A Sampling-and-Discarding Approach to Chance-Constrained Optimization: Feasibility and Optimality *

M.C. Campi[†‡] S. Garatti[§]

Communicated by B.T. Polyak

Abstract

In this paper, we study the link between a Chance-Constrained optimization Problem (CCP) and its sample counterpart (SP). SP has a finite number, say N, of sampled constraints. Further, some of these sampled constraints, say k, are discarded, and the final solution is indicated by $x^*_{N,k}$. Extending previous results on the feasibility of sample convex optimization programs, we establish the feasibility of $x^*_{N,k}$ for the initial CCP problem.

Constraints removal allows one to improve the cost function at the price of a decreased feasibility. The cost improvement can be inspected directly from the optimization result, while the theory here developed permits to keep control on the other side of the coin, the feasibility of the obtained solution. In this way, trading feasibility for performance is put on solid mathematical grounds in this paper.

The feasibility result here obtained applies to all chance-constrained optimization problems with convex constraints, and has the distinctive feature that it holds true irrespective of the algorithm used to discard k constraints in the SP problem. For constraints discarding, one can thus, e.g., resort to one of the many methods introduced in the literature to solve chance-constrained problems with discrete distribution, or even use a greedy algorithm, which is computationally very low-demanding, and the feasibility result remains intact.

We further prove that, if constraints in the SP problem are optimally removed – i.e., one deletes those constraints leading to the largest possible cost improvement –, then a precise optimality link to the original chance-constrained problem CCP in addition holds.

Keywords: Chance-constrained optimization, Stochastic optimization, Convex optimization, Sample-based optimization, Randomized methods.

1 Introduction

Letting $\mathcal{X} \subseteq \mathbb{R}^d$ be a convex and closed domain of optimization, consider a family of constraints $x \in \mathcal{X}_\delta$ parameterized in $\delta \in \Delta$, where the sets $\mathcal{X}_\delta$ are convex and closed. Convexity of the sets $\mathcal{X}_\delta$ is an assumption

*This paper was supported by the MIUR project "Identification and adaptive control of industrial systems".

†Corresponding Author.

‡Dipartimento di Ingegneria dell'Informazione - Università di Brescia, via Branze 38, 25123 Brescia, Italia. E-mail: marco.campi@ing.unibs.it, web-site: http://bsing.ing.unibs.it/∼campi/

§Dipartimento di Elettronica e Informazione - Politecnico di Milano, piazza Leonardo da Vinci 32, 20133 Milano, Italia. E-mail: sgaratti@elet.polimi.it, web-site: http://home.dei.polimi.it/sgaratti/

in effect throughout this paper. δ is the *uncertain* parameter and it describes different instances of an uncertain optimization scenario. Adopting a probabilistic description of uncertainty, we suppose that the support Δ for δ be endowed with a σ-algebra $\mathcal{D}$ and that a probability measure $\mathbb{P}$ be defined over $\mathcal{D}$. $\mathbb{P}$ describes the probability with which the uncertain parameter δ takes value in Δ. Then, a chance-constrained optimization program is written as:

$$\begin{aligned} \text{CCP}_\epsilon: \quad & \min_{x\in\mathcal{X}} c^T x \\ & \text{s.t. } \mathbb{P}\{\delta : x \in \mathcal{X}_\delta\} \geq 1-\epsilon. \end{aligned} \tag{1}$$

Here, the σ-algebra $\mathcal{D}$ is large enough, so that $\{\delta : x \in \mathcal{X}_\delta\} \in \mathcal{D}$, that is $\{\delta : x \in \mathcal{X}_\delta\}$ is a measurable set. Also, linearity of the objective function is without loss of generality, since any objective of the kind $\min_{x\in\mathcal{X}} c(x)$, where $c(x) : \mathcal{X} \to \mathbb{R}$ is a convex function, can be re-written as $\min_{x\in\mathcal{X}, y\geq c(x)} y$, where y is a scalar variable.

In CCP_ϵ, constraint violation is tolerated, but the violated constraint set must be no larger than ϵ. This parameter ϵ allows one to trade robustness for performance: the optimal objective value J^*_ϵ of CCP_ϵ is a decreasing function of ϵ and provides a quantification of such a trade-off. Depending on the application at hand, which can cover a wide range from control to prediction and from engineering design to financial economics, ϵ can take different values and has not necessarily to be thought of as a "small" parameter.

Chance-constrained programming has been around for a long time, at least since the work of Charnes, Cooper and Symonds in the fifties, see [1]. In [1], however, only individual chance-constraints were considered. Joint probabilistic constraints, as in (1), were first considered by Miller and Wagner, [2], in an independent context, while a general theory is due to the work of Prèkopa, see [3, 4]. Prèkopa also introduced the convexity theory based on logconcavity, a fundamental step toward solvability of a large class of chance-constrained problems. [5, 6, 7] are excellent references providing a broad overview on logconcavity theory in stochastic programming, and related results. Yet another study about the convexity of chance-constrained problems is [8], while convex approximations of chance-constrained problems are considered in [9, 10, 11, 12]. Stability of the solution under perturbation of the chance-constrained problem is studied in [13, 14]. The case of discrete distribution is dealt with using efficient points in [15, 16, 17, 18, 19, 20], see also [21, 22, 23] for other studies.

Though chance-constrained problems can be efficiently solved in some notable cases as outlined above, it remains true that the feasible set of CCP_ϵ is in general non-convex in spite of the convexity of the sets $\mathcal{X}_\delta$. Consequently, an exact numerical solution of CCP_ϵ is in general hard to find.

1.1 Contribution of this paper

In this paper, we consider sample-based approximations of chance-constrained optimization problems. Precisely, by replacing Δ with a finite sample of independent instances $\delta^{(1)}, \delta^{(2)}, \ldots, \delta^{(N)} \in \Delta$ distributed according to $\mathbb{P}$, an optimization program with a *finite* number of constraints is obtained (the sample optimization program), and we further allow one for removal of constraints from this finite set to improve the cost value[1]. Our main objective is to *quantitatively relate the sample-based approximation to the initial chance-constrained optimization program and to provide sample size results on the number of constraints to sample to make the solution of the former a feasible solution of the latter.*

Sample-based approximations are chance-constrained problems with discrete distribution. Chance-constrained problems with discrete distribution have been previously considered in many papers, e.g. [15, 21, 16, 17, 18, 19, 22, 20, 23], where effective resolution methods have been introduced. Our standpoint here is that *one can use anyone of these resolution methods, or even any other method, e.g., based on a greedy removal of the constraints, and our feasibility Theorem 2.1 always holds true to establish the feasibility of the solution with respect to the initial* CCP_ϵ *problem.* This theorem precisely establishes that:

> if N constraints are sampled and k of them are eliminated according to any arbitrary rule, then the solution that satisfies the remaining $N-k$ constraints is, with high confidence, feasible for the chance-constrained optimization program CCP_ϵ in (1), provided that N and k satisfy a certain condition (3).

This theorem justifies, at a very deep theoretical level, the use of sampling in chance-constrained. Its strength is that it applies to *all* chance-constrained problems with convex constraints. Moreover, condition (3) is very tight, in the sense that it returns values for N and k close to the best possible values guaranteeing feasibility, a fact also shown in this paper.

Theorem 2.1 opens up practical routes to address chance-constrained optimization problems: after sampling, constraints are removed according to any procedure, and, at the end of the elimination process, the actually incurred optimization cost is inspected for satisfaction, while Theorem 2.1 allows one to keep control on the feasibility of the obtained solution. To illustrate the result in more concrete terms, we anticipate here in Figure 1 a graph referring to a numerical example that we shall develop in full later in Section 3. In the figure, the curve with circled markers represents the optimization cost of the sample-based solution as a function of the number k of removed constraints, while the curve with squared markers represents the constraints violation of the sample-based solution as guaranteed by the theory developed in this paper.

[1] These $\delta^{(i)}$ can have in applications one of the following two interpretations: 1. they are artificially extracted by the user from Δ; 2. they come as observations. Section 4.1 provides a more detailed discussion of these interpretations.

Based on an assessment of the two curves, the user can choose his favorite compromise between performance and violation.

Depending on the elimination rule, the incurred objective value can be close or less close to the optimal objective value J_ϵ^* of the CCP_ϵ optimization program. Working on this aspect, we further prove in Theorem 6.1 that, when k constraints are optimally removed, the objective value gets close to J_ϵ^* in a sense precisely stated in the theorem.

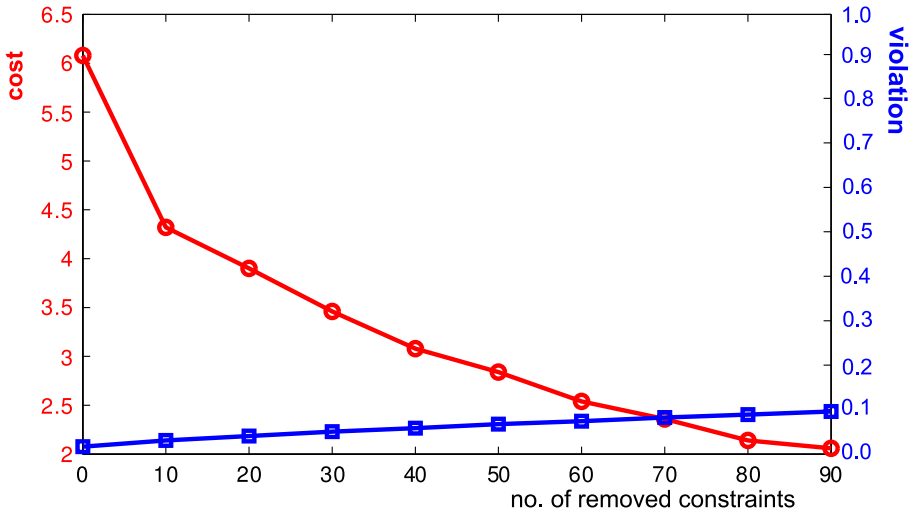

Figure 1: Optimization cost (curve with circled markers) and constraints violation (curve with squared markers) against the number of the removed constraints in the x-axis.

