approach provides an approximate solution to Problem 2: Extract N i.i.d. samples $q^{(1)}, \ldots, q^{(N)}$, of the random vector $q \in \mathbb{Q}$, according to Prob, and compute the *empirical performance level*

$$\hat{\gamma}_N = \max_{i=1,\ldots,N} f(q^{(i)}).$$

Also in this case it is possible to determine an a priori bound on the number of uncertainty samples necessary for guaranteeing that (5) holds with high probability $(1 - \delta)$. In fact, in Tempo, Bai, and Dabbene (1997) it is shown that if

$$N \geq \frac{\log\frac{1}{\delta}}{\log\frac{1}{1-\epsilon}} \simeq \frac{1}{\epsilon}\log\frac{1}{\delta} \quad \text{for small } \epsilon, \tag{9}$$

then (5) holds with probability larger than $(1 - \delta)$; see also Chen and Zhou (1998) for further extensions.

Note that in both bounds (8) and (9), the level δ can be set to a very small number (say, $\delta = 10^{-9}$) without making the required number of samples grow too much. Note further that both expressions depend polynomially on the probabilistic levels and are *independent* of the dimension of the uncertainty vector q. This is one of the main reasons that makes randomized algorithms appealing for control applications: they are easily implementable and run in "polynomial time" (assuming that the cost of sample extraction is also polynomial). Moreover, the above randomized schemes are intrinsically parallelizable. Algorithms for computing $\hat{R}_N$ and $\hat{\gamma}_N$ on a distributed computing architecture are proposed and analyzed in Calafiore (2009a).

Remark 1 (*Random Sample Generation*)**.** A successful implementation of this randomized estimation approach requires the availability of efficient algorithms for the generation of i.i.d. random samples according to the distribution Prob. This issue has been the subject of active research, since standard univariate generation techniques discussed e.g. in Devroye (1986) are not readily extendable to the sets usually encountered in robust control, and standard rejection methods can hardly be used for these sets, due to their inefficiency; see the details in Calafiore, Dabbene, and Tempo (2000). Methods for generating uniform (or radially symmetric) samples in the ℓ_p vector norm ball are discussed in Tempo et al. (2005). In the papers (Calafiore & Dabbene, 2002; Calafiore et al., 2000) methods for random sample generation in real and complex spectral norm balls are developed; see also Zhou and Feng (2006) for generation of matrices having a Toeplitz structure. Also, in Chen, Zhou, and Aravena (2004) a sample reuse technique is proposed for improving efficiency when samples with increasing norm need to be generated. The generation of causal stable dynamic operator has been the subject of different studies. In paper (Shcherbakov & Dabbene, in press) different techniques for the generation of stable polynomials are presented. Sampling techniques for transfer functions in $\mathcal{RH}_\infty$ are discussed in Calafiore and Dabbene (2004) and Sznaier, Lagoa, and Mazzaro (2005).

3.1. Choice of probability measure

Note that, in the probabilistic setting described in this paper, it is not necessary to know the probability measure of the uncertainty. Indeed, all that is needed is the availability of samples drawn from this probability, and the stated results hold irrespective of the probability measure. This is important, since in some practical situations the samples can be directly acquired through measurements or experiments on the real plant.

In other cases, when samples are not directly available, one needs to generate them, and hence a probability measure on the uncertainty set has to be assumed. In this situation, clearly, the system reliability R defined in (4) depends on the specific choice of this measure. In extreme cases, this probability may vary between zero and one, when considering different measures. Therefore, without any guideline on the choice of the measure, the obtained probability estimate may be meaningless. Depending on the application at hand, the "right" probability measure may be estimated directly from available data, or elicited from a priori knowledge. In any case, this selection should be performed with great care. Some of these issues have been studied, for instance, in Bai, Tempo, and Fu (1998) and Barmish and Lagoa (1997); see also Lagoa (2003) and Lagoa, Shcherbakov, and Barmish (1998). In particular, (Barmish & Lagoa, 1997) introduces the notion of "distribution-free robustness", with the objective of determining a worst-case measure in a given class of distributions. In many cases of interest, with bounded uncertainty set $\mathbb{Q}$, the *uniform* probability distribution possesses such worst-case properties, and this distribution is indeed often used in practice, when little is known about the actual distribution. In other cases (such as, for instance, when one knows that the uncertainty is concentrated around the mean), the traditional Gaussian distribution (or truncated Gaussian) can be used.

3.2. Other simulation-based methods

The probabilistic estimation techniques discussed in Section 3 are indeed an application of the classical Monte Carlo method; see Fishman (1996) and Rubinstein and Kroese (2008). A key feature in this approach is that the random samples used for estimation are *independent* and identically distributed (i.i.d.) according to some *fixed* probability distribution. This hypothesis permits us to obtain explicit analytic bounds on the number of samples that are required to achieve a desired confidence in the estimate, such as the bound in (8). However, on the one hand it may be very difficult to generate truly i.i.d. samples according to the desired

distribution, and on the other hand this distribution itself could hardly be elicited in advance. The outcome could be an underestimate the true risk of system failure. To resolve the first issue, specific techniques have been developed for uniform generation in the sets usually encountered in the robust control literature, as discussed in Remark 1. For more general sets, asymptotic sampling techniques that progressively reach a steady-state distribution that coincides with the desired target distribution are usually employed. These latter methods are broadly known under the name of Markov Chain Monte Carlo (MCMC) methods; see e.g. Spall (2003a) and the recent developments in Lecchini-Visintini, Lygeros, and Maciejowski (2010). We remark that these methods do not produce i.i.d. samples at steady state, and also the "time" needed for such algorithms in order to reach steady state is not usually known in advance. Hence, explicit analytic results for finite-sample complexity are not generally available. Among these methods we recall the Metropolis–Hastings algorithm (Hastings, 1970; Spall, 2003b), and the Hit-and-Run method for generation of samples in convex bodies, according to uniform and other distributions, (Kalai & Vempala, 2006; Lovász, 1999; Smith, 1984). Also, the so-called "importance sampling" techniques have been developed (see Fishman, 1996; Rubinstein & Kroese, 2008) to progressively shift the sampling distribution towards the failure region, so as to gain information from rare events more efficiently.

As can be seen from (8), the number of samples required to estimate the unknown reliability R (or, equivalently, the violation probability $V = 1 - R$) within ϵ accuracy grows as $1/\epsilon^2$, and this can be quite a large number, if high accuracy is desired. Moreover, if the probability V that we are trying to estimate is itself very small (rare-event probability), then we would need on average many samples before a "failure" occurs (that is, before a sample $q^{(i)}$ is found such that $f(q^{(i)}) > \gamma$). To show this latter fact, let $V = \text{Prob}\{f(q) > \gamma\}$ be small (rare failure), let $q^{(1)}, q^{(2)}, \ldots$, be a sequence of i.i.d. samples, and define the random variable

$$n \doteq \inf\{i = 1, 2, \ldots : f(q^{(i)}) > \gamma\}.$$

Then, we have that

$$\begin{aligned}\mathrm{E}\{n\} &= \text{Prob}\{f(q^{(1)}) > \gamma\} + 2\text{Prob}\{f(q^{(1)}) \le \gamma, f(q^{(2)}) > \gamma\} \\ &\quad + \cdots + k\text{Prob}\{f(q^{(1)}) \le \gamma, \ldots, f(q^{(k-1)}) \\ &\le \gamma, f(q^{(k)}) > \gamma\} + \cdots \\ &= \sum_{k=0}^{\infty} k(1-V)^{k-1}V = \frac{V}{1-V}\sum_{k=0}^{\infty} k(1-V)^k = \frac{1}{V},\end{aligned}$$

where E denotes statistical expectation. Finding small probabilities thus requires information from rare samples corresponding to failures and, on average, $1/V$ samples are needed before a failure is detected.

Other methods for computing small failure probabilities for certain classes of dynamical systems subject to stochastic excitation include the "subset simulation" methods (Au & Beck, 2001; Ching, Au, & Beck, 2005), which are based on the idea of factoring the failure probability in the product of larger conditional failure probabilities that can be estimated with lower computational effort, thus replacing the problem in the original probability space by a sequence of simulations of more frequent events in the conditional probability spaces.

4. Probabilistic design

Probabilistic design techniques are based on the interplay of random sampling in the uncertainty space, and deterministic optimization in the design parameter space. Formally, let $\theta \in \mathbb{R}^{n_\theta}$ denote the vector of design parameters, and define a *performance function* that takes into account the design and performance constraints related to the uncertainty system. These are rewritten in the form of the following (uncertain) inequality

$$f(\theta, q) \le 0, \tag{10}$$

where $f(\theta, q) : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \mathbb{R}$ is a scalar-valued function. A design vector θ such that inequality (10) is satisfied "for most" (in a probabilistic sense) of the outcomes of q is called a probabilistic robust design. More precisely, we have the following definition.

Definition 1. Let $\epsilon \in (0, 1)$ be a given probability level, and let $\theta \in \mathbb{R}^{n_\theta}$ be a given design vector. We define the *probability of violation* for θ as

$$V(\theta) \doteq \text{Prob}\{q \in \mathbb{Q} : f(\theta, q) > 0\} \tag{11}$$

and the *reliability* of the design θ as $R(\theta) \doteq 1 - V(\theta)$. We say that θ is a probabilistic robust design to reliability level $(1 - \epsilon)$, or in short *ϵ-reliable design*, if $V(\theta) \le \epsilon$ or, equivalently, $R(\theta) \ge 1 - \epsilon$.

Most of the results presented in the literature for solving the probabilistic design problem have been derived under the assumption that the function $f(\theta, q)$ is *convex* in θ for all $q \in \mathbb{Q}$. This assumption is used in Sections 5 and 6 and is now formally stated. Section 7 discusses instead the nonconvex case, which is approached using statistical learning techniques; see Vapnik (1998) and Vidyasagar (2002). A different "mixed" approach for control design that falls outside these classes is analyzed in Fujisaki, Oishi, and Tempo (2008).

Assumption 1 (*Convexity*)**.** The function $f(\theta, q)$ is convex in θ for any fixed value of $q \in \mathbb{Q}$.

Note that the previous assumption requires convexity only with respect to the design variable θ, while generic nonlinear dependence with respect to q is allowed. A standard example of convex function f arises when considering performance requirements expressed as uncertain linear matrix inequality (LMI) conditions, (El Ghaoui & Lebret, 1998; Scherer, Gahinet, & Chilali, 1997).

In detail, a robust LMI feasibility problem is expressed in the form

$$\text{Find } \theta \text{ such that } F(\theta, q) \preceq 0 \quad \forall q \in \mathbb{Q},$$

where $F(\theta, q) \preceq 0$ means that $F(\theta, q)$ is negative semi-definite. This constraint is an LMI in θ for fixed q, that is

$$F(\theta, q) = F_0(q) + \sum_{i=1}^{n_\theta} \theta_i F_i(q), \tag{12}$$

and $F_i(q)$, $i = 0, \ldots, n_\theta$, are symmetric real matrices of appropriate dimensions, that depend in a possibly nonlinear way on the uncertainty $q \in \mathbb{Q}$. This problem is rewritten in the scalar-function framework (10) by setting, for instance,

$$f(\theta, q) \doteq \lambda_{\max}(F(\theta, q)),$$

where $\lambda_{\max}$ denotes the largest eigenvalue. The following alternative choice in the LMI case was proposed in Calafiore and Polyak (2001) and Polyak and Tempo (2001)

$$f(\theta, q) = \|[F(\theta, q)]^+\|, \tag{13}$$

where $[A]^+$ denotes the projection on the cone of positive definite matrices, i.e.

$$[A]^+ \doteq \arg\min_{X \in \mathcal{C}} \|A - X\|,$$

and $\mathcal{C} \doteq \left\{X \in \mathbb{R}^{n,n} : X = X^\top \succeq 0\right\}$.

