Wait-and-judge scenario optimization

M. C. Campi[1] · S. Garatti[2]

Abstract We consider convex optimization problems with uncertain, probabilistically described, constraints. In this context, scenario optimization is a well recognized methodology where a sample of the constraints is used to describe uncertainty. One says that the scenario solution generalizes well, or has a high robustness level, if it sat-isfies most of the other constraints besides those in the sample. Over the past 10 years, the main theoretical investigations on the scenario approach have related the robust-ness level of the scenario solution to the number of optimization variables. This paper breaks into the new paradigm that the robustness level is a-posteriori evaluated after the solution is computed and the actual number of the so-called support constraints is assessed (wait-and-judge). A new theory is presented which shows that a-posteriori observing k support constraints in dimension $d > k$ allows one to draw conclusions close to those obtainable when the problem is from the outset in dimension k. This new theory provides evaluations of the robustness that largely outperform those carried out based on the number of optimization variables.

Keywords Sample-based optimization · Scenario approach · Stochastic optimization · Probabilistic constraints

Received: 15 September 2015
Accepted: 13 July 2016
ublished online: 26 July 2016

✉ M. C. Campi
marco.campi@unibs.it
http://bsing.ing.unibs.it/∼campi/

S. Garatti
simone.garatti@polimi.it
http://home.deib.polimi.it/sgaratti/

[1] Dipartimento di Ingegneria dell'Informazione, Università di Brescia, Via Branze 38, 25123 Brescia, Italy

[2] Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, piazza Leonardo da Vinci 32, 20133 Milan, Italy

This is a post-peer-review, pre-copyedit version of an article published in Mathematical Programming. The final authenticated version is available online at: https://doi.org/10.1007/s10107-016-1056-9

Mathematics Subject Classification 90C15 · 90C25

1 Introduction

Given a linear cost function $c^T x$, where $x \in \mathcal{X} \subseteq \mathbb{R}^d$ is the optimization variable and $\mathcal{X}$ is a convex set, consider a family of convex constraints $x \in \mathcal{X}_\delta \subseteq \mathbb{R}^d$ parameterized by δ, where δ is a random outcome from a probability space $(\Delta, \mathcal{F}, \mathbb{P})$. Probability $\mathbb{P}$ is not known, but one is given a sample $\delta^{(i)}$, $i = 1, \ldots, N$, with $N > d$, of independent and identically distributed (i.i.d.) realizations from $(\Delta, \mathcal{F}, \mathbb{P})$ and constructs the scenario optimization program

$$\begin{aligned} \mathrm{SP}_N : & \min_{x \in \mathcal{X}} c^T x \\ & \text{subject to: } x \in \bigcap_{i=1,\ldots,N} \mathcal{X}_{\delta^{(i)}}, \end{aligned} \tag{1}$$

In (1), $\delta^{(1)}, \delta^{(2)}, \ldots, \delta^{(N)}$ are called "scenarios", and they have to be interpreted as observations from which one wants to make a decision, i.e., select a value of x. In other words, SP_N is used to make data-based optimization [5,18,49].

To make (1) more concrete, one can e.g. think of an example where one wants to split a given amount of money among q different assets so as to minimize a loss function. Assuming that the loss function is written as $\sum_{j=1}^q \ell_j p_j$, where ℓ_j is a random loss associated to asset j and p_j is the percentage of capital allocated to asset j, and that one has observed the losses $\ell_j^{(i)}$ over N periods $i = 1, \ldots, N$, one may opt to hedge against the worst among the observed periods and set out to split the capital according to program

$$\min_{p_j \geq 0,\ \sum_{j=1}^q p_j = 1} \ \max_{i=1,\ldots,N} \sum_{j=1}^q \ell_j^{(i)} p_j.$$

This program can be written as in (1) by an epigraphic reformulation:

$$\begin{aligned} & \min_{p_j \geq 0,\ \sum_{j=1}^q p_j = 1,\ L} \quad L \\ & \text{subject to: } L \geq \sum_{j=1}^q \ell_j^{(i)} p_j,\ \ i = 1, \ldots, N. \end{aligned}$$

Besides the example above, the setup of (1) accommodates many other problems in fields ranging from machine learning and prediction [9,10,17,35], to quantitative finance [25,40–42], from management [21], to control design [14]. We do not dwell on describing these application domains here, and refer the reader to the above references to this purpose.

Scenario optimization has been introduced in [7], and has ever since attracted an increasing interest. The SAA approach with $\gamma = 0$ of [40] coincides with (1). Robustness properties have been studied in [8,12,16] and, under regularization and structural assumptions, further investigated in [2,11,45,53]. Papers [13,22] consider constraints removal, and [50] examines multi-stage problems. Generalizations to a non-convex setup are proposed in [1,20,23]. See also [8,14,34,36,43,46,48] for a comparison of scenario optimization with other methods in stochastic optimization.

Since (1) only includes the constraints associated to N scenarios, it is a standard convex program that can be numerically solved via common optimization software such as CVX [24], or YALMIP [29], which are handy interfaces for various solvers. Note that a linear cost function $c^T x$ in (1), rather than a more general convex cost function $f(x)$, comes to no loss of generality. In fact, should the cost be $f(x)$, one could reformulate the program in epigraphic form, that is, a new variable y is introduced and this variable is minimized (so that the cost function becomes linear) under the additional constraint $y \geq f(x)$.

Throughout the paper we assume that program (1) admits solution. A classical condition for the existence of a solution is that the intersections between the feasibility domain and the level sets of (1) are compact, see [6,30]. If more than one solution exists, we assume that a solution is singled out by means of a convex rule, that is, the tie is broken by minimizing an additional convex function $t_1(x)$, and, possibly, other convex functions $t_2(x), t_3(x), \ldots$ if the tie still occurs; this is the same approach as Rule 1 in [7]. An example of a tie-break function is the norm of x, $t_1(x) = \|x\|$. Another example is the lexicographic rule, which consists in minimizing the components of x in succession, i.e. $t_1(x) = x_1, t_2(x) = x_2, \ldots$. After breaking the tie, the unique solution is denoted by x_N^*.

1.1 The robustness quantification issue

Upon solving program (1), the solution x_N^* becomes available to the user, and the corresponding cost $c^T x_N^*$ can be calculated. On the other hand, the decision as to whether x_N^* is adopted and implemented in practice also depends on the level of constraint satisfaction warranted by the solution. Depending on the application at hand, constraint satisfaction means e.g. that the prediction is correct, or that the estimated return level in a financial investment is met, or that the designed system, or controller, satisfies the indicated specifications. Enforcing the scenario constraints $x \in \mathcal{X}_{\delta^{(i)}}, i = 1, \ldots, N$, in (1) aims at finding a solution which is "robust" against constraint violation, and one hopes that the solution to program (1) satisfies the constraints associated to most δ's beyond those corresponding to the scenarios. To formalize this concept, introduce the following definition.

Definition 1 (*Violation*) The *violation* of a given $x \in \mathcal{X}$ is defined as

$$V(x) = \mathbb{P}\{\delta \in \Delta : x \notin \mathcal{X}_\delta\}.$$

$V(x)$ quantifies the probability with which a new randomly selected constraint $\mathcal{X}_\delta$ is violated by x. If $V(x) \leq \epsilon$, then x is robust against constraint violation at level ϵ. In general, however, $V(x)$ is not directly computable since $\mathbb{P}$ is not known. In this paper, we are interested in evaluating the violation $V(x_N^*)$ of the solution x_N^* to program

(1). Often times, the observations $\delta^{(i)}$'s are a costly and limited resource, and the assessment of $V(x_N^*)$ is better done without resorting to new observations besides the $\delta^{(1)}, \ldots, \delta^{(N)}$ used in (1). In this setup, the studies in [7,8,12] have pioneered a theory that links the sample size N to $V(x_N^*)$. We here revise the tightest among these results, which has been proven in [12], as we shall have to compare it with the findings of this paper.

$V(x_N^*)$ is a random variable[1] because of the dependence of x_N^* on $\delta^{(1)}, \ldots, \delta^{(N)}$ and the main result of [12] states that $V(x_N^*)$ is bigger than ϵ with a probability upper-bounded by a Beta distribution according to the following formula (in the formula, $\mathbb{P}^N$ refers to the sample $\delta^{(1)}, \ldots, \delta^{(N)}$, which determines x_N^*; $\mathbb{P}^N$ is a product probability due to independence of $\delta^{(1)}, \ldots, \delta^{(N)}$):

$$\mathbb{P}^N\{V(x_N^*) > \epsilon\} \le \sum_{i=0}^{d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i}. \tag{2}$$

This result provides a quantitative tool to evaluate the confidence with which x_N^* is robust at level ϵ, and it represents a fundamental breakthrough in the theoretical study of scenario optimization.[2] Importantly, (2) is valid for any program of the form (1), that is, it holds irrespective of all elements, c, $\mathcal{X}$, $\{\mathcal{X}_\delta\}$, $\mathbb{P}$ which define (1). For short, in the sequel the quadruple $(c, \mathcal{X}, \{\mathcal{X}_\delta\}, \mathbb{P})$ will be referred to as a "problem".

Looking into the derivation of (2) given in [12], one sees that a crucial concept is that of support constraint.

Definition 2 (*Support constraint*) A constraint of the scenario program (1) is a *support constraint* if its removal changes the solution x_N^*. □

In paper [7], it is shown that SP_N cannot have more than d support constraints, that is, the number of support constraints is no more than the number of optimization variables. Correspondingly, in [12], a problem $(c, \mathcal{X}, \{\mathcal{X}_\delta\}, \mathbb{P})$ is called fully-supported if, for any $N > d$, the scenario program (1) has d support constraints with probability 1. En route towards (2), in [12] it is first shown that (2) holds with equality for fully-supported problems, yielding

$$\mathbb{P}^N\{V(x_N^*) > \epsilon\} = \sum_{i=0}^{d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i}, \tag{3}$$

and then it is proven that any other problem is "dominated" by fully-supported problems, which gives (2).

[1] Measurability of $V(x_N^*)$, as well as of other quantities, is taken for granted in this paper.

[2] An alternative way to express that x_N^* is robust at level ϵ is that x_N^* is chance-constrained feasible at level ϵ [19,44,46]. See [4,31,32,37–39,54] for more discussion on chance-constrained optimization and its connection with scenario optimization.

1.2 The wait-and-judge perspective of this paper

Often, optimization problems encountered in applications are not fully-supported. In fact, in scenario programs with many variables it is not rare that way fewer support constraints are found than there are optimization variables, see e.g. [15,45,51,52] and the example in Sect. 2 below. When a problem is not fully-supported, one can object against applying the bound in (2), which is tight only for fully-supported problems according to (3). Hence, one wishes to incorporate in the theory that less than d support constraints have been seen.