1.2 Connection with other results on sample-based methods

Sample-based techniques, also known as Monte Carlo methods, have been around for decades, but only recently they have started to spread in the context of stochastic optimization due to the increase of computing capabilities, see e.g. [24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35], and sample-based optimization has also been used in connection of various application domains, see e.g. [36, 37, 38, 39, 40]. We here give a brief resume of previous results about the feasibility of sample-based optimization programs, that have a close relation to the present paper.

Chapter 5 in [34] provides a thorough presentation of the conditions for the sample-based approximation to asymptotically reconstruct the original chance-constrained problem. Unlike Chapter 5 in [34], the present paper deals with a *finite-sample* analysis, in that we want to determine the sample size N guaranteeing a given level of approximation. Finite-sample properties are also the topic of the interesting paper [41],

which presents an analysis applicable in a set-up complementary to that of the present paper. Specifically, feasibility results are established for possibly non-convex constraints, provided that the optimization domain has finite cardinality or in situations that can be reduced to this finite cardinality set-up. An excellent contribution with many results for possibly non-convex constraints is also [42].
The approach of the present paper builds on the so-called "scenario" approach of [43, 44, 45]. *The fundamental progress over [43, 44, 45] here made is that the results in [43, 44, 45] do not allow one for constraints removal*, a possibility which is of crucial importance any time one wants to trade feasibility for performance.

1.3 Structure of the paper

In next Section 2, we formally introduce the sample-based approach with constraints elimination to prepare the terrain for the feasibility Theorem 2.1 given at the end of the same section. To avoid breaking the flow of discourse, the proof of Theorem 2.1 is postponed to Section 5, while Section 3 presents how to trade feasibility for performance and Section 4 provides complimentary theoretical material. Finally, optimality results are given in Section 6.

2 Sample-based chance-constrained optimization: feasibility results

Suppose that N samples $\delta^{(1)}, \delta^{(2)}, \ldots, \delta^{(N)}$ independent and identically distributed according to the probability measure $\mathbb{P}$ are available. The idea behind the scenario approach of [43, 44, 45] is to substitute the vast multitude of constraints in the infinite initial domain Δ with these N constraints only, and to find the optimal solution that satisfies *all* of these N constraints. If all the N constraints are enforced, however, one cannot expect that good approximations of chance-constrained solutions are obtained[2]. Thus, we allow in this paper for violating part of the sampled constraints to improve the optimization value. A general removal procedure is formalized in the following definition.

Definition 2.1 *Let $k < N$. An algorithm $\mathcal{A}$ for constraints removal is any rule by which k constraints out of a set of N constraints are selected and removed. The output of $\mathcal{A}$ is the set $\mathcal{A}\{\delta^{(1)}, \ldots, \delta^{(N)}\} = \{i_1, \ldots, i_k\}$ of the indexes of the k removed constraints.* □

The fact that $\mathcal{A}$ can be any removal algorithm provides us with an opportunity to pick the most suitable algorithm for the situation at hand, selecting from a range that goes from a handy greedy algorithm to the

[2]The fundamental role of constraint removal can be fully appreciated by way of a simple 1-dimensional example. We provide such an example in Appendix A, part A.1, for the benefit of the reader who is interested to gain more insight.

optimal algorithm where k constraints are eliminated to best improve the cost objective. All methods in [15, 21, 16, 17, 18, 19, 22, 20, 23] can be used and thus this paper leverages on these previous contributions. Another choice is a recursive optimal elimination of groups of p, with $p \ll k$, constraints at a time (when $p = 1$ the greedy algorithm is recovered). Yet another choice consists in progressively updating the solution by eliminating all the active constraints at the currently reached solution.

The sample-based optimization program where k constraints are removed as indicated by $\mathcal{A}$ is expressed as

$$\begin{aligned} \mathrm{SP}^{\mathcal{A}}_{N,k}: \quad & \min_{x\in\mathcal{X}} c^T x \\ & \text{s.t. } x \in \mathcal{X}_{\delta^{(i)}},\ i \in \{1,\dots,N\} - \mathcal{A}\{\delta^{(1)},\dots,\delta^{(N)}\}, \end{aligned}$$

and its solution will be hereafter indicated as $x^*_{N,k}$.

We introduce the following assumptions.

Assumption 2.1 *Every optimization problem subject to only a finite subset F of constraints from Δ, i.e.*

$$\min_{x\in\mathcal{X}} c^T x \quad \text{s.t. } x \in \mathcal{X}_\delta, \quad \delta \in F \subseteq \Delta, \tag{2}$$

is feasible, and its feasibility domain has a nonempty interior. Moreover, the solution of (2) *exists and is unique.* □

Assumption 2.1 requires that problems with finitely many constraints be feasible. It applies to most situations of practical interest, notably in all chance-constrained minmax problems where one minimizes the maximum value of an uncertain cost $c_\delta(x)$, with maximum taken with respect to all $\delta \in \Delta_\epsilon$ and minimum with respect to $x \in \mathcal{X}$ and to the choice of the subset $\Delta_\epsilon \subseteq \Delta$ among sets whose probability is at least $1-\epsilon$, viz. $\min_{x\in\mathcal{X},\Delta_\epsilon} \max_{\delta\in\Delta_\epsilon} c_\delta(x)$. This problem can be rewritten within the formalism of (1) as $\min_{x\in\mathcal{X},h\in\mathbb{R}} h$ s.t. $\mathbb{P}\{\delta : c_\delta(x) \le h\} \ge 1-\epsilon$. One example of this type of problems is provided in Section 3.

We also introduce the following assumption on algorithm $\mathcal{A}$.

Assumption 2.2 *Almost surely with respect to the multi-sample $(\delta^{(1)},\dots,\delta^{(N)})$, the solution $x^*_{N,k}$ of the sample-based optimization program $SP^{\mathcal{A}}_{N,k}$ violates all the k constraints that $\mathcal{A}$ has removed.* □

This assumption requires that $\mathcal{A}$ chooses constraints whose removal improves the solution by violating the removed constraints, and it rules out for example algorithms that remove inactive constraints only. Thus, this assumption is very natural and reflects the fact that constraints are removed for the purpose of improving the optimization value. In general, Assumption 2.2 can be made true by simply incorporating in $\mathcal{A}$ a

test to verify if the removed constraints are indeed violated and, if not, by further allowing the algorithm to remove other constraints. The "almost surely" specification at the beginning of the assumption is introduced because for non-generic $\delta^{(1)}, \ldots, \delta^{(N)}$ samples, that e.g. cluster together ($\delta^{(1)} = \delta^{(2)} = \ldots = \delta^{(N)}$), removal of k constraints so that they are violated may be impossible.

Finally, we introduce the following definition.

Definition 2.2 (violation probability) *The* violation probability *of a given $x \in \mathcal{X}$ is defined as* $V(x) = \mathbb{P}\{\delta \in \Delta : x \notin \mathcal{X}_\delta\}$. □

The next Theorem 2.1 provides theoretical guarantees that $V(x^*_{N,k}) \leq \epsilon$, i.e. that the solution $x^*_{N,k}$ of the optimization program $\mathrm{SP}^{\mathcal{A}}_{N,k}$ is feasible for problem CCP_ϵ. Note that $x^*_{N,k}$ is a random variable because it depends on the random multi-sample $(\delta^{(1)}, \ldots, \delta^{(N)})$, so that its violation probability $V(x^*_{N,k})$ is a random variable too defined over the product space $\Delta^N = \Delta \times \cdots \times \Delta$ endowed with the product σ-algebra $\mathcal{D}^N = \mathcal{D} \otimes \cdots \otimes \mathcal{D}$ and the probability measure $\mathbb{P}^N = \mathbb{P} \times \cdots \times \mathbb{P}$, where the probability is a product probability because the samples $\delta^{(i)}$ are independent. Thus, $V(x^*_{N,k})$ can be less than ϵ for some multi-samples $(\delta^{(1)}, \ldots, \delta^{(N)})$ and not for others, and the theorem establishes the condition under which $V(x^*_{N,k}) > \epsilon$ has any arbitrarily small probability β.

Theorem 2.1 *Let $\beta \in (0, 1)$ be any small confidence parameter value. If N and k are such that*

$$\binom{k+d-1}{k} \sum_{i=0}^{k+d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i} \leq \beta \tag{3}$$

(recall that d is the number of optimization variables), then $\mathbb{P}^N\{V(x^*_{N,k}) \leq \epsilon\} \geq 1 - \beta$. □

In the theorem the measurability of set $\{V(x^*_{N,k}) \leq \epsilon\}$ is taken as an assumption; the same convention applies elsewhere to other subsets of Δ^N. The theorem holds true for any optimization problem with convex constraints, any constraints removal algorithm $\mathcal{A}$, and any probability measure $\mathbb{P}$. To avoid breaking the flow of presentation, the proof is given in the next Section 5.1.