Finally, we remark that considering scalar-valued constraint functions is without loss of generality, since multiple constraints $f_1(\theta, q) \le 0, \ldots, f_{n_f}(\theta, q) \le 0$ can be reduced to a single scalar-valued constraint by simply setting $f(\theta, q) = \max_{i=1,\ldots,n_f} f_i(\theta, q)$.

Note that if functions $f_i(\theta, q)$ are convex in θ, then also the pointwise maximum $f(\theta, q)$ is convex in θ. We can now define the two problems that we aim to solve. The first amounts to the determination of an ϵ-reliable design, that is a feasibility problem with constraints expressed in probability (chance constraints).

Problem 3 (*Find an ϵ-Reliable Design*)**.** Given a violation level $\epsilon \in (0, 1)$, determine $\theta \in \mathbb{R}^{n_\theta}$ such that $V(\theta) \leq \epsilon$.

The second problem relates to the optimization of a linear function of the design parameter under a probability constraint.

Problem 4 (*Optimize an ϵ-Reliable Design*)**.** Given a violation level $\epsilon \in (0, 1)$, and an objective vector $c \in \mathbb{R}^{n_\theta}$, solve

$$\min_{\theta \in \mathbb{R}^{n_\theta}} c^\top \theta \quad \text{subject to } V(\theta) \leq \epsilon. \tag{14}$$

We stress again that both these problems are possibly nonconvex (despite convexity of f) and numerically hard to solve exactly, in general. The randomized algorithms we discuss in the following sections provide a numerically viable way to compute probabilistic approximate solutions for the above problems.

5. Sequential design methods

In this section, we introduce a unifying theoretical framework that encompasses most of the sequential algorithms appeared so far in the literature for the solution of Problem 3. All these algorithms follow a general iterative scheme, which consists of successive randomization steps to handle uncertainty, and optimization steps to iteratively update the design parameters.

We first state the following definition.

Definition 2 (*r-Feasibility*)**.** For given $r > 0$, we say that Problem 3 is r-feasible if the solution set

$$\mathcal{S} = \left\{\theta \in \mathbb{R}^{n_\theta} : f(\theta, q) \leq 0,\ \forall q \in \mathbb{Q}\right\}$$

contains a full-dimensional ball $\mathcal{B}(r)$ of radius r called the r-feasibility ball.

A careful analysis shows that the algorithms presented in the literature for finding a probabilistic feasible solution share two fundamental ingredients: (i) an *ϵ-probabilistic oracle* (shortly, the ϵ-oracle), which has the purpose of checking whether the violation probability of the current candidate solution is less than ϵ, and (ii) an *update rule* which exploits the convexity of the problem for constructing a new candidate solution based on the oracle outcome. With these ingredients at hand, these iterative schemes can be recast in the form of the following meta-algorithm.

Algorithm 1 Sequential scheme for probabilistic design

1. INITIALIZATION Set $k = 0$ and choose an initial candidate solution θ_k.
2. ϵ-Oracle Invoke the ϵ-oracle with θ_k
   - If the ϵ-oracle returns `eps_reliable`, then return $\theta_{\text{pr}} = \theta_k$, and exit.
   - Otherwise, the ϵ-oracle returns `unfeas`, together with a *violation certificate* q_k, that is a realization of the uncertainty q such that $f(\theta_k, q_k) > 0$.
3. UPDATE Construct a new candidate solution θ_{k+1} based on θ_k and on the violation certificate q_k.
4. OUTER ITERATION Set $k = k + 1$ and goto 2.

In the next section we describe a key feature of the proposed scheme, namely that the ϵ-oracle, due to its probabilistic nature, may possibly declare as `eps_reliable` a solution θ for which $V(\theta) \leq \epsilon$ is not satisfied. However, this situation may only happen with a very low and predetermined probability.

5.1. Probabilistic oracle

The ϵ-oracle constitutes the randomized part of the algorithm, and its role is to check the feasibility of the current solution, based on random samples of the uncertainty. More precisely, a number N_k of independently identically distributed (i.i.d) uncertainty samples

$$q^{(1)}, \ldots, q^{(N_k)} \in \mathbb{Q}$$

are drawn according to the underlying distribution Prob, and the candidate design θ_k is deemed `eps_reliable` if

$$f(\theta_k, q^{(i)}) \leq 0, \quad i = 1, 2, \ldots, N_k.$$

This leads to the following simple randomized scheme.

Algorithm 2 `EpsOracle`

Input: θ_k, N_k
Output: `sol` (`eps_reliable`/`unfeas`) and violation certificate q_k

for $i = 0$ to N_k **do**
  draw a random sample $q^{(i)}$ according to Prob

  RANDOMIZED TEST

  if $f(q^{(i)}, \theta_k) > 0$ **then**
    set $q_k = q^{(i)}$, `sol=unfeas`
    return `sol`, q_k and exit
  end if
end for
return `sol=eps_reliable`

Note that at step k the ϵ-feasibility of the candidate solution θ_k is verified with respect to a *finite* number of samples N_k. If the test is passed, the solution is considered probabilistically robust, and labeled `eps_reliable`; otherwise, the uncertainty value q_k for which the randomized test failed is returned as a violation certificate. The *sample size* N_k depends on k, and has to be chosen to guarantee the desired probabilistic properties of the solution. The following theorem states that, with high probability, the point θ_k is indeed an ϵ-reliable solution.

To this end, define the *probability of misclassification* of the ϵ-oracle as the probability that the ϵ-oracle labels as `eps_reliable` a *bad* solution, i.e. a solution for which $V(\theta_k) > \epsilon$. Formally, define the event

$$\text{Misclass} \doteq \{\text{the } \epsilon\text{-oracle labels } \theta_k \text{ as } \texttt{eps\_reliable} \cap V(\theta_k) > \epsilon\}.$$

Then, the following theorem holds; see Calafiore and Dabbene (2007).

Theorem 1 (*Probability of Misclassification of the ϵ-Oracle*)**.** *Let $\epsilon \in (0, 1)$ be a given probability level, and let θ_k be a query point. Then, the probability of misclassification of the ϵ-oracle is less than $(1 - \epsilon)^{N_k}$, i.e.*

$$\text{Prob}^{N_k}\{\textit{Misclass}\} \leq (1 - \epsilon)^{N_k},$$

where Prob^{N_k} *is the product probability measure* $\text{Prob} \times \text{Prob} \times \cdots \times \text{Prob}$ *(N_k times).*

Remark 2. Note that by appropriate choice of the iterations limit N_k, we can make the probability of misclassification of the ϵ-oracle as close as desired to zero. In particular, to achieve a desired success probability $1 - \delta$, where $\delta \in (0, 1)$ is a small number, we need

$$N_k \geq N_{\text{oracle}}(\epsilon, \delta) \doteq \frac{\log \frac{1}{\delta}}{\log \frac{1}{1-\epsilon}}. \tag{15}$$

5.2. Unified analysis of the sequential scheme

The second main component of the sequential scheme (Algorithm 1) is the update rule. Various update rules have been proposed in the literature, among which we recall gradient update, ellipsoidal, and cutting plane update rules; see Section 5.3.

Next, in this section we present a unifying view for studying the probabilistic behavior of Algorithm 1. To this end, note first that the update step is completely *deterministic* and does not involve randomization. To clarify this point, consider again Algorithm 1, and suppose that an *exact robust oracle* were available, that is a deterministic oracle able to exactly discern robust feasibility of the candidate solution θ_k. Such an exact oracle would return `rob_feas` whenever $f(x, q) \leq 0$ holds robustly for all $q \in \mathbb{Q}$, and `unfeas` otherwise. In such a case, Algorithm 1 would be thoroughly deterministic, and its convergence properties could be analyzed in a non-probabilistic setting. Clearly, since inequality (10) usually involves an infinite number of constraints, one for each value of q in $\mathbb{Q}$, such an oracle can rarely be constructed in practice. However, its introduction is of theoretical importance, since it allows us to formally unify all the randomized sequential algorithms presented so far in the literature. We thus introduce the following assumption.

Assumption 2 (*Outer Convergence with Exact Robust Oracle*)**.** We assume that the update rule in Algorithm 1 is such that the following implication holds true:

if the problem is r-feasible (see Definition 2), and a robust exact oracle is available,
then Algorithm 1 converges in a finite number of outer iterations. Moreover, this number of iterations is bounded from above by a known function $N_{\text{outer}}(r)$.

Note that $N_{\text{outer}}(r)$ may also depend on other specific rule-dependent quantities.

With these positions, the meta-algorithm can be now formally stated as follows.

Algorithm 3 `SequentialDesign`

Input: $\epsilon, \delta \in (0, 1)$, N_{outer}
Output: `sol` (`eps_reliable`/`unfeas`) and θ_{pr}

INITIALIZATION

choose $\theta_0 \in \mathbb{R}^{n_\theta}$, set $k = 0$ and `sol=unfeas`

OUTER ITERATION

while `sol=unfeas` and $k < N_{\text{outer}}$ **do**
  determine the sample size N_k according to (16)
  ϵ-ORACLE

  (`sol`, q_k) = `EpsOracle`(θ_k, N_k)

  UPDATE

  if `sol=eps_reliable` **then**
    return $\theta_{\text{pr}} = \theta_k$ and exit
  else
    update $\theta_{k+1} =$ `UpdateRule`(θ_k)
  end if
  set $k = k + 1$
end while

The probabilistic properties of Algorithm 3 are formally derived in the next theorem, which constitutes a slight improvement upon original results which first appeared in Oishi (2007a) and then in Fujisaki and Oishi (2007) and Calafiore and Dabbene (2007). The proof follows the same lines of the proof of Theorem 5.3 in Dabbene, Shcherbakov, and Polyak (2010) and is omitted for brevity.