In [12], ϵ is a deterministic constant, set in advance prior to seeing any $\delta^{(i)}$. The new perspective introduced in this paper is that ϵ is a function of the number of support constraints that have been found in the instance of the scenario program (1) at hand.[3] To this aim, we let $\epsilon(k)$ be a function that takes values in [0, 1], where k is an integer ranging over $\{0, 1, \ldots, d\}$. After computing the solution x_N^*, one also evaluates the number of support constraints s_N^* of the scenario program (1), and makes the statement that $V(x_N^*) \leq \epsilon(s_N^*)$. In this way, the bound on the violation is a-posteriori determined and it is adjusted to the number of support constraints.

$V(x_N^*)$ is a random variable, and so is $\epsilon(s_N^*)$, because of the dependence of x_N^* and s_N^* on $\delta^{(1)}, \ldots, \delta^{(N)}$. The probability space on which $V(x_N^*)$ and $\epsilon(s_N^*)$ are defined is $(\Delta^N, \mathcal{D}^N, \mathbb{P}^N)$. This paper establishes that for any problem $(c, \mathcal{X}, \{\mathcal{X}_\delta\}, \mathbb{P})$ it holds that

$$\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} \leq \gamma^*, \tag{4}$$

where γ^* depends on N and function $\epsilon(k)$ (Theorem 1). Notice that the left-hand side of (4) can also be written as

$$\mathbb{P}^N\left(\bigcup_{k=0}^{d}\left\{V(x_N^*) > \epsilon(k) \text{ and } s_N^* = k\right\}\right), \tag{5}$$

so that γ^* bounds the probability of seeing k support constraints, where k can be any number in $\{0, 1, \ldots, d\}$, and then a wrong statement that $V(x_N^*) \leq \epsilon(k)$ is made. One is interested in making γ^* very small, for example $\gamma^* = 10^{-6}$, which means that $V(x_N^*) \leq \epsilon(s_N^*)$ holds with very high confidence $1 - \gamma^*$. When $\epsilon(k)$ is chosen to be constant, $\epsilon(k) = \epsilon$ for all k, the γ^* in (4) turns out to be the same as the right-hand side of (2), and result (2) is recovered as a particular case of result (4) of this paper (Corollary 1). On the other hand, we show that the function $\epsilon(k)$ can be selected so that it largely improves over the constant ϵ, i.e., $\epsilon(k)$ is significantly smaller than ϵ for most values of k when γ^* is equal to the right-hand side of (2). Hence, the flexibility introduced by the new theory permits one to formulate much stronger conclusions on the violation $V(x_N^*)$ than the result in [12].

Interestingly, establishing (4) requires a genuinely new line of work and a simplistic approach where Eq. (3) with k in place of d is used to bound (5) leads to a wrong conclusion. In more specific terms, after rewriting (5) as

[3] Computing the number of support constraints requires removing one by one the active constraints and verifying whether the solution changes, an operation that can be carried out at reasonably low computational cost.

$$\sum_{k=0}^{d} \mathbb{P}^N \left\{ V(x_N^*) > \epsilon(k) \text{ and } s_N^* = k \right\}, \tag{6}$$

if each probability in (6) is bounded by $\sum_{i=0}^{k-1} \binom{N}{i} \epsilon(k)^i (1-\epsilon(k))^{N-i}$ (which is obtained from (3) with k in place of d and $\epsilon(k)$ in place of ϵ), then one obtains

$$\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} \leq \sum_{k=0}^{d} \sum_{i=0}^{k-1} \binom{N}{i} \epsilon(k)^i (1-\epsilon(k))^{N-i}. \tag{7}$$

However, as shown in Appendix 1, Eq. (7) is incorrect. This is the sign of a deep fact: a-posteriori observing k in dimension d is not the same as working in dimension k, or, said differently, simple solutions (supported by k constraints) to complex problems (in dimension $d > k$) are not as guaranteed as solutions to simple problems (in dimension k).

Although observing k support constraints is not the same as working in dimension k, this paper establishes the fundamental fact that the gap between the two is quantitatively minor: the function $\epsilon(k)$ can be chosen so that it is close for all k to the result that is valid in dimension k. In other words, a-posteriori checking the number of support constraints does lead to conclusions quantitatively similar to those achievable when the optimization problem has as many optimization variables as there are support constraints in the program (1) at hand. Remarkably, this result holds true distribution-free, and it is applicable without any knowledge on the probability $\mathbb{P}$.

1.3 Structure of the paper

The example in the next Sect. 2 gives a quantitative preview of the findings of this paper. The main result is presented in Sect. 3, Sect. 4 discusses the interpretation and use of the main result, and Sect. 5 contains the proofs of all results given in this first part of the paper. Section 6 takes a broader perspective and the theory developed in the first five sections is generalized to optimization problems in infinite-dimensional/general spaces. The proofs of the results of this section can be found in Sect. 7. Section 8 closes the paper with a critical overview of the presented theory.

2 A preview example

$N = 1000$ points $p^{(i)} \in \mathbb{R}^{100}$ are independently sampled from an unknown probability density, and the hyper-sphere S of smallest volume that contains all the points is constructed by resolving the following program where q is the center of S and r is its radius:

$$\min_{q \in \mathbb{R}^{100}, r \in \mathbb{R}} \quad r$$
$$\text{subject to: } \|p^{(i)} - q\| \leq r, \quad i = 1, \ldots, N. \tag{8}$$

We want to provide estimates on the probabilistic mass contained in the hyper-sphere S, or, which is the same, on the probability that one next point sampled independently of the initial set of 1000 points falls in S.

In this problem, we identify a point p in $\mathbb{R}^{100}$ with the uncertainty parameter δ, the $p^{(i)}$'s are the scenarios $\delta^{(i)}$'s, and (8) is a scenario program of the type (1). A new point p falls outside the hyper-sphere constructed by resolving (8) if $\|p - q^*_{1000}\| > r^*_{1000}$, where (q^*_{1000}, r^*_{1000}) is the solution of (8). The probability for this to happen is the violation $V(q^*_{1000}, r^*_{1000})$ of the solution (q^*_{1000}, r^*_{1000}).

The optimization variables are the radius r and the 100 coordinates defining the center q, which yields $d = 101$ and $N = 1000$. With these values, an application of Theorem 2 in Sect. 4 gives that $V(q^*_{1000}, r^*_{1000}) \leq \epsilon(s^*_{1000})$ holds with high confidence $1 - 10^{-6}$ with the function $\epsilon(k)$ that is profiled in Fig. 1. Upon resolving program (8), we found 28 support constraints. This is the number of points $p^{(i)}$ that are on the surface of S. Since $\epsilon(28) = 6.71\,\%$, we conclude that the probabilistic volume outside the hyper-sphere does not exceed 6.71 %.
Some remarks are in order.

(i) Figure 1 profiles $\epsilon(k)$ as given by Theorem 2 when $N = 1000$, $d = 101$, and $\beta = 10^{-6}$. Using instead Eq. (2) with the same values for N, d, and β, one finds $\epsilon = 15.17\,\%$, so that resorting to the theory of [12] a weaker conclusion by a factor more than 2 is drawn. For easy reference, ϵ is also represented in Fig. 1. One sees that Theorem 2 improves over the result in [12] for most values of k. The fewer the support constraints, the larger the improvement. Notice that improving for all values of k is impossible due to fundamental theoretical reasons explained in Sect. 4.

(ii) The use of Theorem 2 does not require any knowledge of the distribution according to which the points in $\mathbb{R}^{100}$ are sampled. The distribution-free nature of Theorem 2 makes it perfectly suitable for observation-based problems.

(iii) In our example, which is by simulation, some post-experiment analysis is possible because we actually generated the points and their distribution is therefore known. This analysis highlights some important features of the method. The points were generated from a Gaussian distribution with zero mean and identity covariance matrix. The probabilistic mass outside the hyper-sphere found by solving (8) was

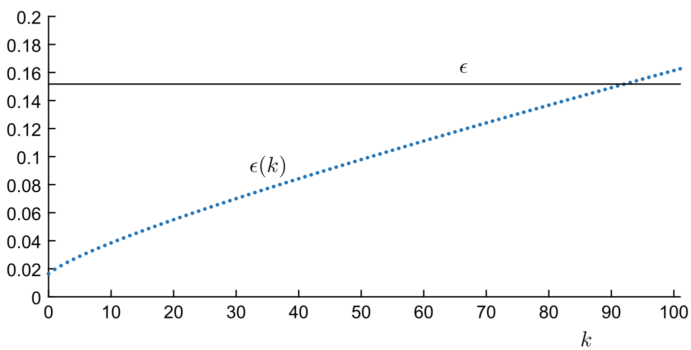

Fig. 1 $\epsilon(k)$ and ϵ for the preview example

3.67 %, below the value $\epsilon(28) = 6.71\,\%$. We then performed a repetition of 500 trials of the same experiment and the number of support constraints was always between 20 and 43. All the times the probabilistic mass was below the value $\epsilon(s_N^*)$ as it is expected since the confidence is $1 - 10^{-6}$. On average, the probabilistic mass outside the sphere was in a ratio of 0.44 with $\epsilon(s_N^*)$. A margin between the real mass and $\epsilon(s_N^*)$ is required because the mass outside the hyper-sphere is subject to stochastic fluctuation.

(iv) In the simulation run of the example, we found 28 support constraints. Given any fully-supported problem in $d = 28$ dimensions, it is not difficult to augment the optimization domain with 73 dummy variables to make the total number of optimization variables equal to 101, which is the same as the number of variables we have in this example, while the number of support constraints remains 28. For these problems, one can show that result (3) can be applied with $d = 28$. Therefore, any result that is valid distribution-free for all problems with 101 variables cannot possibly return a value for $\epsilon(28)$ that outdoes the ϵ obtained from (3) with $d = 28$. Interestingly, setting the right-hand side of (3) to 10^{-6} gives $\epsilon = 5.97\,\%$, a value not too different from $\epsilon(28) = 6.71\,\%$ obtained with Theorem 2. This result is interpreted that a-posteriori discovering that there are 28 support constraints leads to certificates on the violation that are not too different from a-priori knowing that the support constraints are always deterministically equal to 28.