Theorem 2.1 is a *feasibility* theorem and says that the solution $x^*_{N,k}$ obtained by inspecting N constraints only is a feasible solution for CCP_ϵ with high probability $1 - \beta$, provided that N and k fulfill condition (3).

Formula (3) establishes a relation among variables N, k, ϵ, and β. A typical use of this formula consists in selecting an N within the computational limit of the used solver, ϵ according to the acceptable level of risk, and β small enough to be negligible, e.g. $\beta = 10^{-10}$, and computing from (3) the largest number k of constraints that can be discarded.

For an easy visualization of Theorem 2.1, we have represented in Figure 2 the region in the N, k space such that condition (3) is satisfied when $\epsilon = 0.1$, $\beta = 10^{-10}$ and $d = 5$. The interpretation is that, if any

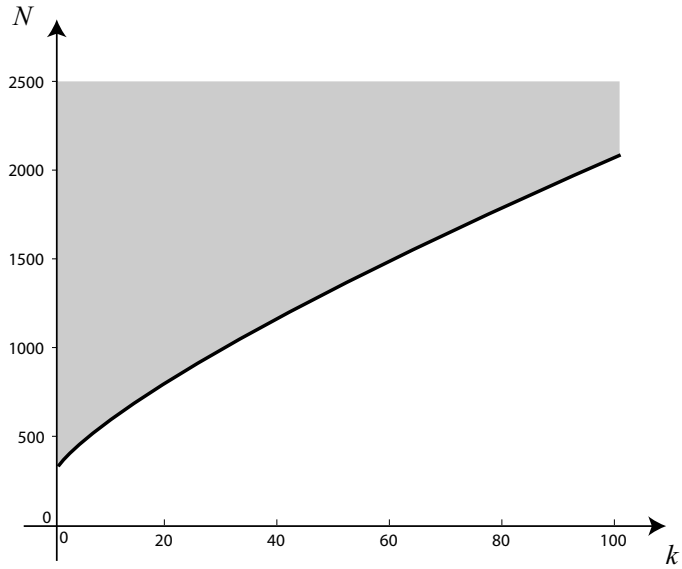

Figure 2: Grey region: values of N and k satisfying condition (3) for $\epsilon = 0.1$, $\beta = 10^{-10}$ and $d = 5$.

pair (N, k) is picked from the grey region, then, if N constraints are sampled and k of them are removed, the obtained solution is, with high confidence $1 - 10^{-10}$, feasible for the chance-constrained problem with parameter $\epsilon = 0.1$. As we shall discuss in detail in the Section 4.2, formula (3) provides a tight evaluation for N and k.

3 Trading feasibility for performance

In some applications, one may want to reach a suitable compromise between violation and performance, in which case one discards a progressively increasing number k of constraints, while inspecting the corresponding cost improvement for satisfaction. Theorem 2.1 holds for given N and k; yet, by a repeated application of the theorem for $k = k_1, k_2, \ldots, k_M$, one concludes that

$$\mathbb{P}^N\{V(x^*_{N,k_i}) \leq \epsilon_i,\ \forall i = 1, \ldots, M\} \geq 1 - \sum_{i=1}^{M} \beta_i. \tag{4}$$

This result permits one to keep simultaneous control on the obtained violations after an increasing number of constraints are eliminated. Thus, after the user has computed x^*_{N,k_i}, $i = 1, \ldots, M$, and the corresponding cost values $c^T x^*_{N,k_i}$, he can select his favorite violation/performance trade-off by assessing the ϵ_i's against

the $c^T x^*_{N,k_i}$ values. We also note that having a sum of β_i in (4) is not a hurdle since the β_i can be chosen to be very small in normal situations. To illustrate ideas, we provide next an example taken from linear regression.

Example 3.1 (minimax regression) *The $N = 2000$ points (u_i, y_i), $i = 1, \ldots, 2000$, displayed in Figure 3 are independently generated in $\mathbb{R}^2$ according to an* unknown *probability measure $\mathbb{P}$. We want to construct*

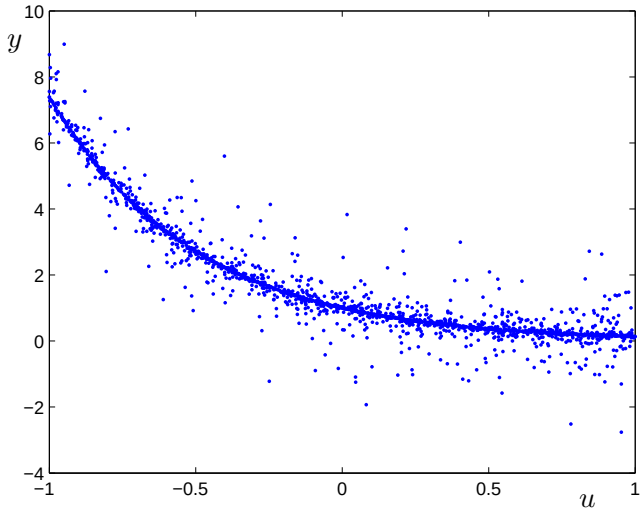

Figure 3: (u_i, y_i) data points.

an interpolating polynomial of degree 3, $y = x_0 + x_1 u + x_2 u^2 + x_3 u^3$, where $x_0, \ldots, x_3$ are parameters to be chosen, so that a strip of minimal vertical width centered around the polynomial contains all the generated points. In mathematical terms, this problem can be cast as the following optimization program

$$\min_{x_0,\ldots,x_4} x_4 \quad \text{s.t.} \quad \left|y_i - [x_0 + x_1 u_i + x_2 u_i^2 + x_3 u_i^3]\right| \leq x_4, \quad i = 1, \ldots, 2000, \tag{5}$$

and the optimal polynomial is named minimax regressor. Problem (5) *is a sample optimization program where $\delta^{(i)} = (u_i, y_i)$. The solution we have obtained with the data at hand is shown in Figure 4.*

The above problem can be interpreted as an identification problem where u is the input, y is the output, $\mathbb{P}$ is the probability measure describing an underlying data generation mechanism, the $N = 2000$ points are the data, and the strip is a data-based descriptor of the generator. Given the next input u, the interval in the strip corresponding to that u provides a prediction of the associated unseen y. Correspondingly, one would like to have a strip of small width so as to make a tight prediction, while keeping low the probability of not capturing the next unseen y, that is the probability that (u, y) falls outside the strip.

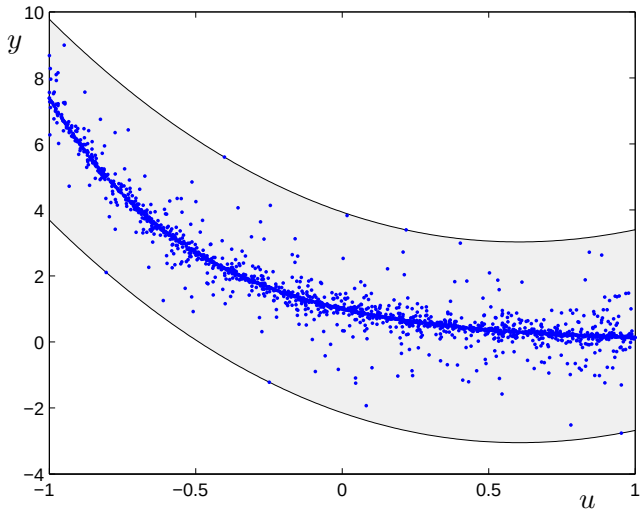

Figure 4: Strip containing all points.

To make the strip width smaller, we further removed some of the sampled (u_i, y_i) *points. Figure 5 depicts, stacked one on top of the other, the strips obtained by a greedy removal of* $k_i = 10, 20, \ldots, 90$ *points.*

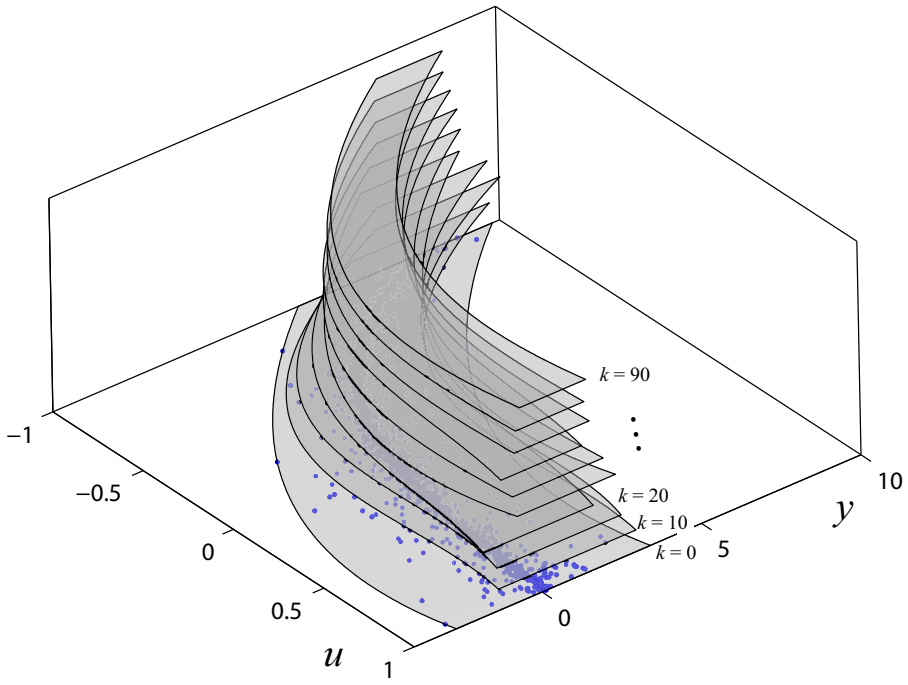

Figure 5: Strips with increasing points violation.