Theorem 2 (*Probabilistic Properties of Algorithm* 3)**.** *Let Assumption* 1 *hold, and let* r *be an assumed radius of* r*-feasibility. Let further* $\epsilon, \delta \in (0, 1)$ *be given probability levels, and assume that at step* k *of Algorithm* 3 *the* ϵ*-oracle is invoked with sample size*

$$N_k \geq N_{\text{oracle}}(\epsilon, \delta) + \frac{\alpha \ln k + \log \zeta(\alpha)}{\log \frac{1}{1-\epsilon}}, \quad \alpha > 1 \tag{16}$$

where $N_{\text{oracle}}(\cdot)$ *is given in* (15) *and* $\zeta(\cdot)$ *is the Riemann zeta function. Then, the following statements hold*

1. *The probability that Algorithm* 3 *terminates at some outer iteration* $k < N_{\text{outer}}$ *returning a design* θ_{pr} *which is not* ϵ*-reliable (i.e. such that* $V(\theta_{\text{pr}}) > \epsilon$*) is less than* δ.
2. *If Algorithm* 3 *reaches the outer iteration count* N_{outer}*, then the problem is not* r*-feasible.*

Remark 3 (*Number of Outer Iterations*)**.** If Algorithm 3 exits due to occurrence of case (1) of Theorem 2, then we may declare with high confidence that the solution θ_{pr} is ϵ-reliable, and this corresponds to a *successful exit* of the algorithm. If instead the algorithm exits due to the occurrence of case (2), no solution has been found (*unsuccessful exit*), but we may declare with certainty that the problem is not r-feasible. Note that, in practice, one may not know in advance whether the problem is r-feasible or not. The described randomized schemes can however be used also if the problem is not r-feasible, or even if no robustly feasible solution exists at all (empty $\mathcal{S}$). The point is that *if* Algorithm 3 terminates at some outer iteration $k < N_{\text{outer}}$, then the returned solution θ_{pr} will be ϵ-reliable, unless a rare event of small probability δ occurred. Probability δ has of course to be set to an appropriately low value, so as to make the a priori chance of occurrence of this event negligible in practice. The introduction of the $N_{\text{outer}}(r)$ limit has the purpose of guaranteeing finite-time termination of the sequential scheme in Algorithm 3, in cases when the oracle cannot find a probabilistic solution. If we do not assume r-feasibility, we cannot determine $N_{\text{outer}}(r)$, but we can still use the algorithm letting it run indefinitely, without a priori guarantee of finite termination.

Note that if we set $\alpha = 2$ in (16), then we recover the bound in Oishi (2007a). Choosing instead, for instance, $\alpha = 1.11$, we get the bound

$$N_k \geq N_{\text{oracle}}(\epsilon, \delta) + \frac{1.11 \ln k + 2.27}{\log \frac{1}{1-\epsilon}}, \tag{17}$$

which improves upon the bound in Oishi (2007a), for $k > 3$. Note that any N_k which satisfies (16) also satisfies the weaker bound (15).

It is also important to remark that the sample size N_k in (16) is independent of the number of uncertain parameters and of the dimension of the design parameter.

5.3. Update rules

All the update rules we discuss in this section assume the availability of a *subgradient* $\partial_k(\theta)$ of the function $f(\theta, q)$ at the violation certificate q_k. Note that, in the case where $f(\theta, q_k)$ is differentiable at θ, then $\partial_k(\theta)$ is simply the gradient of f, i.e. $\partial_k(\theta) = \nabla_\theta f(\theta, q_k)$.

Remark 4 (*Subgradient for LMIs*)**.** For the LMI problem defined in (12), a subgradient of the function $f(\theta, q_k) = \lambda_{\max}(F(\theta, q_k))$

at $\theta = \theta_k$ is readily computable as $\partial_k(\theta_k) = [\xi_{\max}{}^\top F_1(q_k)\xi_{\max} \cdots \xi_{\max}{}^\top F_{n_\theta}(q_k)\xi_{\max}]^\top$, where $\xi_{\max}$ is a unit norm eigenvector associated with the largest eigenvalue of $F(\theta_k, q_k)$. Similarly, a subgradient of the function defined in (13) can be computed in closed form; see Calafiore and Polyak (2001) for details.

5.3.1. Gradient update

The first update rule proposed in the literature, see for instance Calafiore and Polyak (2001) and Polyak and Tempo (2001), is also the simplest one, and is based on a (sub)gradient descent technique. This rule is summarized in Algorithm 4: the main distinguishing feature of the method lies in the particular choice of the stepsize η_k

$$\eta_k = \frac{f(\theta_k, q_k)}{\|\partial_k(\theta_k)\|} + r \tag{18}$$

where r is the assumed r-feasibility radius.

Algorithm 4 `UpdateRule` (gradient method)

Input: θ_k, q_k
Output: θ_{k+1}

compute the subgradient $\partial_k(\theta)$ of $f(\theta, q_k)$.
compute the stepsize η_k according to (17)
set $\theta_{k+1} = \theta_k - \eta_k \frac{\partial_k(\theta_k)}{\|\partial_k(\theta_k)\|}$

This update rule fulfills Assumption 2: in particular, the stepsize (18) guarantees finite convergence of the deterministic version of the algorithm (i.e. with exact oracle) in a number of steps bounded by

$$N_{\text{outer}}(r) = \left\lceil \frac{\Omega^2}{r^2} \right\rceil,$$

where Ω is an a priori upper bound on the distance of the initial solution θ_0 from the center of the r-feasibility ball $\mathcal{B}(r)$ (see Definition 2). Algorithms using this rule have been developed in different contexts: (Fujisaki, Dabbene, & Tempo, 2003; Polyak & Tempo, 2001) develop gradient schemes specifically tailored for the design of linear parameter systems (LPV) and for the solution of guaranteed cost quadratic regulator problems. Further, (Calafiore & Polyak, 2001), analyzes an LMI setting, while (Dabbene, Gay, & Polyak, 2003) is oriented to an identification context.

5.3.2. Localization methods

More sophisticated randomized algorithms, that still guarantee probabilistic properties of the ensuing solution while providing improved convergence rates, have been proposed in the literature. These techniques fall in the class of the so-called localization methods. In these methods, the update rule is based on the computation of a center of a suitably defined *localization set* $\mathcal{L}_k$. The set $\mathcal{L}_k$ is guaranteed to contain at each step the feasible set $\mathcal{S}$, that is $\mathcal{S} \subseteq \mathcal{L}_k$, and is constructed based on the violation certificate q_k returned by the ϵ-oracle. In particular, the point q_k is used to construct a separating hyperplane $h_k \doteq \{\xi \in \mathbb{R}^{n_\theta} : a_k{}^\top \xi = b_k\}$ having the property that $a_k{}^\top \theta_k \geq b_k$ and $a_k{}^\top \theta \leq b_k$, for all $\theta \in \mathcal{S}$. Specifically, if $\partial_k(\theta_k)$ is a subgradient of $f(\theta, q_k)$ at $\theta = \theta_k$, then a separating hyperplane may be obtained as $a_k = \partial_k(\theta_k)$; $b_k = \partial_k{}^\top(\theta_k)\theta_k - f(\theta_k, q_k)$.

The separating hyperplane h_k indicates that the half-space $\{\theta : a_k{}^\top \theta > b_k\}$ cannot contain a feasible point and can therefore be eliminated (cut) in the subsequent steps of the algorithm. In this case, we know that $\mathcal{S} \subseteq \mathcal{L}_k \cap \mathcal{H}_k$, where

$$\mathcal{H}_k \doteq \{\theta : a_k{}^\top \theta \leq b_k\}$$

and the algorithm constructs an updated localization set $\mathcal{L}_{k+1}$ such that $\mathcal{L}_{k+1} \supseteq \mathcal{L}_k \cap \mathcal{H}_k$. A new query point $\theta_{k+1} \in \mathcal{L}_{k+1}$ is then computed, and the process is repeated. This is summarized in the following scheme.

Algorithm 5 `UpdateRule` (localization methods)

Input: θ_k, q_k
Output: θ_{k+1}

compute the subgradient $\partial_k(\theta)$ of $f(\theta, q_k)$
construct the half-space $\mathcal{H}_k$ based on the subgradient
update the localization set $\mathcal{L}_{k+1} \supseteq \mathcal{L}_k \cap \mathcal{H}_k$
return $\theta_{k+1} = \texttt{Center}(\mathcal{L}_k)$

The convergence of these methods hinges upon the fact that each time a cut is performed, the localization set shrinks by a certain factor. Intuitively, this guarantees that eventually, for sufficiently large k, either we terminate by finding a feasible point, or we declare that the problem is not r-feasible. Different methods descend from different choices of the shape and description of the localization sets. In particular, in the probabilistic ellipsoid algorithm the localization set is an ellipsoid and the candidate solution θ_{k+1} is the ellipsoid center. In the probabilistic cutting plane methods, the localization set is instead a polytope, and the candidate solution is a center of this polytope (usually, the analytic center). These two classes of localization methods are summarized next.

5.3.2.1. Probabilistic ellipsoid algorithm. This is the first randomized localization scheme proposed in the literature; see Kanev, De Schutter, and Verhaegen (2003). The method represents a probabilistic extension of the classical ellipsoid algorithm (see Fig. 3(a)) originally proposed in Shor (1977) and Yudin and Nemirovski (1977). Ellipsoids are described by means of a center θ and a symmetric positive definite shape matrix W

$$\mathcal{E}(\theta, W) \doteq \{x : (x-\theta)^\top W^{-1}(x-\theta) \leq 1\}.$$

We assume that an initial ellipsoid $\mathcal{E}_0 = \mathcal{E}(\theta_0, W_0)$ is given, such that $\mathcal{S} \subseteq \mathcal{E}_0$. The advantage of this method is that the information relative to the localization set at step k is captured by two parameters only, the ellipsoid center θ_k and the shape matrix W_k. This allows us to write the update rule in a simple closed form. If $\partial_k(\theta_k)$ is a subgradient of $f(\theta, q_k)$ at $\theta = \theta_k$, then the new centers and shape matrices can be obtained as follows

$$\theta_{k+1} = \theta_k - \frac{1}{n_\theta + 1} \frac{W_k \partial_k(\theta_k)}{\sqrt{\partial_k{}^\top(\theta_k) W_k \partial_k(\theta_k)}};$$

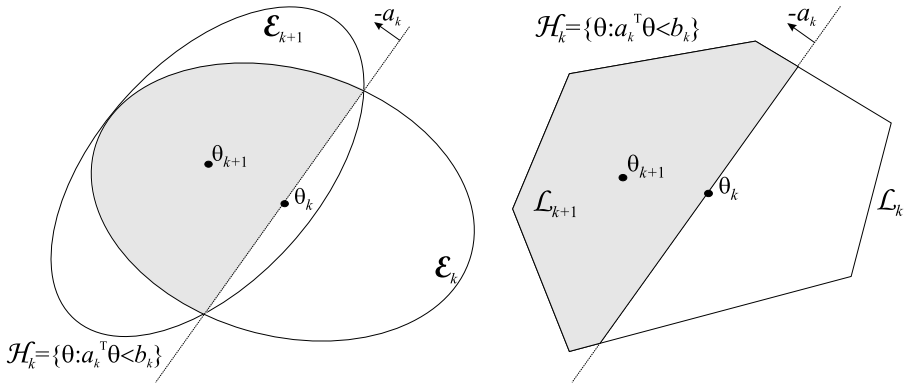
$$W_{k+1} = \frac{{n_\theta}^2}{{n_\theta}^2 - 1}\left(W_k - \frac{2}{n_\theta + 1}\frac{W_k \partial_k(\theta_k)\partial_k{}^\top(\theta_k) W_k}{\partial_k{}^\top(\theta_k) W_k \partial_k{}^\top(\theta_k)}\right).$$

For this algorithm, the results in Kanev et al. (2003) imply that Theorem 2 holds with

$$N_{\text{outer}}(r) = 2n_\theta \log \frac{\text{vol}\,(\mathcal{E}_0)}{\text{vol}\,(\mathcal{B}(r))}$$

where $\mathcal{B}(r)$ is the r-feasibility ball.