(v) The result of this section is relevant to the theory of tolerance regions, [26,28], which we briefly recall here. A well studied problem in statistics is that of estimating a cumulative distribution function from data. This concerns with evaluating the probability of sets having specific shapes, like half-lines or quadrants. When more general shapes are considered, one speaks of "tolerance regions", and the results in our previous contribution [12] can be applied to this context. The example of this section shows the potentials of the new theory of this paper to obtain results for tolerance regions that are tighter than those obtainable from [12].

3 Main result

It turns out that studying the properties of the solution of the convex scenario optimization program SP_N in (1), which has N scenarios, calls for the consideration of other programs with the same structure as SP_N but with a set of constraints whose cardinality ranges over all integers $m = 0, 1, 2, \ldots$. Accordingly, we consider replicas of (1) with m constraints as follows:

$$\min_{x \in \mathcal{X}} c^T x, \tag{9a}$$

when $m = 0$, and

$$\begin{aligned} &\min_{x \in \mathcal{X}} c^T x \\ &\text{subject to: } x \in \bigcap_{i=1,\ldots,m} \mathcal{X}_{\delta^{(i)}}, \end{aligned} \tag{9b}$$

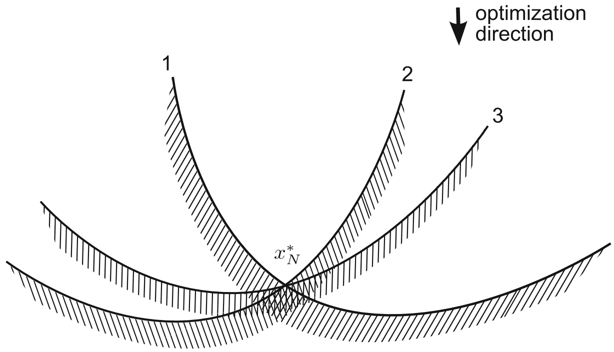

Fig. 2 Constraints 1, 2, and 3 are active, but 1 is the only support constraint, since removing 2 while maintaining 1 and 3 or removing 3 while maintaining 1 and 2 does not change the solution x^*_N. If the sole support constraint is maintained, then the solution moves to a lower value

when $m = 1, 2, \ldots$, where $\delta^{(i)}$, $i = 1, \ldots, m$, is an i.i.d. sample from $(\Delta, \mathcal{F}, \mathbb{P})$. To these programs the same tie-break rule used to make x^*_N unique in (1) is applied.

The following assumptions are in order.

Assumption 1 (*Existence and uniqueness*) For every m and for every sample $\delta^{(i)}$, $i = 1, \ldots, m$, program (9) admits solution, which becomes unique after the application of the tie-break rule.

Similarly to Definition 2, a constraint $\mathcal{X}_{\delta^{(i)}}$ of (9) is called a "support constraint" if its removal changes the solution of (9). Support constraints are always active constraints. The converse is not true in general, and an active constraint need not be a support constraint. This can be easily understood by thinking of a given program to which a new constraint is added whose boundary passes through the solution of the initial program. The new constraint is not a support constraint of the augmented program since with or without this constraint the solution remains the same, and yet the new constraint is active. When all the active constraints are support constraints, which is the typical case, keeping the support constraints and removing all the other constraints leaves the solution unchanged. On the other hand, when some of the active constraints are not support constraints, maintaining only the support constraints gives a new program whose solution may be different from the solution of the initial program, see Fig. 2 for an example. If this happens, program (9) is called *degenerate*. The following assumption requires that program (9) is non-degenerate with probability 1.

Assumption 2 (*Non-degeneracy*) For every m, with probability 1 with respect to the sample $\delta^{(i)}$ $i = 1, \ldots, m$, the solution to program (9) with all constraints in place coincides with the solution to the program where only the support constraints are kept.

Assumption 2 rules out situations where the boundary of distinct constraints accumulate anomalously with nonzero probability. Indeed, given $\delta^{(i)}$, $i = 1, \ldots, m$, one can isolate a minimal (of smallest cardinality) subset of constraints giving the same solution as (9). Then, in order for (9) to be degenerate, at least the boundary of one more constraint must pass through the solution given by the subset of constraints. Moreover, in degenerate cases, one can conceive using a heating and cooling approach akin to that of Section 3 of [12] to remove this assumption. More discussion on Assumption 2 is provided in the concluding Sect. 8.

In preparation of the main Theorem 1 below, consider now the following auxiliary variational problem (recall that d is the number of optimization variables in (1)):

$$\begin{aligned}\gamma^* = \inf_{\xi(\cdot)\in \mathsf{C}^d[0,1]} \quad & \xi(1) \\ \text{subject to:} \quad & \frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}\xi(t) \geq \binom{N}{k} t^{N-k} \cdot \mathbf{1}_{[0,1-\epsilon(k))}(t), \quad t \in [0,1], \\ & k = 0, 1, \ldots, d, \end{aligned} \tag{10}$$

where $\mathbf{1}_A(t)$ denotes the indicator function of set A, $\mathsf{C}^d[0,1]$ is the class of d times differentiable functions with continuous d-th derivative over the interval $[0,1]$, and $\frac{\mathrm{d}^k}{\mathrm{d}t^k}$ with $k=0$ means that no derivative operator is applied.[4]

Theorem 1 *Let* $\epsilon(k)$, $k = 0, 1, \ldots, d$, *be any* $[0,1]$-*valued function. Under Assumptions 1 and 2, it holds that*

$$\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} \leq \gamma^*,$$

where γ^* *is given by* (10).

[4] Problem (10) can be equivalently written as the following optimal control problem, which may offer further insight in the usability of the approach.
Auxiliary dynamical system:

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t} z_0(t) = z_1(t) \\ \frac{\mathrm{d}}{\mathrm{d}t} z_1(t) = 2\, z_2(t) \\ \quad \vdots \\ \frac{\mathrm{d}}{\mathrm{d}t} z_{d-2}(t) = (d-1)\, z_{d-1}(t) \\ \frac{\mathrm{d}}{\mathrm{d}t} z_{d-1}(t) = d\, u(t). \end{cases}$$

Auxiliary optimal control problem:

$$\begin{aligned}\gamma^* = \inf_{z_0(0),\ldots,z_{d-1}(0),u(\cdot)\in\mathsf{C}^0[0,1]} \quad & z_0(1) \\ \text{subject to:} \quad & z_k(t) \geq \binom{N}{k} t^{N-k} \cdot \mathbf{1}_{[0,1-\epsilon(k))}(t), \quad t\in[0,1],\ k = 0,1,\ldots,d-1, \\ & u(t) \geq \binom{N}{d} t^{N-d} \cdot \mathbf{1}_{[0,1-\epsilon(d))}(t), \quad t \in [0,1]. \end{aligned}$$

In the auxiliary optimal control problem, the optimization variables are the initial state and the input of the auxiliary dynamical system.

Proof See Sect. 5.1. □

Obtaining the result stated in Theorem 1 requires a main departure from the proof machinery developed in paper [12]. In the proof provided in Sect. 5.1, problems $(c, \mathcal{X}, \{\mathcal{X}_\delta\}, \mathbb{P})$ are first characterized in terms of a generalized moment problem, and the theorem is then proved by duality theory based on this characterization. Theorem 1 bears a new vision on scenario programs with profound implications, as discussed in the next section. We conclude this section by noticing that the main result (2) of [12] is a corollary of Theorem 1 by the selection $\epsilon(k) = \epsilon \ \forall k$.

Corollary 1 *Take $\epsilon(k) = \epsilon \ \forall k$. Under Assumptions 1 and 2, it holds that*

$$\mathbb{P}^N\{V(x_N^*) > \epsilon\} \leq \sum_{i=0}^{d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i}.$$

Proof See Sect. 5.2. □

4 Consequences and practical use of Theorem 1

As compared to Corollary 1, the power of Theorem 1 stems from the flexibility given by the fact that $\epsilon(s_N^*)$ is a-posteriori evaluated depending on the number of support constraints found in the scenario program at hand. This flexibility is explored in this section. Implementation schemes are also provided that allow for an easy use of Theorem 1.

Theorem 1 gives a bound γ^* on the probability that $V(x_N^*) > \epsilon(s_N^*)$. In normal cases, one desires that $V(x_N^*) > \epsilon(s_N^*)$ happens with very low probability so that $\epsilon(s_N^*)$ can be taken as an upper bound to $V(x_N^*)$ with high confidence. This suggests that for a practical use of Theorem 1 one reverts the order in which quantities in (10) are computed: one first assigns a very small β (e.g. $\beta = 10^{-6}$), and then a function $\epsilon(k)$ is determined, which, substituted in (10), gives $\gamma^* \leq \beta$ so that $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} \leq \beta$ by Theorem 1. It turns out that, given a β, infinitely many functions $\epsilon(k)$ attain the desired result. The following Theorem 2 describes one such function. The significance of this choice is discussed after the theorem.

Theorem 2 *Given $\beta \in (0, 1)$, for any $k = 0, 1, \ldots, d$, the polynomial equation in the t variable*

$$\frac{\beta}{N+1} \sum_{m=k}^{N} \binom{m}{k} t^{m-k} - \binom{N}{k} t^{N-k} = 0 \tag{11}$$

has one and only one solution $t(k)$ in the interval $(0, 1)$. Letting $\epsilon(k) := 1 - t(k)$, under Assumptions 1 and 2, it holds that

$$\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} \leq \beta.$$

Proof See Sect. 5.3. □

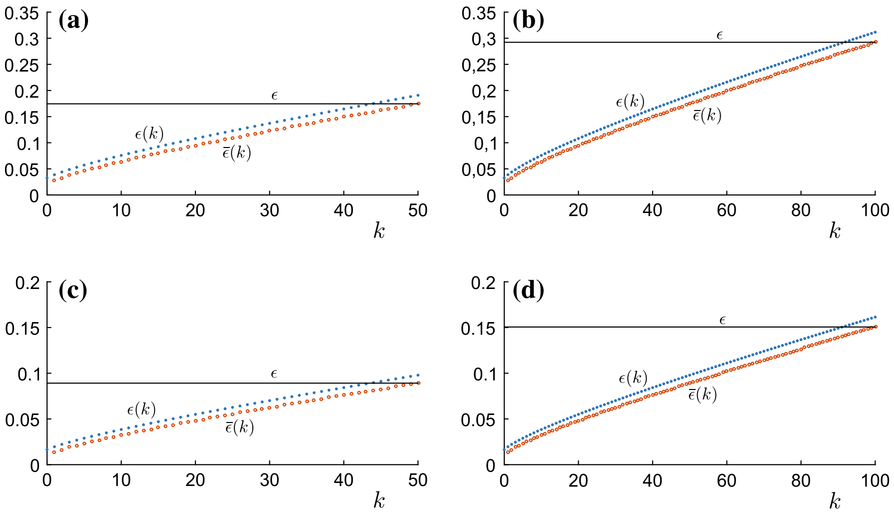

Fig. 3 Comparison between $\epsilon(k)$, ϵ, and $\bar{\epsilon}(k)$. **a** $N = 500$ $d = 50$; **b** $N = 500$ $d = 100$; **c** $N = 1000$ $d = 50$; **d** $N = 1000$ $d = 100$

To compute the $\epsilon(k)$, $k = 0, 1, \ldots, d$, given in Theorem 2 one can e.g. use a bisection numerical algorithm. A MATLAB code for this is provided in Appendix 2. It can be cut and pasted into the MATLAB workspace for a handy implementation.