The corresponding strip widths are displayed in Table 1.

k_i	0	10	20	30	40	50	60	70	80	90
width	6.08	4.32	3.90	3.46	3.08	2.84	2.54	2.36	2.14	2.06

Table 1: Strip widths vs. k_i.

Turning to consider the reliability of the various strips obtained for different k_i values, from (5) *it is clear that each point corresponds to a constraint and the probability that (u, y) falls outside the i-th strip is the same as the probability of constraint violation $V(x^*_{N,k_i})$. Using $k_i = 0, 10, 20, \ldots, 90$ and $\beta_i = 10^{-10}$, $\forall i$, in* (3)*, this formula is satisfied for the ϵ_i values given in Table 2. These ϵ_i are upper bounds to the probability*

k_i	0	10	20	30	40	50	60	70	80	90
ϵ_i	0.017	0.031	0.041	0.051	0.059	0.068	0.075	0.083	0.090	0.097

Table 2: ϵ_i vs. k_i.

that (u, y) does not belong to the i-th strip and Table 2 should be assessed by the user against Table 1 for his violation/width favorite compromise. Figure 1 visually depicts the values in the tables. According to equation (4)*, the confidence in the final result will be $1 - \sum_{i=1}^{10} 10^{-10} = 1 - 10^{-9}$.* □

4 Remarks on Theorem 2.1

4.1 Data-based optimization

The example of the previous section offers an opportunity to broaden the discussion so far in relation to the interpretation and the applicability of the theoretical findings of this paper.

In some applications, Δ and $\mathbb{P}$ represent a model we introduce to describe uncertainty. Correspondingly, samples $\delta^{(i)}$ are extracted by us along the process of replacing the CCP_ϵ program with its sample counterpart $\text{SP}^{\mathcal{A}}_{N,k}$, having in mind the goal of making the problem computationally tractable. This way of proceeding has been sometimes called "randomization", referring to the artificial process of sampling as part of the algorithmic solution methodology.

On the other hand, the example of Section 3 offers a second interpretation of the sampling scheme discussed in this paper: there $\delta^{(i)} = (u_i, y_i)$ is an observation that comes to us through a procedure of data acquisition. (u_i, y_i) is generated by an underlying system, so that Δ and $\mathbb{P}$ exist but are not known to us. The achievements of this paper find direct applicability to this context as well since all the optimization program $\text{SP}^{\mathcal{A}}_{N,k}$ uses are the samples $\delta^{(i)}$ (so that if the $\delta^{(i)}$ are available we do not need to know Δ and $\mathbb{P}$) and since applying the theoretical result of Theorem 2.1 does not require any knowledge of Δ and $\mathbb{P}$ as well (distribution-free result). For these reasons, we expect that the findings of this paper will have a significant impact in all fields

where data-based optimization is used including signal processing, system identification, statistical learning, financial economics, and others.

4.2 A remark on the quality of bound (3)

We discuss the bound (3) and its margin of improvement.

From Theorem 2.1, we have that $\binom{k+d-1}{k}\sum_{i=0}^{k+d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i}$ is an upper bound to $\mathbb{P}^N\{V(x^*_{N,k})>\epsilon\}$ valid for any removal algorithm $\mathcal{A}$ and for any optimization problem P (that is for any set of constraints $\mathcal{X}_\delta$, $\delta\in\Delta$, probability measure $\mathbb{P}$, and cost function c^Tx). I.e.

$$\sup_{\mathrm{P},\mathcal{A}}\mathbb{P}^N\{V(x^*_{N,k})>\epsilon\}\leq\binom{k+d-1}{k}\sum_{i=0}^{k+d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i}. \tag{6}$$

It is a fact that a lower bound to the same probability is

$$\sup_{\mathrm{P},\mathcal{A}}\mathbb{P}^N\{V(x^*_{N,k})>\epsilon\}\geq\sum_{i=0}^{k+d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i}, \tag{7}$$

and this sets a limit to the margin of improvement of the bound in Theorem 2.1. For a visual understanding

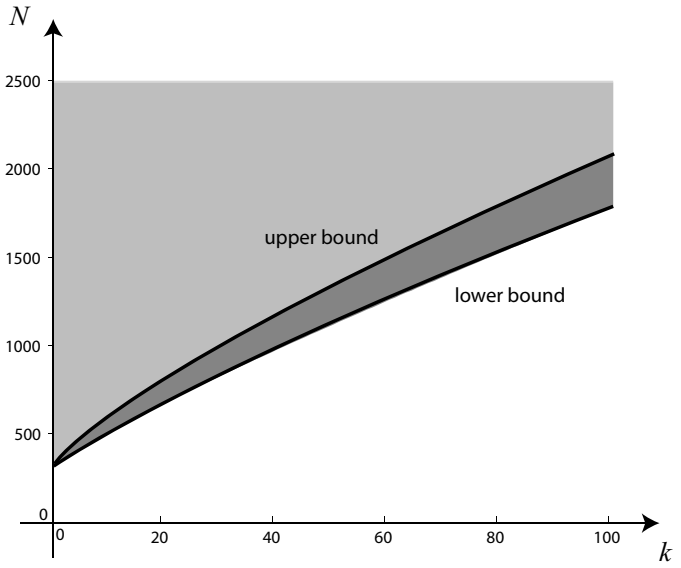

Figure 6: Grey region: values of N and k satisfying condition $\sum_{i=0}^{k+d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i}\leq\beta$; Light grey region: values of N and k satisfying condition $\binom{k+d-1}{k}\sum_{i=0}^{k+d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i}\leq\beta$; $\epsilon=0.1$, $\beta=10^{-10}$ and $d=5$.

of this result, in Figure 6 we have represented the region in the N,k space where condition $\sum_{i=0}^{k+d-1}\binom{N}{i}\epsilon^i(1-$

$\epsilon)^{N-i} \leq \beta$ is satisfied for $\epsilon = 0.1$, $\beta = 10^{-10}$ and $d = 5$ superimposed to the region obtained by using (6), i.e. by imposing $\binom{k+d-1}{k} \sum_{i=0}^{k+d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i} \leq \beta$ (this is the same region as in Figure 2).
The proof of (7) is provided in Section 5.2.

4.3 An explicit formula for k

Using the Chernoff bound for the Binomial tail (see e.g. Section 2.3.1 in [46]) yields

$$\sum_{i=0}^{k+d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i} \leq e^{-\frac{(\epsilon N - k - d + 1)^2}{2\epsilon N}}, \quad \text{for } \epsilon N \geq k + d - 1.$$

Moreover,

$$\binom{k+d-1}{k} = \frac{(k+d-1)!}{(d-1)!k!} \leq (k+d-1)(k+d-2)\cdots(k+1) \leq (k+d-1)^{d-1} \leq (\epsilon N)^{d-1},$$

where the last inequality follows from condition $\epsilon N \geq k + d - 1$.
Hence, the left-hand-side of (3) is bounded by $(\epsilon N)^{d-1} \cdot e^{-\frac{(\epsilon N - k - d + 1)^2}{2\epsilon N}}$. Given N, ϵ, β, and d, we compute a k such that

$$(\epsilon N)^{d-1} \cdot e^{-\frac{(\epsilon N - k - d + 1)^2}{2\epsilon N}} \leq \beta.$$

This latter equation can be rewritten as

$$(\epsilon N - k - d + 1)^2 \geq 2\epsilon N \ln \frac{(\epsilon N)^{d-1}}{\beta},$$

which, solved for k with the condition $\epsilon N \geq k + d - 1$, gives

$$k \leq \epsilon N - d + 1 - \sqrt{2\epsilon N \ln \frac{(\epsilon N)^{d-1}}{\beta}}. \tag{8}$$

Equation (8) is an explicit handy formula for k which can be used to compute the number of constraints that can be discarded. More precise evaluations can be obtained by numerically solving (3).

4.4 A remark on the fact that, although the bound is problem-independent, it is non-conservative

In [45], we showed that the only characteristic of an optimization problem that counts in determining the violation properties of a solution that satisfies all the sampled constraints is the number of support constraints of the sample-based optimization program. This number of support constraints is further bounded in [45] by d, the number of optimization variables, and this may introduce some conservatism. When we allow for constraints removal, as we do here, we see from formula (3) that d does not count as such, and it only appears summed to k: $d + k$, where k is the number of removed constraints. The presence of k,

which is deterministically known and chosen by the user, levels out the differences among optimization problems, so that results like those of this paper that hold true for any chance-constrained problem become nonconservative.