5.3.2.2. Cutting plane techniques. The randomized localization scheme based on cutting planes has been proposed in Calafiore and Dabbene (2007). In probabilistic cutting plane methods, the localization set is described by means of a polytope $\mathcal{P}_k$. In the update phase, a new polytope $\mathcal{P}_{k+1}$ is constructed as the intersection of the current localization set $\mathcal{P}_k$ and the cutting plane $\mathcal{H}_k$. Then, a new query point θ_k is computed as a center of the polytope; see Fig. 3(b).



(a) Ellipsoid method. (b) Cutting plane method.

Fig. 3. Update steps of location methods.

Different families of cutting plane methods use different strategies in the outer phase for updating the localization set and constructing the new query point. This is a very delicate phase, since the numerical complexity and convergence of the method critically rely on these choices. Detailed discussions of the various methods fall outside the scope of this paper, so that we only mention some of the most widely known, and we refer the reader to the specific literature (see, e.g., Goffin & Vial, 2002; Mitchell, 2003, for further details). Currently, the most popular family of cutting plane methods is that based on analytic centers, (Goffin & Vial, 2002; Nesterov, 1995). In this case, the query point is computed as the analytic center of the polytope $\mathcal{P}_k$, or of a suitably pruned version of it; see Atkinson and Vaidya (1995). The analytic center is defined as the unique minimizer of the logarithmic barrier function

$$\Phi(\theta) \doteq -\sum_i \log(b_i - a_i{}^\top\theta),$$

where $a_i \in \mathbb{R}^{n_\theta}$, $b_i \in \mathbb{R}$ are the parameters of the hyperplanes defining the polytope $\mathcal{P}_k$. The probabilistic version of the analytic center cutting plane method has been introduced and analyzed in Calafiore and Dabbene (2007), obtaining the explicit bound

$$N_{\text{outer}}(r) = \max\left\{50n_\theta,\, 13.87n_\theta^2,\, 8n_\theta^2\left(\frac{\Omega}{r}\right)^{2.1}\right\}$$

where Ω is the radius of a hypercube known to contain $\mathcal{S}$ and r is the r-feasibility radius.

Remark 5 (*Complexity of Different Randomized Schemes*)**.** We remark that, if we define the quantity $h = \Omega/r$, Ω being the radius of a ball inscribing the set $\mathcal{S}$, then it can be seen that N_{outer} grows as $O(h^2)$, for the gradient rule, and as $O(n_\theta^2 \log(\sqrt{n_\theta}h))$, for the ellipsoid method. For one of the best-known cutting plane methods detailed in Atkinson and Vaidya (1995), the convergence is instead of the order of $O(n_\theta \log^2(h))$. We also remark that all the update rules implicitly satisfy Assumption 2.

5.4. Sequential methods for optimization

The sequential randomized methods discussed in the previous sections are aimed at solving *feasibility* problems (Problem 3). However, these techniques can be adapted also to solve *optimization* problems (Problem 4), by suitably embedding them into bisection or similar techniques. The basic idea is to fix an initial objective level γ and reformulate the optimization problem (Problem 4) into feasibility epigraphic format as

Find θ such that : $c^\top\theta \le \gamma$, $V(\theta) \le \epsilon$,

and iteratively adjust the level γ until it cannot be further reduced while maintaining feasibility. A sequential randomized algorithm for robust optimization which uses stochastic bisection has been introduced in Oishi (2007b), whereas an improvement based on iterative decrease of the objective level is proposed in Fujisaki and Kozawa (2006) and Wada and Fujisaki (2007).

6. Scenario methods for optimization

Scenario techniques provide a simple and theoretically sound way to approximately solve the optimal probabilistic design Problem 4. The idea is to replace the hard optimization problem (14) by the following sampled counterpart (the scenario problem)

$$\text{SP}_N: \quad \theta_{\text{sc}} = \arg\min_{\theta\in\mathbb{R}^{n_\theta}} c^\top\theta \quad \text{subject to :} \tag{19}$$

$$f(\theta, q^{(i)}) \le 0, \quad i = 1, \ldots, N, \tag{20}$$

where $q^{(i)}$, $i = 1, \ldots, N$, are i.i.d. samples extracted according to Prob. Note that (19), (20) is a standard convex optimization problem with a finite number of constraints and therefore it is efficiently solvable in many specific cases of interest in control.

The optimal solution θ_{sc} of this program (the scenario solution) is a random variable that depends on the random extractions $q^{(1)}, \ldots, q^{(N)}$. We then ask the following question: what is the probability of violation of θ_{sc}? That is, what can we say a priori on $V(\theta_{\text{sc}})$? To answer this question avoiding unduly technical details, we shall work under the following assumption.

Assumption 3. For all $N \ge n_\theta$, problem SP_N is feasible and attains a unique optimal solution θ_{sc}.

Note that non-uniqueness of the optimal solution can be circumvented by imposing additional "tie-break" rules in the problem; see, e.g., Appendix A of Calafiore and Campi (2006). The theory of sampled convex programs has been laid in the two papers (Calafiore & Campi, 2005, 2006). In particular, from a result in equation (11) of Calafiore and Campi (2006) we obtain that, for any given $\epsilon \in (0, 1)$, the probability of the event $V(\theta_{\text{sc}}) > \epsilon$ is bounded from above as follows

$$\text{Prob}^N\{V(\theta_{\text{sc}}) > \epsilon\} \le \tilde{\delta}(\epsilon) \tag{21}$$

where

$$\tilde{\delta}(\epsilon) \doteq \begin{cases} 1, & \text{if } \epsilon \le 1 - \binom{N}{n_\theta}^{-\frac{1}{N-n_\theta}} \\ \binom{N}{n_\theta}(1-\epsilon)^{N-n_\theta}, & \text{otherwise.} \end{cases}$$

This means that the a priori probability of producing a scenario solution with a violation larger than ϵ can be made arbitrarily small by choosing a sufficiently high number N of scenarios. More precisely, let $\delta \in (0, 1)$ be a given (small) probability level. Then, a key result in Calafiore and Campi (2006, Theorem 1 and Corollary 1) states that if

$$N \geq N(\epsilon, \delta) = \left\lceil \frac{2}{\epsilon} \log \frac{1}{\delta} + 2n_\theta + \frac{2n_\theta}{\epsilon} \log \frac{2}{\epsilon} \right\rceil \tag{22}$$

samples are taken in the scenario problem (19), (20), then it holds that

$$\text{Prob}^N\{V(\theta_{\text{sc}}) \leq \epsilon\} \geq 1 - \delta. \tag{23}$$

In other words, with high probability $1 - \delta$, the scenario solution is feasible for all the constraints in (20), except possibly for those in a set having probability measure smaller than ϵ. A fundamental point is that $N(\epsilon, \delta)$ is computed a priori, before any constraint is extracted, and that this bound holds in full generality for any uncertain convex program, and any probability distribution on the uncertainties. Since (22) scales essentially as $O\left(\epsilon^{-1} \log \delta^{-1}\right)$, in practice the δ level can be fixed to a very small value (say, 10^{-9}), without increasing too much the required number of samples.

6.1. Tight bounds on the violation tail

In the recent work (Campi & Garatti, 2008) an improved bound on the tail violation probability (21) has been derived under the hypothesis in Assumption 3 that any realization of the constraints, for all $N \geq n_\theta$, attains an optimal solution. That is, the key result in Campi and Garatti (2008) states that (21) holds for

$$\tilde{\delta}(\epsilon) \doteq \sum_{i=0}^{n_\theta - 1} \binom{N}{i} \epsilon^i (1 - \epsilon)^{N-i}.$$

Moreover, (21) holds with equality for the class of fully supported problems,[1] hence this bound is tight. This result has been extended to the possibly unfeasible case in Calafiore (2010a). Interestingly, the above probability corresponds to the lower tail of a binomial distribution and can be given the following interpretation in terms of coin tossing (Bernoulli trials): suppose we toss a coin which heads with probability ϵ and tails with probability $1 - \epsilon$. Then, $\tilde{\delta}(\epsilon)$ coincides with the probability of obtaining less than n_θ heads out of N of such tosses. Thus, if x denotes the number of heads obtained in N tosses, we have that $\tilde{\delta}(\epsilon) = \text{Prob}\{x \leq n_\theta - 1\}$. In Calafiore (2009b), the Chernoff bound on the lower binomial tail (see Chernoff, 1952) is used to "invert" the bound $\tilde{\delta}(\epsilon) \leq \delta$ and make it explicit with respect to N, as detailed in the next proposition.

Proposition 1. *Let Assumption 3 be satisfied. Given $\epsilon \in (0, 1)$ and $\delta \in (0, 1)$, let θ_{sc} be a scenario solution with*

$$N \geq \frac{2}{\epsilon}\left(\log \frac{1}{\delta} + n_\theta\right). \tag{24}$$

Then,

$$\text{Prob}^N\{V(\theta_{\text{sc}}) > \epsilon\} \leq \delta.$$

More refined lower bounds can be found for N, but (24) has the advantage of being simple and showing that the sample complexity has log dependence in δ^{-1} and linear dependence in ϵ^{-1}. Moreover, (24) clearly improves upon (22). An explicit bound that further improves upon (24) and other existing bounds is given in Alamo, Tempo, and Luque (2010).

6.2. Assessments with a single level of probability

Note that results in the previous two sections involve a double level of probability (probability of a probability): Eq. (23) states that the probability of violation $V(\theta_{\text{sc}})$ is less than or equal to ϵ, with probability at least $1 - \delta$. The latter probability is measured in the space $\mathbb{Q}^N$ of the multi-extractions $q^{(1)}, \ldots, q^{(N)}$, whereas $V(\theta_{\text{sc}})$ is measured in $\mathbb{Q}$. We can actually provide an alternative statement with a single probability in the product space $\mathbb{Q}^N \times \mathbb{Q}$. That is, we evaluate the probability

$$P_B \doteq \text{Prob}^{N+1}\{(q^{(1)}, \ldots, q^{(N)}, q) \in \mathbb{Q}^N \times \mathbb{Q} : f(\theta_{\text{sc}}, q) > 0\}. \tag{25}$$

The following result is from Campi and Calafiore (2009).

Proposition 2. *Let Assumption 3 be satisfied. Consider the scenario solution θ_{sc}, with $N \geq n_\theta$. The a priori probability (25) of the event $f(\theta_{\text{sc}}, q) > 0$ is*

$$P_B = \text{E}^N\{V(\theta_{\text{sc}})\} \leq \frac{n_\theta}{N + 1}, \tag{26}$$

where E^N denotes the expectation over Prob^N.

Formula (26) expresses the probability that, if we solve a scenario optimization problem on the basis of N samples, the obtained solution turns out to be unfeasible for another sample extracted according to the same probability. It is also not difficult to prove by counting arguments that expression (26) actually holds with equality for the class of fully supported problems; see Calafiore (2009b).