Figure 3 profiles the function $\epsilon(k)$ given by Theorem 2 when $\beta = 10^{-6}$ against the ϵ obtained from Corollary 1 for the same confidence value, i.e. ϵ such that $\sum_{i=0}^{d-1} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i} = 10^{-6}$. The figure refers to various choices of N and d. $\epsilon(k)$ is smaller than ϵ for most values of k, while $\epsilon(k)$ is slightly bigger than ϵ for k close to d. Notice that no theory can deliver a valid $\epsilon(k)$ such that $\epsilon(k) < \epsilon$ $\forall k$, since the ϵ given by Corollary 1 is tight for fully supported problems in dimension d (see [12]).[5]

Further, it is interesting to compare the function $\epsilon(k)$ of Theorem 2 with an insurmountable lower limit. Similarly to point (iii) in the preview example of Sect. 2, one can consider fully-supported problems in dimension k, $k = 1, \ldots, d$, and augment them with $d - k$ dummy variables to recast the problems in dimension d. After computing $\bar{\epsilon}(k)$ from relation $\sum_{i=0}^{k-1} \binom{N}{i} \bar{\epsilon}(k)^i (1 - \bar{\epsilon}(k))^{N-i} = \beta$, one notices that $\bar{\epsilon}(k)$ is exact for fully-supported problems in dimension k, i.e. (3) gives $\mathbb{P}^N\{V(x_N^*) > \bar{\epsilon}(k)\} = \beta$, and the same holds for the augmented problems in dimension d. Since $\epsilon(k)$ is valid for any problem in dimension d, it must necessarily be no less than $\bar{\epsilon}(k)$, that is, $\epsilon(k) \geq \bar{\epsilon}(k)$, $k = 1, \ldots, d$. Figure 3 also profiles $\bar{\epsilon}(k)$ for an easy comparison with $\epsilon(k)$.

Figure 3 suggests an additional interesting observation. Since $\bar{\epsilon}(k)$ is also insurmountable in dimension k, one sees that a-posteriori observing k support constraints leads to a conclusion which is close to the best possible result attainable when one

[5] At the present state of knowledge, whether or not a valid $\epsilon(k)$ exists such that $\epsilon(k) < \epsilon$ for some k while $\epsilon(k) \leq \epsilon$ for any k is still an open problem.

works with k optimization variables only. Similarly, knowing in advance that the number of support constraints never exceeds k in a problem in dimension $d > k$ provides little advantage as compared to waiting and seeing that the cardinality of the set of support constraints is k. This result embodies the essence of the "wait-and-judge" philosophy: exploiting the information contained in $\delta^{(1)}, \ldots, \delta^{(N)}$ by a-posteriori assessing the number of support constraints in the program at hand "kills" the advantage that comes from knowing in advance an upper limit to the largest possible cardinality of the support constraint set.

5 Proofs of results in Sects. 3 and 4

5.1 Proof of Theorem 1

5.1.1 The distributions $F_0, F_1, \ldots, F_d$

For $k = 0, 1, \ldots, d$, consider the scenario program with only k constraints

$$\begin{aligned} \mathrm{SP}_k \, : \, & \min_{x \in \mathcal{X}} c^T x \\ & \text{subject to: } x \in \bigcap_{i=1,\ldots,k} \mathcal{X}_{\delta^{(i)}}, \end{aligned}$$

where it is understood that the "subject to" part is suppressed when $k = 0$, and let x_k^* be its solution and s_k^* be the number of support constraints. Define

$$F_k(v) = \mathbb{P}^k\{V(x_k^*) \leq v \ \wedge \ s_k^* = k\}, \tag{12}$$

where $\wedge$ denotes the "and" operator. $F_k(v)$ is the probability that the sample $\delta^{(1)}, \ldots, \delta^{(k)}$ gives that all k constraints are of support and the solution has a violation no more than v. The F_k's are generalized distribution functions, that is, each F_k has the same properties as a distribution function except that its limit when $v \to +\infty$ need not be 1, see e.g. [47]. To a problem $\mathcal{P}$, that is, to a choice of $(c, \mathcal{X}, \{\mathcal{X}_\delta\}, \mathbb{P})$, there is associated a $(d{+}1)$-tuple $F_0, F_1, \ldots, F_d$, and the $(d{+}1)$-tuple $F_0, F_1, \ldots, F_d$ is different for different problems.

We next show that $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\}$ can be computed from $F_0, F_1, \ldots, F_d$. In words, $F_0, F_1, \ldots, F_d$ is the "backbone" that permits one to characterize the violation of the solution of a scenario program.

Start by noting that s_N^* takes value in $\{0, 1, \ldots, d\}$ and that the event where s_N^* takes on one value does not overlap with the event where it takes another value. Thus,

$$\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} = \sum_{k=0}^{d} \mathbb{P}^N\{V(x_N^*) > \epsilon(k) \ \wedge \ s_N^* = k\}. \tag{13}$$

The set

$$S_k = \{V(x_N^*) > \epsilon(k) \ \wedge \ s_N^* = k\} \subseteq \Delta^N,$$

which is the event where the violation of x_N^* is above $\epsilon(k)$ and there are k support constraints, can be decomposed as follows: for each sample $\delta^{(1)}, \ldots, \delta^{(N)} \in S_k$, consider the indexes of the corresponding k support constraints, and group together all the samples with the same indexes. In this way, $\binom{N}{k}$ subsets are constructed forming a partition of S_k. All these subsets have the same probability because of the i.i.d. assumption on the sample. Hence,

$$\mathbb{P}^N\{S_k\} = \binom{N}{k}\mathbb{P}^N\{A\}, \tag{14}$$

where A is one of these subsets, say the one where the indexes of the support constraints are $1, 2, \ldots, k$, viz.,

$$A := \{V(x_N^*) > \epsilon(k) \;\wedge\; s_N^* = k \;\wedge\; \text{the first } k \text{ constraints are of support}\}. \tag{15}$$

We next show that the probability of A is computed as

$$\mathbb{P}^N\{A\} = \int_{(\epsilon(k),1]} (1-v)^{N-k} \mathrm{d}F_k(v). \tag{16}$$

To prove (16), introduce the event

$$\begin{aligned} B \;:= \Big\{ & V(x_k^*) > \epsilon(k) \;\wedge\; s_k^* = k \;\wedge \text{the constraints with indexes} \\ & k+1, \ldots, N \text{ are satisfied by } x_k^* \Big\}. \end{aligned}$$

It can be shown that $A = B$ up to a zero probability set, so that $\mathbb{P}^N\{A\} = \mathbb{P}^N\{B\}$. The proof of this intuitive fact is differed till the end of this Sect. 5.1.1 to avoid breaking the flow of discourse here. We here concentrate on showing that the probability of B is given by the right-hand side of (16), namely, $\mathbb{P}^N\{B\} = \int_{(\epsilon(k),1]}(1-v)^{N-k}\mathrm{d}F_k(v)$. To show this, note that for a fixed value v of the violation of the solution x_k^* generated by the first k constraints, $(1-v)^{N-k}$ is the probability that the other $N-k$ constraints are satisfied by x_k^*, so that, recalling the definition (12) of F_k, the integral of $(1-v)^{N-k}$ over the interval $(\epsilon(k), 1]$ with respect to F_k yields $\mathbb{P}^N\{B\}$. Thus, (16) remains proven. Wrapping up, substituting (16) in (14) and further plugging the result into (13) yields

$$\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} = \sum_{k=0}^{d} \binom{N}{k} \int_{(\epsilon(k),1]} (1-v)^{N-k} \mathrm{d}F_k(v). \tag{17}$$

This is a fundamental formula by which $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\}$ can be computed from $F_0, F_1, \ldots, F_d$.

Proof of the fact that $A = B$ up to a zero probability set

We first prove that $A \subseteq B$ up to a zero probability set.
Since in A the support constraints are the first k, by the non-degeneracy Assumption 2, $x_N^* = x_k^*$ up to a zero probability set. Thus, $V(x_k^*) = V(x_N^*) > \epsilon(k)$ up to a zero

probability set. Moreover, the problem with only the first k constraints has clearly this k constraints as support constraints while all the other constraints are satisfied.

Next, we show that $B \subseteq A$ up to a zero probability set.

In B, the constraints with indexes $k+1, \dots, N$ are satisfied by x_k^*, thus $x_N^* = x_k^*$ and $V(x_N^*) = V(x_k^*) > \epsilon(k)$. The first k constraints are the support constraints for the program with N constraints up to a zero probability set, a fact that we prove by contradiction. Assume that not all the first k constraints are of support for the program with N constraints with non-zero probability. Since the other constraints with indexes $k+1, \dots, N$ cannot be of support for the program with N constraints (because, if one of them is removed, the solution $x_N^* = x_k^*$ does not change), then the set of support constraints for the program with N constraints would be a strict subset of the set of support constraints for the program with the first k constraints. But then the solution of the program with the sole support constraints for the program with N constraints would be different from x_k^* and, hence, different from x_N^* with non-zero probability. This, however, contradicts the non-degeneracy Assumption 2.

5.1.2 Moment conditions on $F_0, F_1, \dots, F_d$

Equation (17) allows one to compute the probability that $V(x_N^*) > \epsilon(s_N^*)$ from $F_0, F_1, \dots, F_d$. Given an arbitrary $(d+1)$-tuple of generalized distribution functions F_k's, it may or may not be the case that these F_k's are associated to some convex optimization problem $\mathcal{P}$. In other words, the set of all the F_k's that are compatible with convex optimization problems does not coincide with the set of all generalized distribution functions. We next give moment conditions that are necessarily satisfied by the F_k's that are compatible with convex optimization problems. The proof of Theorem 1 will then be obtained by taking sup of $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\}$ over the F_k's satisfying these conditions.