To gain extra quantitative insight on this point, notice that, letting $k_{\max}(N)$ be the maximum integer k such that (8) is satisfied for a given ϵ and β, we see from (8) that $\lim_{N\to\infty} \frac{k_{\max}(N)}{N} = \epsilon$. Indeed, the first term in the right hand side increases linearly in N whereas the other terms are sublinear and vanish for $N \to \infty$ when divided by N. The interpretation is that the empirical violation $\frac{k_{\max}(N)}{N}$ tends to the true violation ϵ as N is let increase, and this is the best possible result one can expect. For finite N, instead, random effects introduce a variability in the violation of the solution, so that the number of constraints that can actually be removed need to be less than $\epsilon \cdot N$ to obtain high confidence that the solution violates less then ϵ. Inspecting e.g. Figure 1 we see that to achieve a violation below 10% we have to eliminate no more that 90 constraints out of 2000, i.e. 4.5%, introducing a factor of approximately 2 between the two percentages.

4.5 A further comment on the choice of N

In applications where the $\delta^{(i)}$ are associated to observations and one has no active role in designing the experiment, the number N is dictated by the size of the available data record; this was e.g. the case in the example of Section 3. Other times, however, N can be selected as part of the experiment design. If so, formula (8) offers practical information for the selection of N: compatibly with other limitations, term

$$\frac{-d+1-\sqrt{2\epsilon N \ln \frac{(\epsilon N)^{d-1}}{\beta}}}{N}$$

should be made small enough by properly selecting N so that the mismatch between empirical violation $\frac{k}{N}$ and the theoretical limit ϵ is kept below a desired level.

5 Proofs

5.1 Proof of Theorem 2.1

Introducing the notation $\boldsymbol{\delta} := (\delta^{(1)}, \ldots, \delta^{(N)})$, in this proof we shall write $x^*_{N,k}(\boldsymbol{\delta})$ instead of $x^*_{N,k}$ to emphasize the stochastic nature of the solution. The "bad" multi-samples from Δ^N leading to a solution $x^*_{N,k}(\boldsymbol{\delta})$ violating a portion of constraints larger than ϵ form an event

$$B = \{\boldsymbol{\delta} \in \Delta^N : \ V(x^*_{N,k}(\boldsymbol{\delta})) > \epsilon\},$$

and, in these notations, the theorem statement can be rephrased as $\mathbb{P}^N\{B\} \leq \beta$.

Given a subset $I = \{i_1, \ldots, i_k\}$ of k indexes from $\{1, \ldots, N\}$, let $x_I^*(\boldsymbol{\delta})$ be the optimal solution of the optimization problem where the constraints with index in I have been removed, i.e.

$$x_I^*(\boldsymbol{\delta}) := \arg\min_{x \in \mathcal{X}} c^T x \quad \text{s.t. } x \in \mathcal{X}_{\delta^{(i)}}, \quad i \in \{1, \ldots, N\} - I. \tag{9}$$

Moreover, let

$$\Delta_I^N = \{\boldsymbol{\delta} \in \Delta^N : \ x_I^*(\boldsymbol{\delta}) \text{ violates the constraints } \delta^{(i_1)}, \ldots, \delta^{(i_k)}\}. \tag{10}$$

Thus, Δ_I^N contains the multi-samples such that removing the constraints with indexes in I leads to a solution $x_I^*(\boldsymbol{\delta})$ that violates all the removed constraints.
Since the solution of $\mathrm{SP}_{N,k}^{\mathcal{A}}$ almost surely violates k constraints (Assumption 2.2), it is clear that $x_{N,k}^*(\boldsymbol{\delta}) = x_I^*(\boldsymbol{\delta})$ for some I such that $\boldsymbol{\delta} \in \Delta_I^N$. Thus,

$$B = \{\boldsymbol{\delta} \in \Delta^N : \ V(x_{N,k}^*(\boldsymbol{\delta})) > \epsilon\} \subseteq \bigcup_{I \in \mathcal{I}} \{\boldsymbol{\delta} \in \Delta_I^N : \ V(x_I^*(\boldsymbol{\delta})) > \epsilon\} \tag{11}$$

up to a zero probability set, where $\mathcal{I}$ is the collection of all possible choices of k indexes from $\{1, \ldots, N\}$.
A bound for $\mathbb{P}^N\{B\}$ is now obtained by first bounding $\mathbb{P}^N\{\boldsymbol{\delta} \in \Delta_I^N : \ V(x_I^*(\boldsymbol{\delta})) > \epsilon\}$, and then summing over $I \in \mathcal{I}$.

Fix an $I = \{i_1, \ldots, i_k\}$, and write

$$\begin{aligned}
&\mathbb{P}^N\{\boldsymbol{\delta} \in \Delta_I^N : \ V(x_I^*(\boldsymbol{\delta})) > \epsilon\} \\
&\quad = \int_{(\epsilon,1]} \mathbb{P}^N\{\Delta_I^N | V(x_I^*(\boldsymbol{\delta})) = v\} \ \mathrm{d}F_V(v) \\
&\quad = \int_{(\epsilon,1]} \mathbb{P}^N\{x_I^*(\boldsymbol{\delta}) \text{ violates the constraints } \delta^{(i_1)}, \ldots, \delta^{(i_k)} | V(x_I^*(\boldsymbol{\delta})) = v\} \ \mathrm{d}F_V(v),
\end{aligned} \tag{12}$$

where F_V is the cumulative distribution function of the random variable $V(x_I^*(\boldsymbol{\delta}))$, and $\mathbb{P}^N\{\Delta_I^N | V(x_I^*(\boldsymbol{\delta})) = v\}$ is the conditional probability of the event Δ_I^N under the condition that $V(x_I^*(\boldsymbol{\delta})) = v$ (see eq.(17), § 7, Chapter II of [47]).
To evaluate the integrand in (12), remind that $V(x_I^*(\boldsymbol{\delta})) = v$ means that $x_I^*(\boldsymbol{\delta})$ violates constraints with probability v; then, owing to that the $\delta^{(i)}$ samples are independent, the integrand equals v^k. Substituting in (12) yields

$$\mathbb{P}^N\{\boldsymbol{\delta} \in \Delta_I^N : \ V(x_I^*(\boldsymbol{\delta})) > \epsilon\} = \int_{(\epsilon,1]} v^k \ \mathrm{d}F_V(v). \tag{13}$$

To proceed, we have now to appeal to a result on F_V from [45]: $F_V(v) \geq \bar{F}_V(v) := 1 - \sum_{i=0}^{d-1} \binom{N-k}{i} v^i (1-v)^{N-k-i}$, see Theorem 1 in [45] and recall that $F_V(v)$ is the cumulative distribution function of the violation of a solution obtained with $N-k$ constraints. This inequality is tight, i.e. it holds with equality for a whole class of optimization problems, that called "fully-supported" in [45], Definition 3.
Now, the integrand v^k in (13) is an increasing function of v, so that $F_V(v) \geq \bar{F}_V(v)$ implies that

$\int_{(\epsilon,1]} v^k \, \mathrm{d}F_V(v) \leq \int_{(\epsilon,1]} v^k \, \mathrm{d}\bar{F}_V(v)$. This can be verified by the calculation:

$$\begin{aligned}
\int_{(\epsilon,1]} v^k \, \mathrm{d}F_V(v) &= \text{[Theorem 11, §6, Chapter II of [47]]} \\
&= 1 - \epsilon^k F_V(\epsilon) - \int_{(\epsilon,1]} F_V(v) k v^{k-1} \, \mathrm{d}v \\
&\leq 1 - \epsilon^k \bar{F}_V(\epsilon) - \int_{(\epsilon,1]} \bar{F}_V(v) k v^{k-1} \, \mathrm{d}v \\
&= \int_{(\epsilon,1]} v^k \, \mathrm{d}\bar{F}_V(v).
\end{aligned}$$

Hence, $\mathbb{P}^N\{\boldsymbol{\delta} \in \Delta_I^N : \, V(x_I^*(\boldsymbol{\delta})) > \epsilon\}$ can finally be bounded as follows:

$$\begin{aligned}
\mathbb{P}^N\{\boldsymbol{\delta} \in \Delta_I^N : \, V(x_I^*(\boldsymbol{\delta})) > \epsilon\} &\leq \int_{(\epsilon,1]} v^k \, \mathrm{d}\bar{F}_V(v) \\
&= \text{[the density of } \bar{F}_V \text{ is } d\binom{N-k}{d} v^{d-1}(1-v)^{N-k-d}\text{]} \\
&= \int_{(\epsilon,1]} v^k \cdot d\binom{N-k}{d} v^{d-1}(1-v)^{N-k-d} \, \mathrm{d}v \\
&= \text{[integration by parts]} \\
&= \frac{d\binom{N-k}{d}}{(k+d)\binom{N}{k+d}} \sum_{i=0}^{k+d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i}. \qquad (14)
\end{aligned}$$

To conclude the proof, go back to (11) and note that $\mathcal{I}$ contains $\binom{N}{k}$ choices. Thus,

$$\begin{aligned}
\mathbb{P}^N\{B\} &\leq \sum_{I \in \mathcal{I}} \mathbb{P}^N\{\boldsymbol{\delta} \in \Delta_I^N : \, V(x_I^*(\boldsymbol{\delta})) > \epsilon\} \\
&= \binom{N}{k} \mathbb{P}^N\{\boldsymbol{\delta} \in \Delta_I^N : \, V(x_I^*(\boldsymbol{\delta})) > \epsilon\} \\
&\leq \text{[use (14)]} \\
&\leq \binom{N}{k} \frac{d\binom{N-k}{d}}{(k+d)\binom{N}{k+d}} \sum_{i=0}^{k+d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i} \\
&= \binom{k+d-1}{k} \sum_{i=0}^{k+d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i} \\
&\leq \beta,
\end{aligned}$$

where the last inequality follows from (3). □

5.2 Proof of (7)

Consider a fully-supported problem P (see Definition 3 in [45]). Equation (14) in the proof of Theorem 2.1 holds true in this case with equality, that is

$$\mathbb{P}^N\{\boldsymbol{\delta} \in \Delta_I^N : \, V(x_I^*(\boldsymbol{\delta})) > \epsilon\} = \frac{d\binom{N-k}{d}}{(k+d)\binom{N}{k+d}} \sum_{i=0}^{k+d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i}, \qquad (15)$$

with $x_I^*(\boldsymbol{\delta})$ and Δ_I^N defined as in (9) and (10).