6.3. Optimization with discarded constraints

A common feature of the scenario method, as well as of the sequential design techniques discussed in Section 5, is to produce solutions that tend to satisfy the constraints with probability close to one. While this is exactly the feature we are looking for when designing for robustness, it may constitute a limit in cases where larger risk of constraint violation might be tolerated at the advantage of increased performance. In other words, we may be wishing to give up some reliability in exchange of a good improvement of the optimal objective value.

To address this issue, we consider a modification of the scenario problem (19), (20) in which we allow some of the constraints to be discarded a posteriori (that is, after seeing them). More precisely, let $k < N - n_\theta$ be an a priori fixed number of constraints to be removed, and let R_k be some generic deterministic *rule* for discarding k constraints out of the observed N

$$R_k(F_N) = \{j_1, \ldots, j_{N-k}\} \subset \{1, \ldots, N\},$$

where $F_N = \{f(\theta, q^{(i)}) \leq 0,\ i = 1, \ldots, N\}$ represents the randomly extracted batch of constraints. A scenario problem with k discarded constraints is then defined as follows

$$\text{SP}_{N,k}: \quad \theta_{\text{sc},k} = \arg\min_{\theta \in \mathbb{R}^{n_\theta}} c^\top \theta \quad \text{subject to :}$$
$$f(\theta, q^{(j_i)}) \leq 0, \quad i = 1, \ldots, N - k,$$

where $\{j_1, \ldots, j_{N-k}\} = R_k(F_N)$. A result on the violation probability for the solution $\theta_{\text{sc},k}$ is given in the following proposition.

Proposition 3. *Assume that problem $\text{SP}_{N,k}$ is feasible and attains a unique optimal solution $\theta_{\text{sc},k}$. Then, for any $k < N - n_\theta$ and $\epsilon \in (0, 1)$*

[1] Fully supported problems are problems of the form SP_N for which the number of support constraints is exactly n_θ; see Calafiore and Campi (2005) for a definition of support constraints.

it holds that

$$\text{Prob}^N\{V(\theta_{\text{sc},k}) > \epsilon\} \leq \tilde{\delta}_k(\epsilon)$$

where

$$\tilde{\delta}_k(\epsilon) \doteq \binom{N}{n_\theta} \sum_{i=0}^{k} \binom{N-n_\theta}{i} \epsilon^i (1-\epsilon)^{N-n_\theta-i}.$$

This result appeared for the first time in Campi, Calafiore, and Garatti (2005), in the context of identification of interval predictor models. A full proof is reported in Campi, Calafiore, and Garatti (2009) (see the proof of Theorem 3 in Campi et al., 2009). An improved result still in the context of interval predictors is found in Calafiore (2010b). It is worthy to remark that the result in Proposition 3 holds irrespective of the rule used for discarding the constraints. Therefore, we can apply Proposition 3 to the case when the k constraints are discarded in an "optimal" way, that is when the rule is: "remove those k constraints whose removal provides the largest reduction in objective value". Or, since determining such an optimal set of discarded constraints is of combinatorial complexity, we may remove them via some sub-optimal strategy (such as sequentially removing the constraint that gives the largest objective improvement), and still apply Proposition 3. The point is that no matter how the constraints are removed, the resulting solution is guaranteed to have violation $V(\theta_{\text{sc},k}) \leq \epsilon$, with probability higher than $1 - \tilde{\delta}_k(\epsilon)$. This approach thus gives the designer the possibility of improving the objective by a posteriori removing some of the constraints (those that are more adverse to the optimization objective), while maintaining control on the violation probability through N and k.

Very recently, a significant improvement upon the result in Proposition 3 has been obtained independently in Calafiore (2010a) and in Campi and Garatti (in press). In particular, in Campi and Garatti (in press) the following bound is derived under the hypothesis of existence of a solution in any possible problem instance

$$\text{Prob}^N\{V(\theta_{\text{sc},k}) > \epsilon\} \leq \binom{k+n_\theta-1}{k} \sum_{i=0}^{k+n_\theta-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i},$$

whereas in Calafiore (2010a) the following result is proved under no hypotheses on the existence of the solution

$$\text{Prob}^N\{\{V(\theta_{\text{sc},k}) > \epsilon\} \cap \mathbb{Q}^{N*}\} \leq \binom{k+n_\theta}{k} \sum_{i=0}^{k+n_\theta} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i},$$

where $\mathbb{Q}^{N*}$ denotes the set of multi-extractions where the problem admits a solution.

7. Statistical learning theory for nonconvex control design

Nonconvex design problems may be tackled using a probabilistic approach based on statistical learning theory, see Vapnik (1998) and Vidyasagar (2002), which has the objective to derive *uniform convergence laws*. From the control point of view, this line of research was initiated in Vidyasagar (1998), Vidyasagar (2001) and Vidyasagar and Blondel (2001); see also subsequent developments in Alamo, Tempo, and Camacho (2009), Koltchinskii, Abdallah, Ariola, and Dorato (2001) and Koltchinskii, Abdallah, Ariola, Dorato, and Panchenko (2000). The main utility of statistical learning is to derive convergence results and compute the sample complexity which holds uniformly for all controller parameters θ. In turn, this leads to a powerful methodology for control synthesis which is not based upon a convexity assumption on the controller parameters. We also remark that statistical learning theory may be seen as a major extension to control design of the classical Monte Carlo approach, which is limited to a posteriori performance analysis, once the controller parameters have been selected.

This section is divided into two parts: in the first part, we discuss general statistical learning theory results related to the so-called UCEM property (Uniform Convergence of Empirical Means) and the estimation of the so-called probability of failure using the empirical mean. In the second part, we present a specific result which provides the sample complexity for nonconvex optimization problems subject to binary nonlinear constraints. This result may be considered as a nonconvex generalization of the scenario results given in Section 6. The obtained sample complexity bounds, however, are significantly larger than those derived in the convex case.

In this section we consider constraint functions of binary form, that is

$$f(\theta, q) : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \{0, 1\},$$

understanding that $f(\theta, q) = 0$ means that a performance specification is satisfied for the given design parameter θ and uncertainty q, whereas $f(\theta, q) = 1$ means that the specification is not satisfied. Note that, since f is binary valued, we describe these two situations maintaining the notation introduced in the previous sections, that is $f(\theta, q) \leq 0$ for satisfied constraints, and $f(\theta, q) > 0$ for violated constraints. Similarly, the probability of violation $V(\theta)$ (Definition 1) writes

$$\begin{aligned} V(\theta) &= \text{Prob}\{q \in \mathbb{Q} : f(\theta, q) > 0\} \\ &= \text{Prob}\{q \in \mathbb{Q} : f(\theta, q) = 1\} = \text{E}\{f(\theta, q)\}. \end{aligned} \tag{27}$$

Given $\theta \in \mathbb{R}^{n_\theta}$, it is generally difficult to determine the exact value of $V(\theta)$, since this requires the solution of a multiple integral. However, we can approximate its value using the concept of empirical mean. For given $\theta \in \mathbb{R}^{n_\theta}$, the empirical mean of $f(\theta, q)$ with respect to the multisample $q^{(1)}, \ldots, q^{(N)}$, is defined as

$$\hat{V}_N(\theta) \doteq \frac{1}{N} \sum_{i=1}^{N} f(\theta, q^{(i)}).$$

Clearly, the empirical mean $\hat{V}_N(\theta)$ is a random variable taking values in the interval $[0, 1]$. We refer to this quantity as the *empirical violation*.

7.1. A uniform bound on the probability of two-sided failure

We now address the problem of deriving the sample complexity to guarantee that the empirical violation is within a pre-specified accuracy $\epsilon \in (0, 1)$ from the actual violation probability, with confidence $1 - \delta, \delta \in (0, 1)$. If θ is fixed, then the Hoeffding inequality (7) characterizes how the empirical mean approximates the exact value of $V(\theta)$

$$\text{Prob}^N\left\{|V(\theta) - \hat{V}_N(\theta)| \geq \epsilon\right\} \leq 2\text{e}^{-2N\epsilon^2}.$$

However, since we need to optimize over θ, we need to find a bound which holds uniformly over all possible θ. Motivated by this observation, we now introduce the definition of uniform probability of failure.

Definition 3 (*Uniform Probability of Failure*)**.** Given $\epsilon \in (0, 1)$, N, and $f : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \{0, 1\}$, the uniform probability of failure, denoted by $p_f(N, \epsilon)$ is defined as

$$p_f(N, \epsilon) \doteq \text{Prob}^N\left\{\sup_{\theta \in \mathbb{R}^{n_\theta}} |V(\theta) - \hat{V}_N(\theta)| > \epsilon\right\}.$$

If $p_f(N,\epsilon) \to 0$ as $N \to \infty$, for each $\epsilon > 0$, then we say that function $f : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \{0,1\}$ enjoys the uniform convergence of empirical means (UCEM) property; see Vapnik and Chervonenkis (1981) and Vidyasagar (2002).

A key concept in statistical learning theory is the so-called VC-dimension, which characterizes how difficult it is to "uniformly learn" (that is, for all $\theta \in \mathbb{R}^{n_\theta}$) the function $f(\theta, q)$ from a finite number of samples of q. In other words, the VC-dimension characterizes the problem difficulty. To introduce a formal definition of VC-dimension, we need to present some preliminary definitions.

Formally, let $\mathcal{F}$ denote the family of functions $\{f(\theta,\cdot) : \theta \in \mathbb{R}^{n_\theta}\}$, where $f : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \{0,1\}$. Then, given the multisample $\{q^{(1)}, \ldots, q^{(N)}\} \in \mathbb{Q}^N$, the binary vector $[f(\theta, q^{(1)}) \ldots f(\theta, q^{(N)})]^\top \in \{0,1\}^N$ can attain at most 2^N distinct values, as θ varies in $\mathbb{R}^{n_\theta}$. The maximum number of distinct binary vectors that can be obtained grows with the number of samples N. The next definition introduces in a formal way the notion of growth function (also known as shatter coefficient); see Vapnik (1998), Vapnik and Chervonenkis (1971) and Vidyasagar (2002).

Definition 4 (*Growth Function*)**.** Given the function $f : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \{0,1\}$ and the multisample

$$q^{(1\ldots N)} \doteq \{q^{(1)}, \ldots, q^{(N)}\} \in \mathbb{Q}^N,$$

let $\phi_f(q^{(1\ldots N)})$ denote the number of distinct binary vectors

$$[f(\theta, q^{(1)}) \ldots f(\theta, q^{(N)})]^\top \in \{0,1\}^N$$

that can be obtained as θ varies over $\mathbb{R}^{n_\theta}$. Then, the growth function $\pi_f(N)$ is defined as

$$\pi_f(N) = \sup_{q^{(1\ldots N)} \in \mathbb{Q}^N} \phi_f(q^{(1\ldots N)}).$$

In the celebrated work of Vapnik and Chervonenkis (1971), it has been demonstrated that the growth function can be used to bound the uniform probability of failure. This is now formally stated.

Theorem 3. *Given the function $f : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \{0,1\}$ and $\epsilon \in (0,1)$,*

$$p_f(N,\epsilon) \le 4\pi_f(2N)\mathrm{e}^{\frac{-N\epsilon^2}{8}}.$$

We are now ready to introduce the notion of Vapnik–Chervonenkis dimension (VC-dimension); see Vapnik (1998) and Vidyasagar (2002).