Consider again Eq. (17). If the $>$ in the left-hand side of this equation is replaced by $\geq$, then it is immediate to see that the equation still holds provided that the integral on the right-hand side is computed over the closed interval $[\epsilon(k), 1]$. Moreover, in (17) we can also substitute N with a generic m ranging over all integers $0, 1, \dots$, with the precaution that when $m < d$, so that the number of support constraints cannot possibly be more than m, the summation goes from 0 to m. With these generalizations, and taking $\epsilon(k) = 0$ for any k and further noting that $\mathbb{P}^m\{V(x_m^*) \geq 0\}$ is equal to 1, we obtain

$$\sum_{k=0}^{\min\{m,d\}} \binom{m}{k} \int_{[0,1]} (1-v)^{m-k} \mathrm{d}F_k(v) = 1, \quad m = 0, 1, \dots. \tag{18}$$

Equation (18) gives joint conditions on the moments of the F_k's that need be satisfied for the F_k's to be compatible with convex optimization problems.[6] Since more than one distribution is involved, the infinitely many conditions in (18), one for any m, do not completely determine the F_k's, and more choices of F_k's satisfy (18). This is not

[6] Equation (18) can also be written in a more compact form using moment generating functions. Let $\tilde{F}_k(t) := 1 - F_k(1-t)$ and $\tilde{M}_k(z)$ be the moment generating function of $\tilde{F}_k(t)$. Multiplying the two sides of (18) by $z^m/m!$ and summing up over m gives the equivalent characterization:

surprising as not all convex optimization problems can be expected to have the same F_k's. To help a more concrete vision, Appendix 3 provides a couple of examples of F_k's that satisfy Eq. (18) and are indeed associated to convex optimization problems.

5.1.3 *Primal problem for computing* $\sup_{(c,\mathcal{X},\{\mathcal{X}_\delta\},\mathbb{P})} \mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\}$

The probability $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\}$ can be upper bounded for all convex optimization problems by maximizing the right-hand side of (17) over the set of the F_k's satisfying Eq. (18), i.e., $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} \le \gamma$ with γ given by

$$\begin{aligned} \gamma = \sup_{F_0, F_1, \ldots, F_d} \quad & \sum_{k=0}^{d} \binom{N}{k} \int_{(\epsilon(k),1]} (1-v)^{N-k} \mathrm{d}F_k(v) \\ \text{subject to:} \quad & \sum_{k=0}^{\min\{m,d\}} \binom{m}{k} \int_{[0,1]} (1-v)^{m-k} \mathrm{d}F_k(v) = 1, \quad m = 0, 1, \ldots \\ & F_0, F_1, \ldots, F_d \in \mathcal{C}, \end{aligned} \tag{19}$$

where $\mathcal{C}$ is the positive cone of generalized distribution functions. This is a generalized moment problem that involves $d+1$ distributions [27]. In the next section, problem (19) is studied by duality.

5.1.4 *Duality analysis and conclusions*

For any $M \ge d$, consider the following truncated version of problem (19) that only has finitely many moment constraints

$$\begin{aligned} \gamma_M = \sup_{F_0, F_1, \ldots, F_d} \quad & \sum_{k=0}^{d} \binom{N}{k} \int_{(\epsilon(k),1]} (1-v)^{N-k} \mathrm{d}F_k(v) \\ \text{subject to:} \quad & \sum_{k=0}^{\min\{m,d\}} \binom{m}{k} \int_{[0,1]} (1-v)^{m-k} \mathrm{d}F_k(v) = 1, \quad m = 0, 1, \ldots, M \\ & F_0, F_1, \ldots, F_d \in \mathcal{C}. \end{aligned} \tag{20}$$

Footnote 6 continued

$$\sum_{k=0}^{d} \frac{z^k}{k!} \tilde{M}_k(z) = \exp(z),$$

which is explained as follows. The right-hand side is obtained because $\sum_{m=0}^{\infty} z^m/m! = \exp(z)$, while in the left-hand side the kth term is calculated as follows, $\sum_{m=k}^{\infty} z^m/m! \binom{m}{k} \int_{[0,1]} t^{m-k}\,\mathrm{d}\tilde{F}_k(t) = z^k/k! \int_{[0,1]} \sum_{m=k}^{\infty} (zt)^{m-k}/(m-k)!\,\mathrm{d}\tilde{F}_k(t) = z^k/k! \int_{[0,1]} \exp(zt)\mathrm{d}\tilde{F}_k(t) = (z^k/k!)\tilde{M}_k(z)$.

Since in (20) the number of constraints increases with M and, for any M, (20) is less constrained than (19), γ_M is non increasing and $\gamma \leq \gamma_M$ $\forall M$. The dual problem, [3], of (20) is

$$\begin{aligned}\gamma_M^* = \inf_{\lambda_0,\lambda_1,\dots,\lambda_M} \quad & \sum_{m=0}^{M} \lambda_m \\ \text{subject to:} \quad & \sum_{m=k}^{M} \lambda_m \binom{m}{k} (1-v)^{m-k} \geq \binom{N}{k} (1-v)^{N-k} \cdot \mathbf{1}_{(\epsilon(k),1]}(v), \;\; v \in [0,1] \\ & k = 0, 1, \dots, d, \end{aligned} \tag{21}$$

where $\mathbf{1}_A(v)$ denotes the indicator function of set A. By weak duality, we have that $\gamma_M \leq \gamma_M^*$, as can be easily established by observing that the following inequality holds for any feasible point $F_0, F_1, \dots, F_d$ of (20) and any feasible point $\lambda_0, \lambda_1, \dots, \lambda_M$ of (21):

$$\begin{aligned}\sum_{k=0}^{d} \binom{N}{k} \int_{(\epsilon(k),1]} (1-v)^{N-k} \mathrm{d}F_k(v) &= \sum_{k=0}^{d} \int_{[0,1]} \binom{N}{k} (1-v)^{N-k} \cdot \mathbf{1}_{(\epsilon(k),1]}(v) \mathrm{d}F_k(v) \\ &\leq \sum_{k=0}^{d} \int_{[0,1]} \sum_{m=k}^{M} \lambda_m \binom{m}{k} (1-v)^{m-k} \mathrm{d}F_k(v) \\ &= \sum_{m=0}^{M} \lambda_m \sum_{k=0}^{\min\{m,d\}} \binom{m}{k} \int_{[0,1]} (1-v)^{m-k} \mathrm{d}F_k(v) \\ &= \sum_{m=0}^{M} \lambda_m. \end{aligned}$$

Thus, $\gamma \leq \gamma_M \leq \gamma_M^*$ for any M, and, hence,

$$\gamma \leq \inf_M \gamma_M^*. \tag{22}$$

We next show that problem (21) can be recast as a variational problem from which $\inf_M \gamma_M^*$ can be evaluated.

For convenience let $t := 1 - v$, and note that

$$\frac{1}{k!} \frac{\mathrm{d}^k}{\mathrm{d}t^k} t^m = \begin{cases} 0 & m < k \\ \binom{m}{k} t^{m-k} & m \geq k \end{cases}. \tag{23}$$

Now, define

$$p(t) = \sum_{m=0}^{M} \lambda_m t^m.$$

Using (23), one sees that

$$\frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}p(t) = \sum_{m=k}^{M} \lambda_m \binom{m}{k} t^{m-k},$$

which is the left-hand side of the constraints in (21). Noticing also that the objective $\sum_{m=0}^{M} \lambda_m$ of (21) equals $p(1)$, the dual problem (21) can be rewritten as

$$\begin{aligned} \gamma_M^* = & \inf_{p(\cdot)\in \mathsf{P}_M} \quad p(1) \\ & \text{subject to: } \frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}p(t) \geq \binom{N}{k} t^{N-k} \cdot \mathbf{1}_{[0,1-\epsilon(k))}(t), \quad t \in [0,1], \\ & k = 0, 1, \ldots, d, \end{aligned}$$

where P_M is the class of polynomials of degree M.

We next want to show that $\inf_M \gamma_M^* = \gamma^*$, the optimal value of (10). To this purpose, consider the feasibility domain $\mathcal{F} \subseteq \mathsf{C}^d[0,1]$ of problem (10). We show that $(\bigcup_{M\geq d} \mathsf{P}_M) \cap \mathcal{F}$ is dense in $\mathcal{F}$ with respect to the distance $d(\cdot,\cdot)$ in $\mathsf{C}^d[0,1]$ ($d(\xi,\zeta) = \sum_{k=0}^{d} \max_{t\in[0,1]} \left| \frac{\mathrm{d}^k}{\mathrm{d}t^k}\xi(t) - \frac{\mathrm{d}^k}{\mathrm{d}t^k}\zeta(t) \right|$). Indeed, if $\xi(t) \in \mathcal{F}$, then, for any $\alpha > 0$, $\xi(t) + \alpha \exp(t)$ is an interior point of $\mathcal{F}$ because the term $\alpha \exp(t)$ increases all the derivatives in the left-hand side of the constraints in (10) and moves them away from the boundary of the constraints given by the right-hand side of (10), viz. $\binom{N}{k} t^{N-k} \cdot \mathbf{1}_{[0,1-\epsilon(k))}(t)$. Therefore, any point in $\mathcal{F}$ admits an interior point of $\mathcal{F}$ arbitrarily close to it (take α small enough). If we now take any small ball in the $\mathsf{C}^d[0,1]$ metric all contained in $\mathcal{F}$ and centered in $\xi(t)+\alpha\exp(t)$, we can further find a polynomial $p(t) \in \bigcup_{M\geq d} \mathsf{P}_M$ in this ball, and thereby contained in $\mathcal{F}$ and arbitrarily close to $\xi(t)$. Indeed, polynomial $p(t)$ can be constructed as follows: by Weierstrass theorem, the d-th derivative $\frac{\mathrm{d}^d}{\mathrm{d}t^d}[\xi(t)+\alpha\exp(t)]$ can be approximated uniformly over $[0,1]$ with a polynomial $q(t)$; then, $p(t)$ is the polynomial whose d-th derivative is $q(t)$ and such that

$$\left.\frac{\mathrm{d}^k}{\mathrm{d}t^k}p(t)\right|_{t=0} = \left.\frac{\mathrm{d}^k}{\mathrm{d}t^k}[\xi(t)+\alpha\exp(t)]\right|_{t=0}, \quad k = 0, 1, \ldots, d-1.$$

Hence, density of $(\bigcup_{M\geq d} \mathsf{P}_M) \cap \mathcal{F}$ in $\mathcal{F}$ remains proven. The conclusion that

$$\inf_M \gamma_M^* = \gamma^*, \tag{24}$$

now follows by observing that the cost $\xi(1)$ in (10) is a continuous functional from $\mathsf{C}^d[0,1]$ to $\mathbb{R}$.