For a multi-sample $\boldsymbol{\delta}$, let # be the number of solutions of level k (that is solutions that violate exactly k of the sampled constraints) whose violation is more than ϵ. From (9) and (10), one easily sees that

$$\# = \sum_{I\in\mathcal{I}} 1_{\{\boldsymbol{\delta}\in\Delta_I^N:\, V(x_I^*(\boldsymbol{\delta}))>\epsilon\}},$$

where 1_A denotes the indicator function of set A. We now have that

$$\begin{aligned}
\mathrm{E}_{\Delta^N}[\#] &= \int_{\Delta^N} \sum_{I\in\mathcal{I}} 1_{\{\boldsymbol{\delta}\in\Delta_I^N:\, V(x_I^*(\boldsymbol{\delta}))>\epsilon\}}\, \mathbb{P}^N\{\mathrm{d}\boldsymbol{\delta}\} \\
&= \sum_{I\in\mathcal{I}} \mathbb{P}^N\{\boldsymbol{\delta}\in\Delta_I^N :\, V(x_I^*(\boldsymbol{\delta}))>\epsilon\} \\
&= [\text{use (15) and recall that } \mathcal{I} \text{ contains } \binom{N}{k} \text{ choices}] \\
&= \binom{N}{k}\frac{d\binom{N-k}{d}}{(k+d)\binom{N}{k+d}} \sum_{i=0}^{k+d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i} \\
&= \binom{k+d-1}{k}\sum_{i=0}^{k+d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i}. \qquad (16)
\end{aligned}$$

Now, letting $S(\#)$ be the set in Δ^N where $\# \neq 0$, that is the set of multi-samples such that at least one solution of level k violates more than ϵ, the algorithm $\overline{\mathcal{A}}$ that always selects the solution $x_{N,k}^*(\boldsymbol{\delta})$ of level k with the largest violation leads to the fact that $V(x_{N,k}^*(\boldsymbol{\delta})) > \epsilon$ holds on $S(\#)$, that is with a probability $\mathbb{P}^N\{S(\#)\}$. Nobody to date knows the exact value of $\mathbb{P}^N\{S(\#)\}$, but it turns out that we can compute a lower bound to it. In fact, a fully supported problem has $\binom{k+d-1}{k}$ solutions of level k, see e.g. [48], so that $\# \leq \binom{k+d-1}{k}$. Using this fact in (16) yields:

$$\binom{k+d-1}{k}\sum_{i=0}^{k+d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i} = \mathrm{E}_{\Delta^N}[\#] \leq \binom{k+d-1}{k}\cdot\mathbb{P}^N\{S(\#)\},$$

from which $\mathbb{P}^N\{S(\#)\} \geq \sum_{i=0}^{k+d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i}$. Since $\mathbb{P}^N\{S(\#)\}$ is the probability that the solution of algorithm $\overline{\mathcal{A}}$ violates more than ϵ, the found number $\sum_{i=0}^{k+d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i}$ represents a lower bound to $\sup_{\mathrm{P},\mathcal{A}}\mathbb{P}^N\{V(x_{N,k}^*(\boldsymbol{\delta})) > \epsilon\}$. □

6 Optimality results

In this section we establish the result that the objective value of CCP_ϵ can be approached at will, provided that sampled constraints are optimally removed. Though at the present state of knowledge optimal removal can be impractical due to the ensuing high computational burden, this study has a theoretical interest and sheds further light on the relation between chance-constrained optimization and its sample counterpart.

Let $\mathcal{A}_{opt}$ be the optimal constraints removal algorithm which leads – among all possible eliminations of k constraints out of N – to the best possible improvement in the cost objective; further, let $x^*_{N,k\ opt}$ and $J^*_{N,k\ opt} = c^T x^*_{N,k\ opt}$ be the corresponding optimal solution and cost value. We have the following theorem.

Theorem 6.1 *Let $\beta \in (0,1)$ be any small confidence parameter value, and let $\nu \in (0,\epsilon)$ be a performance degradation parameter value. If N and k are such that*

$$\binom{k+d-1}{k} \sum_{i=0}^{k+d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i} + \sum_{i=k+1}^{N} \binom{N}{i} (\epsilon-\nu)^i (1-\epsilon+\nu)^{N-i} \leq \beta, \tag{17}$$

then

(*i*) $V(x^*_{N,k\ opt}) \leq \epsilon$

(*ii*) $J^*_{N,k\ opt} \leq J^*_{\epsilon-\nu}$

simultaneously hold with probability at least $1-\beta$.

As in Theorem 2.1, point (*i*) is a feasibility result. Instead, point (*ii*) states that the performance achieved by $x^*_{N,k\ opt}$ is no worse than the performance of $\text{CCP}_{\epsilon-\nu}$, where ν is a degradation margin. A result similar to (*ii*) has also been independently established in [41]. A simple example illustrating Theorem 6.1 is provided in Appendix A, part A.2.

Proof: Let

$$B_i = \{\boldsymbol{\delta} \in \Delta^N :\ V(x^*_{N,k\ opt}(\boldsymbol{\delta})) > \epsilon\},$$
$$B_{ii} = \{\boldsymbol{\delta} \in \Delta^N :\ J^*_{N,k\ opt}(\boldsymbol{\delta}) > J^*_{\epsilon-\nu}\}.$$

We have to prove that $\mathbb{P}^N\{B_i \cup B_{ii}\} \leq \beta$.

Write $\mathbb{P}^N\{B_i \cup B_{ii}\} \leq \mathbb{P}^N\{B_i\} + \mathbb{P}^N\{B_{ii}\}$. By Theorem 2.1, $\mathbb{P}^N\{B_i\}$ is bounded by $\binom{k+d-1}{k} \sum_{i=0}^{k+d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i}$. We here bound $\mathbb{P}^N\{B_{ii}\}$.

For the sake of simplicity, in what follows we assume that a solution $x^*_{\epsilon-\nu}$ of $\text{CCP}_{\epsilon-\nu}$ exists. If not, the result is similarly established by a limit reasoning.

Let $\overline{\Delta}_{\epsilon-\nu}$ be the subset of Δ formed by those constraints which are violated by $x^*_{\epsilon-\nu}$. Call $R(\boldsymbol{\delta})$ the number of constraints among the N sampled ones that fall in $\overline{\Delta}_{\epsilon-\nu}$. The objective value obtained by eliminating the $R(\boldsymbol{\delta})$ constraints in $\overline{\Delta}_{\epsilon-\nu}$, say $\overline{J}(\boldsymbol{\delta})$, cannot be worse than $J^*_{\epsilon-\nu}$: $\overline{J}(\boldsymbol{\delta}) \leq J^*_{\epsilon-\nu}$. Thus, if $J^*_{N,k\ opt}(\boldsymbol{\delta}) > J^*_{\epsilon-\nu}$,

then $R(\boldsymbol{\delta}) > k$ for, otherwise, $J^*_{N,k\ opt}(\boldsymbol{\delta}) \leq \overline{J}(\boldsymbol{\delta}) \leq J^*_{\epsilon-\nu}$. Therefore,

$$\begin{aligned}
\mathbb{P}^N\{B_{ii}\} &= \mathbb{P}^N\{J^*_{N,k\ opt}(\boldsymbol{\delta}) > J^*_{\epsilon-\nu}\} \\
&\leq \mathbb{P}^N\{R(\boldsymbol{\delta}) > k\} \\
&= [\text{probability that more than } k \text{ among } N \text{ samples fall in } \overline{\Delta}_{\epsilon-\nu}] \\
&= \sum_{i=k+1}^{N} \binom{N}{i} (\mathbb{P}\{\overline{\Delta}_{\epsilon-\nu}\})^i (1 - \mathbb{P}\{\overline{\Delta}_{\epsilon-\nu}\})^{N-i} \\
&\leq \sum_{i=k+1}^{N} \binom{N}{i} (\epsilon-\nu)^i (1-\epsilon+\nu)^{N-i},
\end{aligned}$$

where the last inequality follows from observing that $\mathbb{P}\{\overline{\Delta}_{\epsilon-\nu}\} \leq \epsilon - \nu$.
Wrapping up the above results, we finally have

$$\begin{aligned}
\mathbb{P}^N\{B_i \cup B_{ii}\} &\leq \mathbb{P}^N\{B_i\} + \mathbb{P}^N\{B_{ii}\} \\
&\leq \binom{k+d-1}{k} \sum_{i=0}^{k+d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i} + \sum_{i=k+1}^{N} \binom{N}{i} (\epsilon-\nu)^i (1-\epsilon+\nu)^{N-i} \\
&\leq \beta,
\end{aligned}$$

where the last inequality is given by (17). □

6.1 Existence of N and k

In Theorem 6.1, ν measures the performance mismatch between $x^*_{N,k\ opt}$ and the solution of CCP_ϵ. For any small ν, N and k satisfying condition (17) always exist, a result shown in this section. As expected, N and k increase as ν approaches zero.