Definition 5 (*VC-Dimension*)**.** Given the function $f : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \{0,1\}$, the VC-dimension, denoted as VC_f, is the largest integer N for which the equality $\pi_f(N) = 2^N$ is satisfied.

We now present a lemma which provides a bound on the VC-dimension for a certain class of binary functions. More precisely, we study the case when the binary function $f(\theta, q)$ can be written as a Boolean polynomial expression on the decision variable θ, as now formally stated.

Definition 6 ((α, m)*-Boolean Function*)**.** The function $f : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \{0,1\}$ is a (α, m)*-Boolean* if, for fixed q, it can be written as a Boolean expression consisting of Boolean operators involving m polynomials $\beta_1(\theta), \ldots, \beta_m(\theta)$ in the variables θ_i, $i = 1, \ldots, n_\theta$, and the degree with respect to θ_i of all these polynomials is no larger than $\alpha > 0$.

The following lemma holds.

Lemma 1. *Suppose that the function $f : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \{0,1\}$ is (α, m)-Boolean. Then, the VC-dimension is bounded as*

$$\mathrm{VC}_f \le 2n_\theta \log_2(4\mathrm{e}\alpha m).$$

This result, which is stated in Vidyasagar (2002), improves a similar bound given in Karpinski and Macintyre (1997). More generally, the VC-dimension establishes the "richness" of a given family $\mathcal{F}$ of functions, and it can be used to determine a bound on the growth function by means of the so-called Sauer Lemma; see Sauer (1972) and Vidyasagar (2002).

Lemma 2. *Let the function $f : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \{0,1\}$ be such that $\mathrm{VC}_f \le d < \infty$. Then, for every $N \ge d$ we have*

$$\pi_f(N) \le \left(\frac{\mathrm{e}N}{d}\right)^d.$$

One of the direct consequences of this result is that, under the assumption of finite VC-dimension, $\pi_f(N)$ is bounded by a polynomial function of N. In turn, this implies that

$$\lim_{N\to\infty} 4\pi_f(2N)\mathrm{e}^{\frac{-N\epsilon^2}{8}} = 0$$

for any $\epsilon > 0$. Hence, we conclude that, if the VC-dimension is bounded, the uniform probability of failure converges to zero when N tends to infinity. Note that there are known cases where the VC-dimension is not bounded, see e.g. Vapnik (1998), and hence the problem is not "uniformly" learnable.

Using Theorem 3 and Lemma 2, the following result is obtained in Vidyasagar (2002).

Theorem 4. *Let $f : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \{0,1\}$ be such that $\mathrm{VC}_f \le d < \infty$. Suppose that $\epsilon \in (0,1)$ and $\delta \in (0,1)$ are given and $N \ge d$. Then,*

$$p_f(N,\epsilon) \le 4\left(\frac{2\mathrm{e}N}{d}\right)^d \mathrm{e}^{\frac{-N\epsilon^2}{8}}.$$

Moreover, $p_f(N,\epsilon) \le \delta$ provided that

$$N \ge \max\left\{\frac{16}{\epsilon^2}\ln\frac{4}{\delta}, \frac{32d}{\epsilon^2}\ln\frac{32\mathrm{e}}{\epsilon^2}\right\}.$$

Theorem 4 is well known in the control community. However, as discussed in Vidyasagar (2002), there exist other results in the literature that allow one to reduce by a factor close to 8 the explicit bound on the number of samples. Basically, these results differ from Theorem 3 in the exponent $-N\epsilon^2/8$ which is replaced by less conservative ones. For example, the exponent $-N\epsilon^2$ can be found in Parrondo and van den Broeck (1993) and Vapnik (1998).

Next, we present a recent sample complexity result stated in Alamo et al. (2009), which slightly improves upon the bound given in Theorem 4. It is interesting to observe that the structure of this bounds is very different than the previous one.

Theorem 5. *Let $f : \mathbb{R}^{n_\theta} \times \mathbb{Q} \to \{0,1\}$ be such that $\mathrm{VC}_f \le d < \infty$. Suppose that $\epsilon \in (0,1)$ and $\delta \in (0,1)$ are given. Then, the uniform probability of failure $p_f(N,\epsilon)$ is smaller than δ, if*

$$N \ge \frac{1.2}{\epsilon^2}\left(\ln\frac{4\mathrm{e}^{2\epsilon}}{\delta} + d\ln\frac{12}{\epsilon^2}\right).$$

We note that the sample complexity given in these results grows with $\frac{1}{\epsilon^2}\ln\frac{1}{\epsilon^2}$. This dependence with respect to ϵ makes the bound of practical interest only for relatively large values of the accuracy parameter $\epsilon \in (0,1)$. A major improvement, leading to a sample complexity bound of order $\frac{1}{\epsilon}\ln\frac{1}{\epsilon}$, can be obtained if, instead of the probability of two-sided failure, one studies the so-called *probability of one-sided constrained failure*; see Alamo et al. (2009).

7.2. Nonconvex scenario optimization with binary functions

We consider in this section a nonconvex version of Problem 4, in which the objective function $J : \mathbb{R}^{n_\theta} \to \mathbb{R}$ is nonlinear (and possibly nonconvex) and the violation function is given in (27)

$$\min_{\theta\in\mathbb{R}^{n_\theta}} J(\theta) \quad \text{subject to } V(\theta) \le \epsilon. \tag{28}$$

More precisely, we study the case when the binary function $f(\theta, q)$ is (α, m)-Boolean (see Definition 6). Similar to the scenario approach in Section 6, the idea is to replace the hard optimization problem (28) by the following sampled counterpart

$$\begin{aligned} \text{NCSP}_N : \quad & \theta_{\text{ncsc}} = \arg\min_{\theta\in\mathbb{R}^{n_\theta}} J(\theta) \quad \text{subject to :} \\ & f(\theta, q^{(i)}) = 0, \quad i = 1, \ldots, N, \end{aligned} \tag{29}$$

where $q^{(i)}$, $i = 1, \ldots, N$, are i.i.d. samples extracted according to Prob.

Note that (29) requires in general the solution of a nonconvex optimization problem, which usually leads only to a local minimum. However, the following theorem, stated in Alamo et al. (2009), guarantees that any local minimum is ϵ-feasible, i.e. we can suitably bound the probability of violation with accuracy ϵ and confidence δ.

Theorem 6 (*Nonconvex Learning-Based Design*). *Let $f(\theta, q)$ be (α, m)-Boolean. Given $\epsilon \in (0, 0.14)$ and $\delta \in (0, 1)$, if*

$$N \ge N_{\text{ncsc}}(\epsilon, \delta, n_\theta) \doteq \left\lceil \frac{4.1}{\epsilon}\left(\ln\frac{21.64}{\delta} + 4.39 n_\theta \log_2\left(\frac{8\mathrm{e}\alpha m}{\epsilon}\right)\right)\right\rceil,$$

where e *is the Euler number, then the probability that $V(\theta_{\text{ncsc}}) > \epsilon$ is at most δ.*

8. Systems and control applications

In this section, we briefly summarize some systems and control applications of probabilistic and randomized methods.

- *Aerospace control*. Randomized strategies for the design of control algorithms in the field of aeronautics and aerospace were initiated by Stengel; see, e.g., Stengel (1980). More recently, Monte Carlo algorithms for controller design have been developed in Lorefice, Pralio, and Tempo (2009) and Wang and Stengel (2005). Similar techniques for detecting conflict resolution in air traffic control have been studied in Lecchini-Visintini, Glover, Lygeros, and Maciejowski (2006). In Lu and Wu (2006) a probabilistic approach is proposed for the design of a linear parameter-varying control of an F-16 aircraft.
- *Flexible and truss structures*. Probabilistic robustness of flexible structures consisting of a mass–spring–damper model affected by random bounded uncertainty with force actuators and position sensors are studied in Calafiore et al. (2000) where comparisons with standard robustness techniques are made. In the field of truss topology optimization, a scenario-based approach has been proposed in Calafiore and Dabbene (2008).
- *Switched and hybrid systems*. Randomized algorithms for the synthesis of multimodal systems with state-dependent switching rules have been studied in Ishii, Başar, and Tempo (2005). Convergence properties (with probability one) of nonconvex sequential algorithms, based on randomized methods and branch-and-bound techniques, are analyzed in this paper. Furthermore, randomized algorithms (of Las Vegas type) for the construction of switching rules to select the most stable system in a certain class are provided in Ishii and Tempo (2009). For stochastic hybrid systems with control inputs probabilistic reachability over a finite horizon is investigated in Abate, Prandini, Lygeros, and Sastry (2008). This problem requires us to determine a set of initial conditions which guarantee that the system evolves within a desired safe region of the state space with a given probability. Randomized algorithms for controllability analysis for discrete-time piecewise affine systems are discussed in Azuma and Imura (2007).
- *Network control*. Congestion control of high-speed communication networks by means of probabilistic methods has been addressed in Alpcan, Başar, and Tempo (2005). Feedback controllers are derived for improving the Quality of Service (QoS) for Available Bit Rate (ABR) in Asynchronous Transmission Mode (ATM) networks utilizing Monte Carlo and Quasi-Monte Carlo techniques. Statistical learning control with application to ATM networks are studied in Abdallah, Ariola, Dorato, and Panchenko (2001).
- *Automotive and driver assistance systems*. A randomization-based approach for validation of advanced driver assistance systems is studied in Gietelink, De Schutter, and Verhaegen (2005). Since the tests required are costly, efficient methodologies to conduct hardware-in-the-loop experiments in a laboratory environment with full-scale intelligent vehicles are developed. Convergence properties of the algorithms regarding the required sample complexity and the sensitivity to model uncertainty are established.
- *Model predictive control (MPC), fault detection and isolation (FDI)*. Sequential methods (ellipsoid based) are derived in Kanev and Verhaegen (2006) to design robustly stable finite-horizon MPC schemes for linear uncertain systems subject to general classes of uncertainty. In Ma, Sznaier, and Lagoa (2007) a risk-adjusted approach based on randomization is proposed for robust simultaneous fault detection and isolation of MIMO systems. Furthermore, a computationally efficient (polynomial time) algorithm for model (in)validation in the presence of structured uncertainty and robust performance over finite horizons is proposed in Sznaier et al. (2005).
- *Circuits and embedded systems*. The performance of electric circuits subject to uncertain components introduced by the manufacturing process has been addressed in several papers. The objective is to evaluate the probability that a given "system property" holds providing "hard" (deterministic) bounds (Lagoa, Dabbene, & Tempo, 2008). In Alippi (2002a,b) randomized techniques are applied to estimate the performance degradation of digital architectures and embedded systems subject to various sources of perturbations. In Kettani and Barmish (2008) the choice of the worst-case distribution for various RCL circuits is addressed.
- *Multiagent systems and PageRank computation*. In Pallottino, Scordio, Frazzoli, and Bicchi (2007) a decentralized approach for studying collision avoidance of a large number of autonomous vehicles is considered. The proposed method is based on a classical Monte Carlo analysis for probabilistic assessment on the satisfaction of given design specifications. A related problem deals with ranking web pages for facilitating the search of specific engines such as Google or Yahoo!. Decentralized randomized algorithms for the computation of the eigenvector (called PageRank) corresponding to the largest eigenvalue of a positive stochastic matrix representing the link structure of the web are proposed in Ishii and Tempo (2010), where the connections between PageRank and multiagent consensus are also outlined.