To conclude the proof of Theorem 1, use (19), (22) and (24) in succession to obtain

$$\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} \leq \gamma \leq \inf_M \gamma_M^* = \gamma^*.$$

□

5.2 Proof of Corollary 1

Consider problem (10) with $\epsilon(0) = \epsilon(1) = \cdots = \epsilon(d) = \epsilon$. In this case, the constraints in (10) are written as

$$\frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}\xi(t) \geq \binom{N}{k} t^{N-k} \cdot \mathbf{1}_{[0,1-\epsilon)}(t), \quad k = 0, 1, \ldots, d. \tag{25}$$

Take $\bar{\xi}(t)$ to be function $d!\binom{N}{d}t^{N-d}\mathbf{1}_{[0,1-\epsilon)}(t)$ integrated d times. A simple computation yields

$$\begin{aligned}\frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}\bar{\xi}(t) = &\binom{N}{k}t^{N-k}\mathbf{1}_{[0,1-\epsilon)}(t)\\ &+\sum_{i=k}^{d-1}\binom{i}{k}\binom{N}{i}(t-1+\epsilon)^{i-k}(1-\epsilon)^{N-i}\mathbf{1}_{[1-\epsilon,1]}(t),\end{aligned}$$

$k = 0, 1, \ldots, d$, showing that $\bar{\xi}(t)$ satisfies (25).

One thing that should be noticed is that $\frac{\mathrm{d}^d}{\mathrm{d}t^d}\bar{\xi}(t) = d!\binom{N}{d}t^{N-d}\mathbf{1}_{[0,1-\epsilon)}(t)$ is not continuous, so that $\bar{\xi}(t)$ does not belong to $\mathsf{C}^d[0, 1]$. However, we show that the optimal value of (10) still satisfies $\gamma^* \leq \bar{\xi}(1)$, and the corollary remains proven in view of Theorem 1 by noticing that

$$\bar{\xi}(1) = \sum_{i=0}^{d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i}. \tag{26}$$

To prove that $\gamma^* \leq \bar{\xi}(1)$, consider the sequence of continuous functions

$$\begin{aligned}f_n(t) = d!\binom{N}{d}t^{N-d}\mathbf{1}_{[0,1-\epsilon)}(t) + \frac{n}{\epsilon}\left(1-\epsilon+\frac{\epsilon}{n}-t\right)d!\\ \times\binom{N}{d}(1-\epsilon)^{N-d}\mathbf{1}_{[1-\epsilon,1-\epsilon+\frac{\epsilon}{n})}(t), \quad n = 1, 2, \ldots,\end{aligned}$$

profiled in Fig. 4, and let $\bar{\xi}_n(t)$ be function $f_n(t)$ integrated d times.

$\bar{\xi}_n(t)$ satisfies (25) for any n, and the result is obtained by letting $n \to \infty$. □

Remark 1 It is actually true that $\gamma^* = \bar{\xi}(1)$, a fact that can be seen in two different ways. First, the d-th derivative $d!\binom{N}{d}t^{N-d}\mathbf{1}_{[0,1-\epsilon)}(t)$ of $\bar{\xi}(t)$ cannot be further decreased without violating (25), and hence γ^* cannot be reduced below $\bar{\xi}(1)$. Second, in [12] it has been shown that $\mathbb{P}^N\{V(x_N^*) > \epsilon\} = \sum_{i=0}^{d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i}$ for fully-supported problems. Should γ^* be further reducible below the value of $\bar{\xi}(1)$ given in (26), one would obtain from Corollary 1 the result that $\mathbb{P}^N\{V(x_N^*) > \epsilon\} < \sum_{i=0}^{d-1}\binom{N}{i}\epsilon^i(1-\epsilon)^{N-i}$, which would be in contradiction with the result in [12].

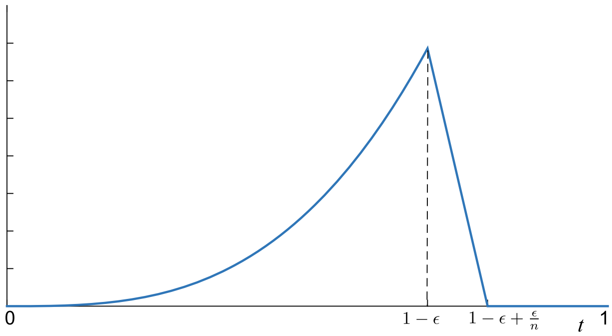

Fig. 4 Function $f_n(t)$

5.3 Proof of Theorem 2

In problem (10), take

$$\bar{\xi}(t) = \frac{\beta}{N+1}\sum_{m=0}^{N} t^m, \tag{27}$$

which gives

$$\frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}\bar{\xi}(t) = \frac{\beta}{N+1}\sum_{m=k}^{N}\binom{m}{k}t^{m-k}, \quad k = 1, \ldots, d. \tag{28}$$

We will show that:

(i) the intersection between $\frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}\bar{\xi}(t)$ and the function $\binom{N}{k}t^{N-k}$ in the interval (0, 1) is unique (and given by $t(k)$ obtained from (11)), and, moreover,

(ii) it holds that

$$\frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}\bar{\xi}(t) \geq \binom{N}{k}t^{N-k}, \quad k = 0, \ldots, d, \tag{29}$$

for $t < t(k) = 1 - \epsilon(k)$.

Hence, $\bar{\xi}(t)$ is feasible for problem (10), and the statement of the theorem easily follows from Theorem 1 because

$$\gamma^* \leq \bar{\xi}(1) = \frac{\beta}{N+1}\sum_{m=0}^{N} 1 = \beta.$$

To prove (i) and (ii), define

$$\varphi_k(t) = \frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}\bar{\xi}(t) - \binom{N}{k}t^{N-k} = \frac{\beta}{N+1}\sum_{m=k}^{N}\binom{m}{k}t^{m-k} - \binom{N}{k}t^{N-k},$$

for all $k = 0, 1, \ldots, N-1$. Here, we regard $\varphi_k(t)$ as a function defined for all $t > 0$. By induction, we show that $\varphi_k(t) = 0$, $k = 0, 1, \ldots, N-1$, has a unique solution in (0, 1), which we denote $t(k)$ also for $k > d$, and that $\varphi_k(t) > 0$ for $t \in [0, t(k))$, $\varphi_k(t) < 0$ for $t > t(k)$, and $\varphi_k(t) \to -\infty$ as $t \to +\infty$. Note that this suffices to prove (i) and (ii) so concluding the proof of the theorem. The result is trivially true for $\varphi_{N-1}(t) = \frac{\beta}{N+1}(1 + Nt) - Nt = \frac{\beta}{N+1} - N(1 - \frac{\beta}{N+1})t$, which is a straight line. If the result holds true for $\varphi_k(t)$, then it also holds true for $\varphi_{k-1}(t)$ because

$$\varphi_{k-1}(t) = \frac{\beta}{N+1} + k \int_0^t \varphi_k(\tau)\mathrm{d}\tau,$$

and, thanks to the inductive assumption on $\varphi_k(t)$, $\int_0^t \varphi_k(\tau)\mathrm{d}\tau$ is strictly increasing and ≥ 0 till $t(k)$ and then strictly decreasing and diverging to $-\infty$ for $t > t(k)$. The fact that $t(k) \in (0, 1)$ holds because

$$\begin{aligned}\varphi_k(1) &= \frac{\beta}{N+1} \sum_{m=k}^{N} \binom{m}{k} - \binom{N}{k} \leq \frac{\beta}{N+1} \sum_{m=0}^{N} \binom{N}{k} - \binom{N}{k} \\ &= (\beta - 1)\binom{N}{k} < 0.\end{aligned}$$

□

6 Scenario optimization over generic sets

In previous sections, convex, finite-dimensional, scenario optimization programs have been investigated. It turns out that the key ideas developed there have more general breadth than what has been exploited so far, and they carry over, with suitable modifications, to optimization problems defined over generic sets. This extension is presented in this section.

Let X be a generic set. For example, X can be an infinite dimensional vector space or just a set without an algebraic structure. Let $f(x)$ be a real-valued function defined over X, and let $\mathcal{X}$, $\mathcal{X}_\delta$ be subsets of X, where δ is a random outcome from a probability space $(\Delta, \mathcal{F}, \mathbb{P})$. No restrictions apply to f, $\mathcal{X}$, and $\mathcal{X}_\delta$. For example, when X is a vector space, $f(x)$ is not required to be a convex function, nor $\mathcal{X}$ and $\mathcal{X}_\delta$ are required to be convex sets. The scenario optimization program is

$$\begin{aligned}&\min_{x \in \mathcal{X}} \ f(x) \\ &\text{subject to: } x \in \bigcap_{i=1,\ldots,N} \mathcal{X}_{\delta^{(i)}},\end{aligned} \tag{30}$$

where $\delta^{(i)}$, $i = 1, \ldots, N$, is an i.i.d. sample from $(\Delta, \mathcal{F}, \mathbb{P})$ and N is any positive integer. Definition 1 of violation and Definition 2 of support constraint hold unchanged. Assumptions 1 and 2 are still in force with (9) written with the obvious modification that $c^T x$ is replaced by $f(x)$. Regarding Assumption 1, conditions for the existence of the solution is a classical topic in optimization and are discussed e.g. in [30,33].

Again, x_N^* is the solution uniquely identified by a tie-break rule, which in the present setup is not required to be convex. Assumption 2 is key to obtain our results, and Sect. 8 provides more discussion on this assumption.

In the present context, the number of support constraints is not a-priori bounded. Hence, in program (30), s_N^* is only bounded by N. For instance, Example 1 below introduces a situation where s_N^* is systematically equal to N. Correspondingly, $\epsilon(k)$ is a function ranging over $k = 0, 1, \ldots, N$. As before, guarantees on the violation of x_N^* are adapted to s_N^*, and we want to compute $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\}$. The apparatus to perform this computation is similar to the one developed for the finite-dimensional convex setup, where the auxiliary variational problem is modified as follows.

$$\begin{aligned} \gamma^* = & \inf_{\xi(\cdot)\in \mathsf{P}_N} \quad \xi(1) \\ & \text{subject to:} \quad \frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}\xi(t) \geq \binom{N}{k} t^{N-k} \cdot \mathbf{1}_{[0,1-\epsilon(k))}(t), \quad t \in [0,1], \\ & \qquad\qquad\quad k = 0, 1, \ldots, N, \end{aligned} \tag{31}$$

where P_N is the class of polynomials of degree N.