To start with, consider (17) and split β evenly between the two terms in the left-hand-side of this equation, that is impose that both terms are less than $\beta/2$. A condition for the first term to be less than $\beta/2$ has been already established in equation (8) (substitute $\beta/2$ for β in that equation). We here work on the second term.
Similarly to (17), we use a Chernoff bound for the Binomial tail, this time the right tail Chernoff bound (see e.g. Section 2.3.1 in [46]) stating that

$$\sum_{i=k+1}^{N} \binom{N}{i} (\epsilon-\nu)^i (1-\epsilon+\nu)^{N-i} \leq e^{-\frac{((\epsilon-\nu)N-k-1)^2}{3(\epsilon-\nu)N}}, \quad \text{for } (k+1)/2 \leq (\epsilon-\nu)N \leq k+1.$$

Further, imposing that $e^{-\frac{((\epsilon-\nu)N-k-1)^2}{3(\epsilon-\nu)N}}$ is less than $\beta/2$ and solving for k yields

$$k \geq (\epsilon-\nu)N - 1 + \sqrt{3(\epsilon-\nu)N \ln \frac{2}{\beta}}. \tag{18}$$

In (8), the term linear in N has slope ϵ, while the other terms are sub-linear. Instead, in (18) the slope is $(\epsilon - \nu)$. Since $\epsilon > (\epsilon - \nu)$, for N large enough there is a gap between the bounds expressed by (8) and (18), and, consequently, an N and a k can be found that simultaneously satisfy (8) and (18).

7 Conclusions

In this paper, we have presented results linking a chance-constrained optimization problem to its sample counterpart, and have further developed a general paradigm to solve chance-constrained problems in practice. The introduced method is grounded on a solid and deep theory, but its practical use is very simple and consists of sampling and discarding of the uncertainty. The potential applications domain is truly vast and includes problems in control, system identification and learning, signal processing, and finance.

A Appendix: illustration of the theoretical results via a simple example

The following simple 1-dimensional example illustrates the theoretical nature of the results contained in this paper.

Let us consider the chance-constrained problem:

$$\min_{x \in [0,1]} x$$

$$\text{s.t. } \mathbb{P}\{\delta : \ x \geq \delta\} \geq 1 - \epsilon,$$

where $\delta \in \Delta = [0, 1]$ and $\mathbb{P}$ is uniform over Δ.
In this simple setting, the CCP_ϵ optimum is achieved by removing the set $[1 - \epsilon, 1]$ from Δ, leading to the optimal solution $x^*_\epsilon = 1 - \epsilon = J^*_\epsilon$. Throughout we take $\epsilon = 0.2$.
Turn now to consider the sample-based optimization program where k constraints are removed. Given a multi-sample $(\delta^{(1)}, \ldots, \delta^{(N)})$, $x^*_{N,k}$ is obtained by removing the k largest $\delta^{(i)}$'s and by letting $x^*_{N,k} =$ the $(k+1)^{\text{th}}$ largest $\delta^{(i)}$ value. Also, $\mathbb{P}\{\delta : \ x^*_{N,k} < \delta\} = 1 - x^*_{N,k}$ and $J^*_{N,k} = x^*_{N,k}$, and all these quantities are random variables as they depend on the multi-sample $(\delta^{(1)}, \ldots, \delta^{(N)})$. To ease the notation, let in the following $V(x^*_{N,k}) = \mathbb{P}\{\delta : \ x^*_{N,k} < \delta\}$.

A.1 The need for constraints removal

Figure 7 depicts the probability density function of $V(x^*_{N,k})$ when $N = 15$ and $k = 0$, that is, no constraints are removed. One sees that $V(x^*_{N,k}) \leq \epsilon = 0.2$ with high probability. On the other hand, the density

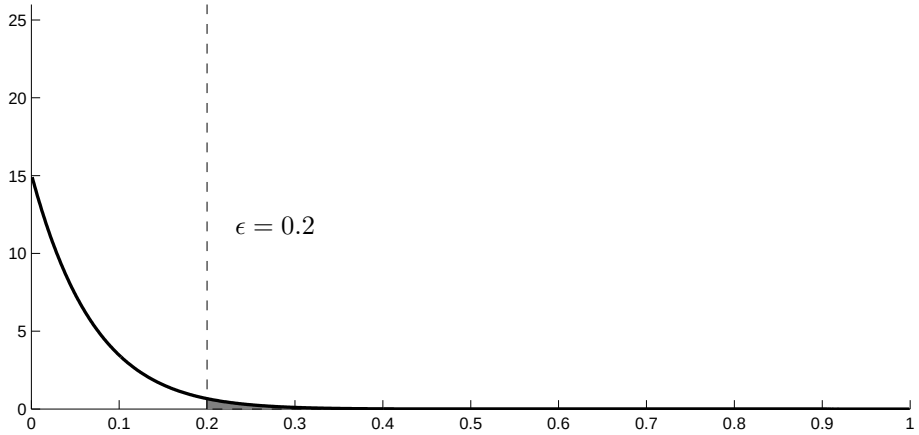

Figure 7: The probability density function of $V(x^*_{N,k})$ for $N = 15$ and $k = 0$; grey area represents the probability that $V(x^*_{N,k}) > \epsilon$.

concentrates near the zero value. This means that the violation of $x^*_{N,k}$ will be much less than that for x^*_ϵ with high probability, entailing that the objective value of $x^*_{N,k}$ will be poor as compared to the chance-constrained solution.

Selecting $N = 552$ and $k = 93$ leads instead to the probability density function in Figure 8. Constraints

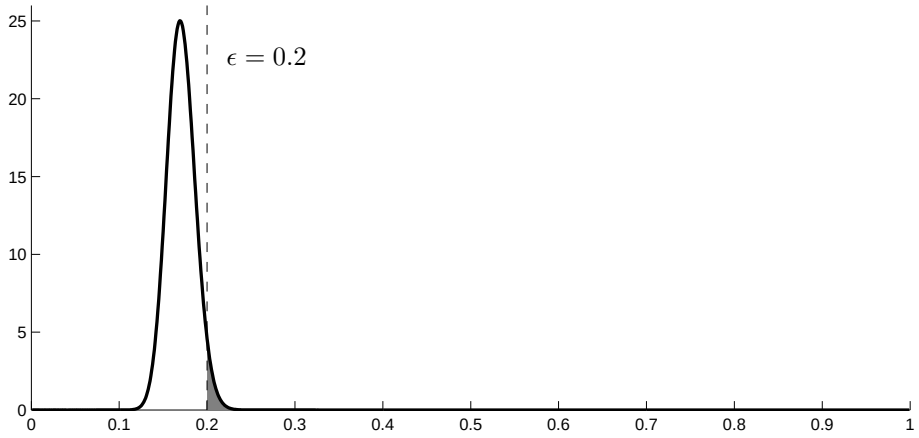

Figure 8: The probability density function of $V(x^*_{N,k})$ for $N = 552$ and $k = 93$; grey area represents the probability that $V(x^*_{N,k}) > \epsilon$.

discarding generated solutions for which $V(x^*_{N,k}) > \epsilon$ has the same probability as for $N = 15$ and $k = 0$, but the violation approaches the desired violation level $\epsilon = 0.2$ of the chance-constrained problem with high

probability. This is the beneficial effect of constraints removal.

A.2 Optimality results

In this subsection we illustrates the results in the optimality Theorem 6.1.

Observe first that, in this 1-dimensional example, the considered removal algorithm coincides with the optimal removal algorithm $\mathcal{A}_{opt}$, i.e. $x^*_{N,k} = x^*_{N,k\ opt}$.