9. Conclusions

In this paper, an overview of the research performed in about a decade in the area of probabilistic methods and randomized algorithms for control of uncertain systems is given. On the one hand, theoretical improvements are still expected in this field,

especially in the areas of efficient robust design and optimization, and distributed computing. On the other hand, probabilistic and randomized techniques have reached a good level of maturity, and appear to be a sound methodology for handing many and diverse engineering design problems. We believe that in the near future an increasing number of applications will be successfully attacked with randomization and simulation-based methods.

References

Abate, A., Prandini, M., Lygeros, J., & Sastry, S. (2008). Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, *44*(11), 2724–2734.

Abdallah, C., Ariola, M., Dorato, P., & Panchenko, D. (2001). Statistical-learning control of multiple-delay systems with applications to ATM networks. *Kybernetika*, *120*, 355–365.

Alamo, T., Tempo, R., & Camacho, E. F. (2009). A randomized strategy for probabilistic solutions of uncertain feasibility and optimization problems. *IEEE Transactions on Automatic Control*, *54*, 2545–2559.

Alamo, T., Tempo, R., & Luque, A. (2010). On the sample complexity of probabilistic analysis and design methods. In J. C. Willems, S. Hara, Y. Ohta, & H. Fujioka (Eds.), *Perspectives in mathematical system theory, control and signal processing* (pp. 39–50). London: Springer-Verlag.

Alippi, C. (2002a). A probably approximately correct framework to estimate performance degradation in embedded systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, *21*, 749–762.

Alippi, C. (2002b). Randomized algorithms: a system-level, poly-time analysis. *IEEE Transactions on Computers*, *51*, 740–749.

Alpcan, T., Başar, T., & Tempo, R. (2005). Randomized algorithms for stability and robustness analysis of high speed communication networks. *IEEE Transactions on Neural Networks*, *16*, 1229–1241.

Atkinson, D. S., & Vaidya, P. M. (1995). A cutting plane algorithm for convex programming that uses analytic centers. *Mathematical Programming, Series B*, *69*, 1–43.

Au, S. K., & Beck, J. L. (2001). Estimation of small failure probability in high dimensions simulation. *Probabilistic Engineering Mechanics*, *16*, 263–277.

Azuma, S.-I., & Imura, J.-I. (2007). Polynomial-time probabilistic controllability analysis of discrete-time piecewise affine systems. *IEEE Transactions on Automatic Control*, *52*(11), 470–482.

Bai, E.-W., Tempo, R., & Fu, M. (1998). Worst-case properties of the uniform distribution and randomized algorithms for robustness analysis. *Mathematics of Control, Signals, and Systems*, *11*, 183–196.

Barmish, B. R. (1994). *New tools for robustness of linear systems*. New York: MacMillan.

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

Barmish, B. R., & Shcherbakov, P. S. (2002). On avoiding vertexization of robustness problems: the approximate feasibility concept. *IEEE Transactions on Automatic Control*, *47*, 819–824.

Calafiore, G. C. (2009a). Distributed randomized algorithms for probabilistic performance analysis. *Systems & Control Letters*, *58*, 202–212.

Calafiore, G. C. (2009b). On the expected probability of constraint violation in sampled convex programs. *Journal of Optimization Theory and Applications*, *143*, 405–412.

Calafiore, G. C. (2010a). Random convex programs. *SIAM Journal on Optimization*, *20*(6), 3427–3464.

Calafiore, G. C. (2010b). Learning noisy functions via interval models. *Systems and Control Letters*, *59*(7), 404–413.

Calafiore, G. C., & Campi, M. C. (2005). Uncertain convex programs: randomized solutions and confidence levels. *Mathematical Programming*, *102*(1), 25–46.

Calafiore, G. C., & Campi, M. C. (2006). The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, *51*(5), 742–753.

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

Calafiore, G. C., & Dabbene, F. (2004). Control design with hard/soft performance specifications: a Q-parameter randomisation approach. *International Journal of Control*, *77*, 461–471.

Calafiore, G. C., & Dabbene, F. (Eds.). (2006). *Probabilistic and randomized methods for design under uncertainty*. London: Springer-Verlag.

Calafiore, G. C., & Dabbene, F. (2007). A probabilistic analytic center cutting plane method for feasibility of uncertain LMIs. *Automatica*, *43*(12), 2022–2033.

Calafiore, G. C., & Dabbene, F. (2008). Optimization under uncertainty with applications to design of truss structures. *Structural and Multidisciplinary Optimization*, *35*, 189–200.

Calafiore, G. C., Dabbene, F., & Tempo, R. (2000). Randomized algorithms for probabilistic robustness with real and complex structured uncertainty. *IEEE Transactions on Automatic Control*, *45*, 2218–2235.

Calafiore, G. C., Dabbene, F., & Tempo, R. (2007). A survey of randomized algorithms for control synthesis and performance verification. *Journal of Complexity*, *23*(3), 301–316.

Calafiore, G. C., & Polyak, B. T. (2001). Stochastic algorithms for exact and approximate feasibility of robust LMIs. *IEEE Transactions on Automatic Control*, *46*, 1755–1759.

Camacho, E. F., & Bordons, C. (2003). *Model predictive control*. London: Springer-Verlag.

Campi, M. C., & Calafiore, G. C. (2009). Notes on the scenario design approach. *IEEE Transactions on Automatic Control*, *54*(2), 382–385.

Campi, M. C., Calafiore, G. C., & Garatti, S. (2005). New results on the identification of interval predictor models. In *Proceedings of the IFAC world congress. Prague*.

Campi, M. C., Calafiore, G. C., & Garatti, S. (2009). Interval predictor models: identification and reliability. *Automatica*, *45*(2), 382–392.

Campi, M. C., & Garatti, S. (2008). The exact feasibility of randomized solutions of robust convex programs. *SIAM Journal on Optimization*, *19*(3), 1211–1230.

Campi, M. C., & Garatti, S. (2011). A sampling-and-discarding approach to chance-constrained optimization: feasibility and optimality. Journal of Optimization Theory and Applications. Vol. 148. No. 2 (in press) (preliminary version available on Optimization Online, 2008).

Chen, X., & Zhou, K. (1998). Order statistics and probabilistic robust control. *Systems & Control Letters*, *35*.

Chen, X., Zhou, K., & Aravena, J. L. (2004). Fast construction of robustness degradation function. *SIAM Journal on Control and Optimization*, *42*, 1960–1971.

Chernoff, H. (1952). A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *Annals of Mathematical Statistics*, *23*, 493–507.

Ching, J., Au, S. K., & Beck, J. L. (2005). Reliability estimation for dynamical systems subject to stochastic excitation using subset simulation with splitting. *Computer Methods in Applied Mechanics and Engineering*, *194*, 1557–1579.

Dabbene, F., Gay, P., & Polyak, B. T. (2003). Recursive algorithms for inner ellipsoidal approximation of convex polytopes. *Automatica*, *39*(10), 1773–1781.

Dabbene, F., Shcherbakov, P. S., & Polyak, B. T. (2010). A randomized cutting plane method with probabilistic geometric convergence. *SIAM Journal on Optimization*, *20*, 3185–3207.

Dabbene, F., & Tempo, R. (2010). Probabilistic and randomized tools for control design. In W. S. Levine (Ed.), *The control handbook* (second ed.) London: Taylor and Francis.

Devroye, L. P. (1986). *Non-uniform random variate generation*. New York: Springer-Verlag.

Doyle, J. C., Francis, B. A., & Tannenbaum, A. R. (1992). *Feedback control theory*. New York: Macmillan Publishing Co..

Fishman, G. S. (1996). *Monte Carlo: concepts, algorithms, and applications*. New York: Springer-Verlag.

Fujisaki, Y., Dabbene, F., & Tempo, R. (2003). *Probabilistic robust design of LPV control systems Automatica*, *39*, 1323–1337.

Fujisaki, Y., & Kozawa, Y. (2006). Probabilistic robust controller design: probable near minmax value and randomized algorithms. In *Probabilistic and randomized methods for design under uncertainty* (pp. 317–329). London: Springer-Verlag.

Fujisaki, Y., & Oishi, Y. (2007). Guaranteed cost regulator design: a probabilistic solution and a randomized algorithm. *Automatica*, *43*, 317–324.

Fujisaki, Y., Oishi, Y., & Tempo, R. (2008). Mixed deterministic/randomized methods for fixed order controller design. *IEEE Transactions on Automatic Control*, *53*(9), 2033–2047.

El Ghaoui, L., & Lebret, H. (1998). Robust solutions to uncertain semidefinite programs. *SIAM Journal on Optimization*, *9*(1), 33–52.

Gietelink, O. J., De Schutter, B., & Verhaegen, M. (2005). Probabilistic approach for validation of advanced driver assistance systems. *Transportation Research Record*, *1910*, 20–28.

Goffin, J.-L., & Vial, J.-P. (2002). Convex non-differentiable optimization: a survey focused on the analytic center cutting plane method. *Optimization Methods & Software*, *17*, 805–867.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, *57*, 97–109.

Hoeffding, W. (1963). Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, *58*, 13–30.

Ishii, H., Başar, T., & Tempo, R. (2005). Randomized algorithms for synthesis of switching rules for multimodal systems. *IEEE Transactions on Automatic Control*, *50*, 754–767.

Ishii, H., & Tempo, R. (2009). Probabilistic sorting and stabilization of switched systems. *Automatica*, *45*, 776–782.

Ishii, H., & Tempo, R. (2010). Distributed randomized algorithms for the PageRank computation. *IEEE Transactions on Automatic Control*, *55*, 1987–2002.

Kalai, A. T., & Vempala, S. (2006). Simulated annealing for convex optimization. *Mathematics of Operations Research*, *31*(2), 253–266.

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

Kanev, S., & Verhaegen, M. (2006). The randomized ellipsoid algorithm for constrained robust least squares problems. In *Probabilistic and randomized methods for design under uncertainty* (pp. 223–242). London: Springer-Verlag.

Karpinski, M., & Macintyre, A. (1997). Polynomial bounds for VC dimension of sigmoidal and general Pfaffian neural networks. *Journal of Computer and System Sciences*, *54*, 169–176.

Kettani, H., & Barmish, B. R. (2008). A new Monte Carlo circuit simulation paradigm with specific results for resistive networks. *IEEE Transactions on Circuits and Systems. I. Regular Papers*, *53*, 1289–1299.

Koltchinskii, V., Abdallah, C. T., Ariola, M., & Dorato, P. (2001). Statistical learning control of uncertain systems: theory and algorithms. *Applied Mathematics and Computation*, *120*, 31–43.

Koltchinskii, V., Abdallah, C. T., Ariola, M., Dorato, P., & Panchenko, D. (2000). Improved sample complexity estimates for statistical learning control of uncertain systems. *IEEE Transactions on Automatic Control*, *46*, 2383–2388.

Lagoa, C. M. (2003). Aprobabilistic enhancement of classical robustness margins: a class of nonsymmetric distributions. *IEEE Transactions on Automatic Control*, *48*(11), 1990–1994.