Theorem 3 *Let* $\epsilon(k)$, $k = 0, 1, \ldots, d$, *be any* $[0, 1]$-*valued function. Under Assumptions* 1 *and* 2 *(with* $c^T x$ *in* (9) *replaced by* $f(x)$ *in the statements of these assumptions), it holds that*

$$\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} \leq \gamma^*,$$

where γ^* *is given by* (31).

Proof See Sect. 7.1. □

The counterpart of Theorem 2 becomes as follows.

Theorem 4 *Given* $\beta \in (0, 1)$, *for any* $k = 0, 1, \ldots, N-1$, *the polynomial equation in the* t *variable*

$$\frac{\beta}{N+1}\sum_{m=k}^{N}\binom{m}{k}t^{m-k} - \binom{N}{k}t^{N-k} = 0 \tag{32}$$

has one and only one solution $t(k)$ *in the interval* $(0, 1)$. *Letting* $\epsilon(k) := 1 - t(k)$, $k = 0, 1, \ldots, N-1$, *and* $\epsilon(N) = 1$, *under Assumptions* 1 *and* 2 *(with* $c^T x$ *in* (9) *replaced by* $f(x)$ *in the statements of these assumptions), it holds that*

$$\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} \leq \beta.$$

Proof See Sect. 7.2.

In the theorem, $\epsilon(N)$ is set to 1. It could not be otherwise, because there are problems where the number of support constraints is systematically equal to N and $V(x_N^*)$ is always equal to 1 so that $\mathbb{P}^N\{V(x_N^*) > \epsilon(N)\} = 1$ whenever $\epsilon(N) < 1$. See Example 1 for one such problem.

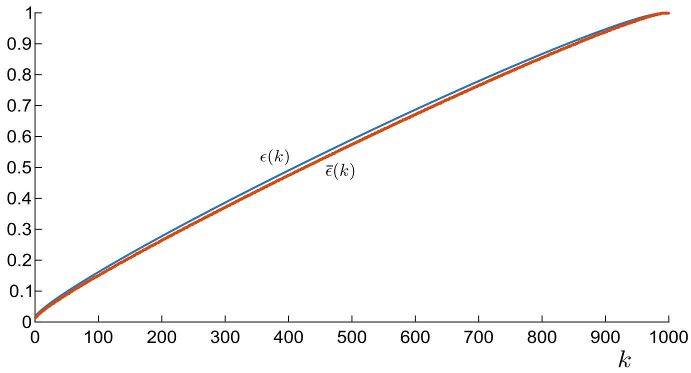

Fig. 5 Function $\epsilon(k)$ versus $\bar{\epsilon}(k)$

Equation (32) is the same as Eq. (11) where, however, k ranges over a wider interval of integers that extends till N. Figure 5 displays function $\epsilon(k)$ obtained from (32) when $\beta = 10^{-6}$ and $N = 1000$. The insurmountable lower limit $\bar{\epsilon}(k)$ obtained from fully-supported problems in dimension $k = 1, \dots, N$ (that is, $\bar{\epsilon}(k)$ is such that $\sum_{i=0}^{k-1} \binom{N}{i} \bar{\epsilon}(k)^i (1-\bar{\epsilon}(k))^{N-i} = \beta$, see the discussion in Sect. 4) is also visualized. The two curves are quite close to each other, which shows a fact that much surprised the authors of this paper at the time of discovery: even in infinite dimensional problems, the conclusion drawn after a-posteriori inspecting that the number of support constraints is k is almost the same as the conclusion obtainable when the problem is from the outset in dimension k.

An example illustrates the results of this section.

Example 1 (*Convex hull in* $\mathbb{R}^2$) Points $p^{(i)}$ $i = 1, \dots, N$, are independently sampled from a probability distribution $\mathbb{P}$ on $\mathbb{R}^2$ and the problem of constructing the smallest convex set that contains all the points is considered:

$$\min_{C \in \mathbf{C}} \ \mu(C) \qquad (33)$$
$$\text{subject to:} \quad p^{(i)} \in C, \quad i = 1, \dots, N,$$

where μ is Lebesgue measure on $\mathbb{R}^2$ and $\mathbf{C}$ is the collection of all convex sets of $\mathbb{R}^2$. Program (33) is a scenario program with $\mathcal{X} = \mathbf{C}$, $f(x) = \mu(C)$, $\delta^{(i)} = p^{(i)}$, and $\mathcal{X}_{\delta^{(i)}} = \{C \in \mathbf{C} : \ p^{(i)} \in C\}$. Its unique solution C_N^* is the convex hull of points $p^{(i)}$ $i = 1, \dots, N$, and the problem is non-degenerate if and only if $\mathbb{P}$ has no concentrated mass on isolated points. As a matter of fact, when the $p^{(i)}$'s are all distinct, the support constraints are those obtained in correspondence of the $p^{(i)}$'s at the vertexes of the convex hull, and the convex hull of the vertex points coincides with the convex hull of all points.

We want to evaluate the probability mass that is left outside the convex hull. This is the same as assessing the violation $V(C_N^*)$, and Theorem 4 is used to this purpose.

We consider two probability distributions $\mathbb{P}$. Suppose first that $\mathbb{P}$ is the uniform distribution on the boundary of a circle. In this case, the convex hull is a polygon

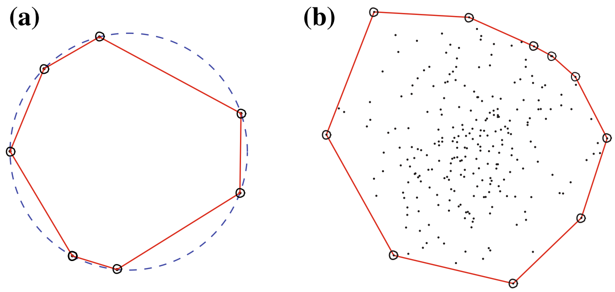

Fig. 6 Two convex hulls of N points. **a** Points are sampled from the boundary of a circle; **b** points are sampled from a Gaussian distribution

inscribed in the circle with vertexes coincident with the points $p^{(i)}$'s, see Fig. 6a for an instance with $N = 7$.

Hence, the number of support constraints is N, i.e. $s_N^* = N$, with probability one and Theorem 4 gives $\epsilon(s_N^*) = \epsilon(N) = 1$. This is the correct evaluation of $V(C_N^*)$ since every polygon inscribed in the circle leaves outside a probability mass equal to 1.

Suppose now that the points are sampled from a Gaussian distribution with zero mean and identity covariance matrix. See Fig. 6b for an instance with $N = 250$, where the number of support constraints is 10. Setting $\beta = 10^{-6}$, Eq. (32) gives $\epsilon(10) = 0.147$, that is, the probabilistic mass outside the obtained convex hull is no more than 14.7 %. To draw this conclusion, no use was made of the fact that the points were generated from a Gaussian distribution.

7 Proofs of the results in Sect. 6

7.1 Proof of Theorem 3

The proof of Theorem 3 is obtained along lines similar to the proof of Theorem 1, and we here highlight the differences.

Since in the present setup the number of support constraints of a program with m constraints is only bounded by m and m grows without limit, we define the F_k's as in Sect. 5.1.1, where this time k can be any nonnegative integer,

$$F_k(v) = \mathbb{P}^k\{V(x_k^*) \le v \ \wedge \ s_k^* = k\}, \quad k = 0, 1, \ldots.$$

Equation (13), and then (17), become

$$\begin{aligned}\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} &= \sum_{k=0}^{N} \mathbb{P}^N\{V(x_N^*) > \epsilon(k) \ \wedge \ s_N^* = k\} \\ &= \sum_{k=0}^{N} \binom{N}{k} \int_{(\epsilon(k),1]} (1-v)^{N-k} \mathrm{d}F_k(v),\end{aligned}$$

where summation runs up to N since the number of support constraints can take any value between 0 and N in this setup. In characterizing optimization problems in terms of $F_0, F_1, \ldots$, as it was done for convex problems in Sect. 5.1.2, we here have

$$\sum_{k=0}^{m} \binom{m}{k} \int_{[0,1]} (1-v)^{m-k} \mathrm{d}F_k(v) = 1, \quad m = 0, 1, \ldots,$$

where the number of terms in the summation grows unbounded with m. Again, we can write $\sup_{(c,\mathcal{X},\{\mathcal{X}_\delta\},\mathbb{P})} \mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\} \leq \gamma$ where γ is here obtained as

$$\begin{aligned} \gamma = \sup_{F_k,\, k=0,1,\ldots} \quad & \sum_{k=0}^{N} \binom{N}{k} \int_{(\epsilon(k),1]} (1-v)^{N-k} \mathrm{d}F_k(v) \\ \text{subject to:} \quad & \sum_{k=0}^{m} \binom{m}{k} \int_{[0,1]} (1-v)^{m-k} \mathrm{d}F_k(v) = 1, \quad m = 0, 1, \ldots \\ & F_k \in \mathcal{C}, k = 0, 1, \ldots . \end{aligned} \tag{34}$$

The conclusion in Theorem 3 is obtained by duality analysis as it was done in Sect. 5.1.4 for the finite dimensional convex case, and at this step we have to register the main differences from before. For any $M \geq N$, consider the following truncated version of problem (34)

$$\begin{aligned} \gamma_M = \sup_{F_0,F_1,\ldots,F_M} \quad & \sum_{k=0}^{N} \binom{N}{k} \int_{(\epsilon(k),1]} (1-v)^{N-k} \mathrm{d}F_k(v) \\ \text{subject to:} \quad & \sum_{k=0}^{m} \binom{m}{k} \int_{[0,1]} (1-v)^{m-k} \mathrm{d}F_k(v) = 1, \quad m = 0, 1, \ldots, M, \\ & F_0, F_1, \ldots, F_M \in \mathcal{C}, \end{aligned} \tag{35}$$

which can be dualized after the substitution $t := 1 - v$ as

$$\gamma_M^* = \inf_{\lambda_0,\lambda_1,\ldots,\lambda_M} \quad \sum_{m=0}^{M} \lambda_m \tag{36}$$

$$\text{subject to:} \quad \sum_{m=k}^{M} \lambda_m \binom{m}{k} t^{m-k} \geq \begin{cases} \binom{N}{k} t^{N-k} \cdot \mathbf{1}_{[0,1-\epsilon(k))}(t), & k = 0, 1, \ldots, N \\ 0, & k = N+1, \ldots, M \end{cases}, \quad t \in [0,1].$$