Again referring to $N = 552$ and $k = 93$, Figure 9 further displays the region B_i where $V(^*_{N,k\ opt}) > \epsilon$ along with region B_{ii} where $J^*_{N,k\ opt} = 1 - V(x^*_{N,k\ opt}) > 1 - (\epsilon - \nu) = J^*_{\epsilon-\nu}$ for $\nu = 0.05$. Here, $\mathbb{P}^N\{B_i \cup B_{ii}\} =$

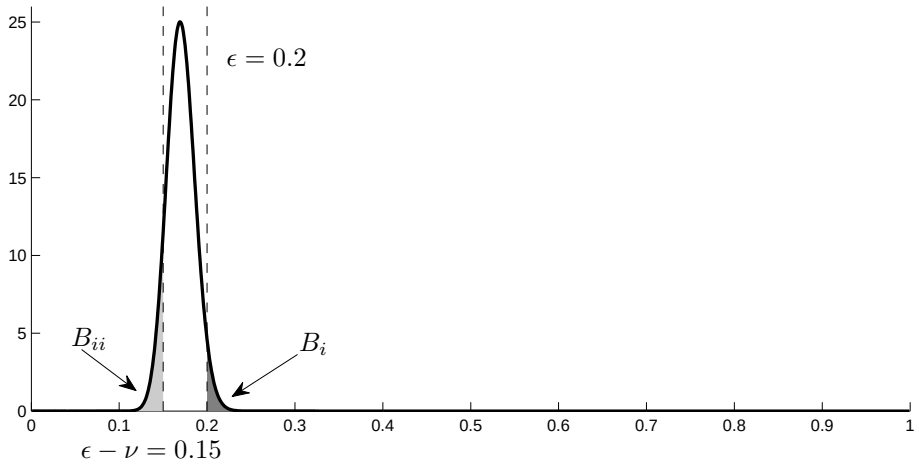

Figure 9: The probability density function of $V(x^*_{N,k\ opt})$ for $N = 552$ and $k = 93$; B_i is the region where $V(x^*_{N,k\ opt}) > \epsilon$, while B_{ii} where $J^*_{N,k\ opt} > J^*_{\epsilon-\nu}$.

0.1352.

Thus, $N = 552$ and $k = 93$ suffice to simultaneously guarantee that $V(x^*_{N,k\ opt}) \leq 0.2$ and $J^*_{N,k\ opt} \leq J^*_{0.15}$ with probability 0.8648. Interestingly enough, applying Theorem 6.1 provides in general upper-bounds for N and k; however, in the present 1-dimensional case, substituting $\epsilon = 0.2$, $\nu = 0.05$, and $\beta = 1-0.8648 = 0.1352$ in (17) just returns $N = 552$ and $k = 93$.

References

[1] A. Charnes, W.W. Cooper, and G.H. Symonds. Cost horizons and certainty equivalents: an approach to stochastic programming of heating oil. *Management Science*, 4:235–263, 1958.

[2] L.B. Miller and H. Wagner. Chance-constrained programming with joint constraints. *Operations Research*, 13:930–945, 1965.

[3] A. Prèkopa. On probabilistic constrained programming. In *Proceedings of the Princeton Symposium on Mathematical Programming*, Princeton, USA, 1970. Princeton University Press.

[4] A. Prèkopa. Contributions to the theory of stochastic programming. *Mathematical programming*, 4:202–221, 1973.

[5] A. Prèkopa. *Stochastic programming*. Kluwer, Boston, MT, USA, 1995.

[6] A. Prèkopa. Probabilistic programming. In A. Ruszczyǹski and A. Shapiro, editors, *Stochastic Programming*, volume 10 of *handbooks in operations research and management science*, London, UK, 2003. Elsevier.

[7] D. Dentcheva. Optimization models with probabilistic constraints. In G. Calafiore and F. Dabbene, editors, *Probabilistic and Randomized Methods for Design under Uncertainty*, London, 2006. Springer-Verlag.

[8] R. Henrion and C. Strugarek. Convexity of chance constraints with independent random variables. *Computational Optimization and Applications*, 41:263–276, 2008.

[9] A. Ben-Tal and A. Nemirovski. Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming*, 88:411–424, 2000.

[10] A. Ben-Tal and A. Nemirovski. Robust optimization – methodology and applications. *Mathematical Programming Series B*, 92:453–480, 2002.

[11] D. Bertsimas and M. Sim. The price of robustness. *Operation Research*, 52(1):35–53, 2004.

[12] A. Nemirovski and A. Shapiro. Convex approximations of chance constrained programs. *SIAM Journal on Optimization*, 17:969–996, 2006.

[13] R. Henrion and W. Römisch. Hölder and lipschitz stability of solution sets in programs with probabilistic constraints. *Mathematical Programming*, 100:589–611, 2004.

[14] R. Henrion and W. Römisch. Metric regularity and quantitative stability in stochastic programs with probabilistic constraints. *Mathematical Programming*, 84:55–88, 1999.

[15] A. Prèkopa. Sharp bound on probabilities using linear programming. *Operations Research*, 38:227–239, 1990.

[16] S. Sen. Relaxations for the probabilistically constrained programs with discrete random variables. *Operation Research Letters*, 11:81–86, 1992.

[17] D. Dentcheva, A. Prèkopa, and A. Ruszczyǹski. Concavity and efficient points of discrete distributions in probabilistic programming. *Mathematical Programming*, 89:55–77, 2000.

[18] D. Dentcheva, A. Prèkopa, and A. Ruszczyǹski. On convex probabilistic programming with discrete distribution. *Nonlinear Analysis*, 47:1997–2009, 2001.

[19] P. Beraldi and A. Ruszczyǹski. A branch and bound method for stochastic integer problems under probabilistic constraints. *Optimization Methods and Software*, 17:359–382, 2002.

[20] D. Dentcheva, B. Lai, and A. Ruszczyǹski. Dual methods for probabilistic optimization problems. *Mathematical Methods of Operations Research (ZOR)*, 60:331–346, 2004.

[21] A. Prèkopa. Dual method for a one-stage stochastic programming problem with random RHS obeying a discrete probability distribution. *Z. Oper. Res.*, 34:441–461, 1990.

[22] A. Ruszczyǹski. Probabilistic programming with discrete distributions and precedence constrained knapsack polyhedra. *Mathematical Programming*, 93(2):195–215, 2002.

[23] J. Luedtke, S. Ahmed, and G.L. Nemhauser. An integer programming approach for linear programs with probabilistic constraints. *Mathematical Programming*, 122:247–272, 2010.

[24] S. Vogel. A stochastic approach to stability in stochastic programming. *Journal of Computational and Applied Mathematics*, 56(1-2):65–96, 1994.

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

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

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

[28] A. Shapiro. Monte-Carlo sampling methods. In A. Ruszczyǹski and A. Shapiro, editors, *Stochastic Programming*, volume 10 of *handbooks in operations research and management science*, London, UK, 2003. Elsevier.

[29] G.C. Pflug. Stochastic optimization and statistical inference. In A. Ruszczyǹski and A. Shapiro, editors, *Stochastic Programming*, volume 10 of *handbooks in operations research and management science*, London, UK, 2003. Elsevier.

[30] D.P. de Farias and B. Van Roy. On constraints sampling in the linear programming approach to approximate dynamic programming. *Mathematics of Operations Research*, 29(3):462–478, 2004.

[31] E. Erdoğan and G. Iyengar. Ambiguous chance constrained problems and robust optimization. *Mathematical Programming Series B*, 107(1-2):37–61, 2006.

[32] A. Nemirovski and A. Shapiro. Scenario approximations of chance constraints. In G. Calafiore and F. Dabbene, editors, *Probabilistic and Randomized Methods for Design under Uncertainty*, London, 2006. Springer-Verlag.

[33] B.T. Polyak and P.S. Scherbakov. Randomized methods for solving semidefinite programming. In O.N.Granichin, editor, *Stochastic Optimization in Informatics*, volume 10, pages 38–70, London, UK, 2006. St.-Petersburg Univ.

[34] A. Shapiro, D. Dentcheva, and A. Ruszczyǹski. *Lectures on stochastic programming - Modeling and theory*. MPS-SIAM, 2009.

[35] T. Alamo, R. Tempo, and E.F. Camacho. Statistical learning theory: a pack-based strategy for uncertain feasibility and optimization problems. In V.D. Blondel, S.P. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, volume 10, London, UK, 2008. Springer- Verlag.

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

[37] Y. Fujisaki, F. Dabbene, and R. Tempo. Probabilistic robust design of lpv control systems. *Automatica*, 39:1323–1337, 2003.

[38] H. Ishii, T. Basar, and R. Tempo. Randomized algorithms for quadratic stability of quantized sampled-data systems. *Automatica*, 40:839–846, 2004.

[39] T. Alpcan, T. Basar, and R. Tempo. Randomized algorithms for stability and robustness of high-speed communication networks. *IEEE Transaction on Neural Networks*, 16:1229–1241, 2005.

[40] R. Tempo, G. Calafiore, and F. Dabbene. *Randomized algorithms for analysis and control of uncertain systems*. Springer-Verlag, London, UK, 2005.

[41] J. Luedtke and S. Ahmed. A sample approximation approach for optimization with probabilistic constraints. *SIAM Journal on Optimization*, 19(2):674–699, 2008.

[42] T. Alamo, R. Tempo, and E.F. Camacho. Randomized strategies for probabilistic solutions of uncertain feasibility and optimization problems. *IEEE Transactions on Automatic Control*, 54:2545–2559, 2009.

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

[44] G. Calafiore and M.C. Campi. The scenario approach to robust control design. *IEEE Transaction on Automatic Control*, 51(5):742–753, 2006.

[45] M.C. Campi and S. Garatti. The exact feasibility of randomized solutions of uncertain convex programs. *SIAM Journal on Optimization*, 19(3):1211–1230, 2008.

[46] M. Vidyasagar. *A theory of learning and generalization: with applications to neural networks and control systems*. Springer-Verlag, London, UK, 1997.

[47] A.N. Shiryaev. *Probability*. Springer, New York, NY, USA, 1996.

[48] J. Matoušek. On geometric optimization with few violated constraints. *Discrete & Computational Geometry*, 14:365–384, 1995.