Lagoa, C., Dabbene, F., & Tempo, R. (2008). Hard bounds on the probability of performance with application to circuit analysis. *IEEE Transactions on Circuits and Systems. I. Regular Papers*, *55*, 3178–3187.
Lagoa, C. M., Shcherbakov, P. S., & Barmish, B. R. (1998). Probabilistic enhancement of classical robustness margins: the unirectangularity concept. *Systems & Control Letters*, *35*, 31–43.
Lecchini-Visintini, A., Glover, W., Lygeros, J., & Maciejowski, J. M. (2006). Monte Carlo optimization for conflict resolution in air traffic control. *IEEE Transactions on Intelligent Transportation Systems*, *7*(4), 470–482.
Lecchini-Visintini, A., Lygeros, A., & Maciejowski, J. (2010). Stochastic optimization on continuous domains with finite-time guarantees by Markov chain Monte Carlo methods. *IEEE Transactions on Automatic Control*, *55*, 2858–2863.
Lorefice, L., Pralio, B., & Tempo, R. (2009). Randomization-based control design for mini-UAVs. *Control Engineering Practice*, *17*, 974–983.
Lovász, L. (1999). Hit-and-run mixes fast. *Mathematical Programming*, *86*, 443–461.
Lu, B., & Wu, F. (2006). Probabilistic robust linear parameter-varying control of an F-16 aircraft. *Journal of Guidance, Control, and Dynamics*, *29*, 1454–1460.
Ma, W., Sznaier, M., & Lagoa, C. M. (2007). A risk adjusted approach to robust simultaneous fault detection and isolation. *Automatica*, *43*(3), 499–504.
Mitchell, J. E. (2003). Polynomial interior point cutting plane methods. *Optimization Methods & Software*, *18*, 507–534.
Nesterov, Yu. (1995). Complexity estimates of some cutting plane methods based on the analytic barrier. *Mathematical Programming, Series B*, *69*, 149–176.
Oishi, Y. (2007a). Polynomial-time algorithms for probabilistic solutions of parameter-dependent linear matrix inequalities. *Automatica*, *43*, 538–545.
Oishi, Y. (2007b). Probabilistic design of a robust state-feedback controller based on parameter-dependent Lyapunov functions. *Automatica*, *43*(3), 538–545.
Pallottino, L., Scordio, V. G., Frazzoli, E., & Bicchi, A. (2007). Decentralized cooperative policy for conflict resolution in multi-vehicle systems. *IEEE Transactions on Robotics*, *23*, 1170–1183.
Parrondo, J. M., & van den Broeck, C. (1993). Vapnik–Chervonenkis bounds for generalization. *Journal of Physica A*, *26*, 2211–2223.
Polyak, B. T., & Tempo, R. (2001). Probabilistic robust design with linear quadratic regulators. *Systems & Control Letters*, *43*, 343–353.
Rubinstein, R. Y., & Kroese, D. P. (2008). *Simulation and the Monte-Carlo method*. New York: Wiley.
Sauer, N. (1972). On the density of families of sets. *Journal of Combinatorial Theory. Series A*, *13*, 145–147.
Scherer, C., Gahinet, P., & Chilali, M. (1997). Multiobjective output feedback control via LMI optimization. *IEEE Transactions on Automatic Control*, *42*, 896–911.
Shcherbakov, P., & Dabbene, F. (2011). On the generation of random stable polynomials. European Journal of Control, 17 (in press).
Shor, N. Z. (1977). Cut-off method with space dilation in convex programming problems. *Cybernetics*, *13*(3), 94–96.
Smith, R. L. (1984). Efficient Monte-Carlo procedures for generating points uniformly distributed over bounded regions. *Operations Research*, *32*, 1296–1308.
Spall, J. C. (2003a). Estimation via Markov chain Monte Carlo. *IEEE Control Systems Magazine*, *23*, 34–45.
Spall, J. C. (2003b). *Introduction to stochastic search and optimization: estimation, simulation, and control*. New York: Wiley.
Stengel, R. F. (1980). Some effects of parameter variations on the lateral-directional stability of aircraft. *AIAA Journal of Guidance and Control*, *3*, 124–131.
Sznaier, M., Lagoa, C. M., & Mazzaro, M. C. (2005). An algorithm for sampling subsets of H_∞ with applications to risk-adjusted performance analysis and model (in)validation. *IEEE Transactions on Automatic Control*, *50*, 410–416.
Tempo, R., Bai, E.-W., & Dabbene, F. (1997). Probabilistic robustness analysis: explicit bounds for the minimum number of samples. *Systems & Control Letters*, *30*, 237–242.
Tempo, R., Calafiore, G. C., & Dabbene, F. (2005). *Communications and control engineering series*, *Randomized algorithms for analysis and control of uncertain systems*. London: Springer-Verlag.
Tempo, R., & Ishii, H. (2007). Monte Carlo and Las Vegas randomized algorithms for systems and control: an introduction. *European Journal of Control*, *13*, 189–203.
Tremba, A., Calafiore, G. C., Dabbene, F., Gryazina, E., Polyak, B. T., & Shcherbakov, P. S. et al. (2008). RACT: Randomized algorithms control toolbox for MATLAB. In *17th IFAC world congress. Seoul. July*.
Vapnik, V. N. (1998). *Statistical learning theory*. New York: Wiley.
Vapnik, V. N., & Chervonenkis, A. Ya. (1971). On the uniform convergence of relative frequencies to their probabilities. *Theory of Probability and its Applications*, *16*, 264–280.
Vapnik, V. N., & Chervonenkis, A. Ya. (1981). Necessary and sufficient conditions for the uniform convergence of means to their expectations. *Theory of Probability and its Applications*, *26*, 532–553.
Vidyasagar, M. (1998). Statistical learning theory and randomized algorithms for control. *IEEE Control Systems Magazine*, *18*, 69–85.
Vidyasagar, M. (2001). Randomized algorithms for robust controller synthesis using statistical learning theory. *Automatica*, *37*, 1515–1528.
Vidyasagar, M. (2002). *Learning and generalization: with applications to neural networks* (second ed.) New York: Springer-Verlag.
Vidyasagar, M., & Blondel, V. (2001). Probabilistic solutions to some NP-hard matrix problems. *Automatica*, *37*, 1397–1405.
Wada, T., & Fujisaki, Y. (2007). Sequential randomized algorithms for robust optimization. In *46th IEEE conference on decision and control* (pp. 6190–6195).
Wang, Q., & Stengel, R. F. (2005). Robust nonlinear flight control of a high performance aircraft. *IEEE Transactions on Control Systems Technology*, *13*, 15–26.
Yudin, D. B., & Nemirovski, A. S. (1977). Informational complexity and efficient methods for solving complex extremal problems. *Matekon*, *13*(3), 25–45.
Zhou, K., Doyle, J. C., & Glover, K. (1996). *Robust and optimal control*. Upper Saddle River: Prentice-Hall.
Zhou, T., & Feng, C. (2006). Uniform sample generation from contractive block Toeplitz matrices. *IEEE Transactions on Automatic Control*, *51*, 1559–1565.

Giuseppe C. Calafiore received the "Laurea" degree in Electrical Engineering from Politecnico di Torino in 1993, and the Ph.D. in Information and System Theory from Politecnico di Torino in 1997. He currently serves as Associate Professor at the "Dipartimento di Automatica e Informatica", Politecnico di Torino, Italy. Dr. Calafiore held visiting positions at Information Systems Laboratory, Stanford University, in 1995, at Ecole Nationale Superieure de Techniques Avencees (ENSTA), Paris, in 1998, at the University of California at Berkeley, in 1999, 2003 and 2007. In 2010 he was a visiting Senior Fellow at the Institute of Pure and Applied Mathematics, UCLA. Dr. Calafiore is Associate Editor for the IEEE Transactions on Automation Science and Engineering, for the IEEE Transactions on Systems, Man, and Cybernetics, and for the Journal Europeen des Systemes Automatises. His research interests are in the fields of analysis and control of uncertain systems, convex optimization, probabilistic algorithms and risk management. He is a recipient of the 2008 IEEE Axelby Award. He is coauthor of 6 books and over 120 publications on international journals and conference proceedings.

Fabrizio Dabbene received the Laurea degree in Electrical Engineering in 1995 and the Ph.D. degree in Systems and Computer Engineering in 1999, both from Politecnico di Torino, Italy. He is currently holding a tenured research position at the institute IEIIT of the National Research Council (CNR) of Italy. He has held visiting and research positions at the Department of Electrical Engineering, University of Iowa and at the RAS Institute of Control Science, Moscow. Dr. Dabbene also holds strict collaboration with Politecnico di Torino, where he teaches several courses in Systems and Control, and with Università degli Studi di Torino, where he does research on modeling of environmental systems.

His research interests include robust control and identification of uncertain systems, randomized algorithms for systems and control, convex optimization and modeling of environmental systems. On these topics he has been the organizer or co-organizer of several invited sessions and special courses, and of two Conference Workshops (CDC-00 and MSC-08). He is author or co-author of more than 80 publications, which include more than 25 articles published in international journals. He is author of the monograph Randomized Algorithms for Analysis and Control of Uncertain Systems and editor of the book Probabilistic and Randomized Methods for Design under Uncertainty, both published by Springer. He is one of the main developers of the recently released Matlab Toolbox RACT (Randomized Algorithms Control Toolbox). He has been awarded the "Outstanding Paper Award 2010" by EurAgeng.

Dr. Dabbene is Senior Member of the IEEE. He serves as Associate Editor for the IFAC Journal Automatica, Elsevier Science and for the IEEE Transactions on Automatic Control. From 2002 to 2008 he has been member of the Conference Editorial Board of the IEEE Control System Society. He is currently chairing IEEE Technical Committee on Computational Aspects of Control System Design (CACSD). He has been member of the Program Committee for the joint conference CDC/ECC-05, for M.Sc. 2008 and for CDC 2009, and he has served as member of several CDC and IFAC Award Committees. He was Program Chair of the CACSD Symposium for M.Sc. 2010.

Roberto Tempo is currently a Director of Research of Systems and Computer Engineering at the IEIIT Institute, National Research Council (CNR), Politecnico di Torino, Italy. He has held visiting positions at various institutions, including The University of Tokyo, Kyoto University, University of Illinois at Urbana-Champaign and Columbia University in New York.

Dr. Tempo's research activities are mainly focused on analysis and design of complex systems with uncertainty, and related control applications. He is author or co-author of more than 170 research papers published in international journals, books and conferences. He is a co-author of the book Randomized Algorithms for Analysis and Control of Uncertain Systems, Springer-Verlag, 2005. He is currently an Editor and Deputy Editor-in-Chief of Automatica and he has served as Editor for Technical Notes and Correspondence of the IEEE Transactions on Automatic Control.

Roberto Tempo received the Outstanding Paper Prize Award from the International Federation of Automatic Control (IFAC) for a paper published in Automatica, and the Distinguished Member Award from the IEEE Control Systems Society. He is a Fellow of the IEEE and a Fellow of the IFAC. He was Program Chair of the first joint IEEE Conference on Decision and Control and European Control Conference, Seville, Spain, 2005. In 2010 he has served as President of the IEEE Control Systems Society.