As in the finite dimensional case, we have $\gamma \leq \gamma_M \leq \gamma_M^*\ \forall M$, where the first inequality is because the truncated primal problem (35) is less constrained than (34) and the second inequality follows from weak duality. Thus, $\gamma \leq \inf_M \gamma_M^*$. Clearly, γ_M^* is a non-increasing function of M, but, somehow unexpectedly, we shall show that $\gamma_M^* = \gamma_N^*$ for any $M \geq N$, so that, unlike the finite dimensional case, pushing one's search beyond $M = N$ has no payoff, and we later use relation

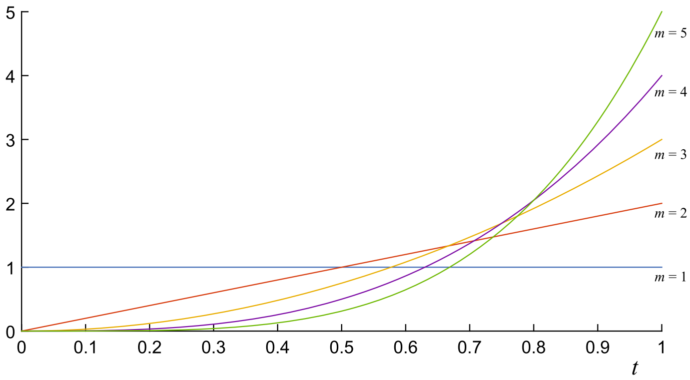

Fig. 7 Functions $\binom{m}{1}t^{m-1}$

$$\gamma \leq \gamma_N^*, \tag{37}$$

to draw the theorem's conclusion.

To prove that $\gamma_M^* = \gamma_N^*$ for any $M \geq N$, consider the family of functions $\binom{m}{k}t^{m-k}$ indexed with $m = k, \ldots, M$. To help intuition, these functions are profiled in Fig. 7 for $k = 1$ and $M = 5$.

Denote by $\bar{t}_{m,k}$, $m = k, \ldots, M-1$, the value $t > 0$ where one such function intersects the next, i.e.

$$\binom{m}{k}\left(\bar{t}_{m,k}\right)^{m-k} = \binom{m+1}{k}\left(\bar{t}_{m,k}\right)^{m+1-k}, \quad \bar{t}_{m,k} > 0.$$

It turns out that $\bar{t}_{m,k} = 1 - \frac{k}{m+1}$. The following properties hold:

(i) $\bar{t}_{m,k} \leq \bar{t}_{m+1,k}$;
(ii) function $\binom{m}{k}t^{m-k}$ is the largest function over $[\bar{t}_{m-1,k}, \bar{t}_{m,k}]$, where we let $\bar{t}_{k-1,k} = 0$;
(iii) $\binom{m}{k}t^{m-k} \geq \binom{m+h}{k}t^{m+h-k}$ over $[0, \bar{t}_{m,k}]$ $\forall h \geq 0$.

We next show that, for any feasible point $\boldsymbol{\lambda} = (\lambda_0, \lambda_1, \ldots, \lambda_M)$, there is another feasible point $\boldsymbol{\lambda}' = (\lambda_0', \lambda_1', \ldots, \lambda_M')$ with $\lambda_m' = 0$ for $m \geq N+1$ which attains the same value as $\boldsymbol{\lambda}$ in the dual problem (36). As a consequence, the optimal value γ_M^* of (36) is achieved in correspondence of a point $\boldsymbol{\lambda}'$ with $\lambda_m' = 0$ for $m \geq N+1$. In turn, this implies that $\gamma_M^* = \gamma_N^*$ for any $M \geq N$.

So, let $\boldsymbol{\lambda}$ be a feasible point, which satisfies the constraints in (36). Notice that evaluating the constraints at $t = 0$ gives $\lambda_m \geq 0$ $m = 0, 1, \ldots, M$. Two cases may occur, $\sum_{m=0}^{M} \lambda_m \geq 1$ or $\sum_{m=0}^{M} \lambda_m < 1$.

If $\sum_{m=0}^{M} \lambda_m \geq 1$, take $\lambda_m' = 0$ for $m \neq N$ and $\lambda_N' = \sum_{m=0}^{M} \lambda_m$. This $\boldsymbol{\lambda}'$ is feasible, because the left-hand side of the constraints in (36) for $k \leq N$ writes $\sum_{m=k}^{M} \lambda_m' \binom{m}{k}t^{m-k} = \lambda_N' \binom{N}{k}t^{N-k} \geq \binom{N}{k}t^{N-k}$ (recall that $\lambda_N' \geq 1$), while for $k > N$ it is equal to 0. Moreover, $\boldsymbol{\lambda}'$ attains the same value as $\boldsymbol{\lambda}$ in the dual problem (36), because $\sum_{m=0}^{M} \lambda_m' = \sum_{m=0}^{M} \lambda_m$.

Suppose instead that $\sum_{m=0}^{M} \lambda_m < 1$. Then, at $\bar{t}_{N,k}$ the left-hand side of the constraints in (36) writes

$$\sum_{m=k}^{M} \lambda_m \binom{m}{k} \left(\bar{t}_{N,k}\right)^{m-k} \leq \text{[use (ii) above]}$$

$$\leq \left[\sum_{m=k}^{M} \lambda_m\right] \binom{N}{k} \left(\bar{t}_{N,k}\right)^{N-k}$$

$$< \left[\text{since } \sum_{m=0}^{M} \lambda_m < 1\right]$$

$$< \binom{N}{k} \left(\bar{t}_{N,k}\right)^{N-k}.$$

Hence, feasibility of $\boldsymbol{\lambda}$ implies that $1 - \epsilon(k) \leq \bar{t}_{N,k}$, since otherwise the constraint in (36) that $\sum_{m=k}^{M} \lambda_m \binom{m}{k} t^{m-k} \geq \binom{N}{k} t^{N-k} \cdot \mathbf{1}_{[0,1-\epsilon(k))}(t)$ would be violated at $\bar{t}_{N,k}$. Choose $\boldsymbol{\lambda}'$ as follows: $\lambda'_m = \lambda_m$ for $m \leq N-1$, $\lambda'_N = \sum_{m=N}^{M} \lambda_m$, and $\lambda'_m = 0$ for $m \geq N+1$. Since $\sum_{m=0}^{M} \lambda'_m = \sum_{m=0}^{M} \lambda_m$, this $\boldsymbol{\lambda}'$ attains the same value as $\boldsymbol{\lambda}$ in the dual problem (36). Moreover, because of (iii) above, it holds that $\binom{N}{k} t^{N-k} \geq \binom{m}{k} t^{m-k}$, $N \leq m \leq M$, over $[0, \bar{t}_{N,k}]$, and hence

$$\sum_{m=k}^{M} \lambda'_m \binom{m}{k} t^{m-k} = \sum_{m=k}^{N-1} \lambda_m \binom{m}{k} t^{m-k} + \left[\sum_{m=N}^{M} \lambda_m\right] \binom{N}{k} t^{N-k}$$

$$\geq \sum_{m=k}^{M} \lambda_m \binom{m}{k} t^{m-k} \tag{38}$$

over the interval $[0, \bar{t}_{N,k}]$. Using that $1 - \epsilon(k) \leq \bar{t}_{N,k}$ and that $\boldsymbol{\lambda}$ is feasible, (38) implies that $\boldsymbol{\lambda}'$ is feasible as well. This shows that $\gamma^*_M = \gamma^*_N$ for any $M \geq N$, and (37) remains proven.

To draw the conclusion of the theorem, we need to rewrite problem (36) with $M = N$ as the auxiliary variational problem (31). This is immediately done by letting

$$\xi(t) = \sum_{m=0}^{N} \lambda_m t^m,$$

and noting that $\xi(1) = \sum_{m=0}^{N} \lambda_m$ and $\frac{1}{k!} \frac{\mathrm{d}^k}{\mathrm{d}t^k} \xi(t) = \sum_{m=k}^{N} \lambda_m \binom{m}{k} t^{m-k}$.

Hence,

$$\gamma^*_N = \gamma^*, \tag{39}$$

and the proof is completed by using (34), (37), and (39) to obtain

$$\mathbb{P}^N\{V(x^*_N) > \epsilon(s^*_N)\} \leq \gamma \leq \gamma^*_N = \gamma^*.$$

□

7.2 Proof of Theorem 4

Consider problem (31) and take

$$\bar{\xi}(t) = \frac{\beta}{N+1} \sum_{m=k}^{N} t^m,$$

which gives

$$\frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}\bar{\xi}(t) = \frac{\beta}{N+1} \sum_{m=k}^{N} \binom{m}{k} t^{m-k}, \quad k = 1, \ldots, N.$$

The conclusion then follows from an application of Theorem 3, after proving that $\frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}\bar{\xi}(t) \geq \binom{N}{k} t^{N-k} \cdot \mathbf{1}_{[0,1-\epsilon(k))}(t)$ so that $\gamma^* \leq \bar{\xi}(1) = \beta$. The fact that $\frac{1}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}\bar{\xi}(t) \geq \binom{N}{k} t^{N-k} \cdot \mathbf{1}_{[0,1-\epsilon(k))}(t)$ can be proven as in the proof of Theorem 2 for $k = 0, 1, \ldots, N-1$, while the result is trivially true for $k = N$. □

8 Overview of the paper and final comments

Judging the robustness properties after a scenario solution has been found provides guarantees similar to those achievable should one know in advance that the number of support constraints never exceeds the number of support constraints that has been seen in the scenario program at hand. This is the main take-home message of this paper, and it bears profound implications. A-priori finding the largest possible number of support constraints is often an arduous endeavor. This paper suggests an alternative way to wait and judge that relieves the scenario's user from paying this effort. More importantly, in most problems an a-priori bound on the number of support constraints does not even exist and, potentially, the support set can be as large as the number of optimization variables. The pleasant message this paper delivers is that this state of things is not of obstacle to establish tight robustness evaluations, which are readily obtainable from an inspection of the found solution.

While the gap between the robustness guarantee delivered after seeing k support constraints and the guarantee that can be established by knowing in advance that k is never exceeded is minor, the results of this paper do not set this gap to zero. It is worthy of note that there are intrinsic reasons for why setting this gap to zero is impossible, and an example in dimension 2 that gives evidence of this fact is provided in Appendix 1. This epistemologically important result can be phrased that a-posteriori ascertaining that a solution is identified by only k observations—or, equivalently, that it has a "representation" of size k in terms of the data set—does not give equal robustness guarantees as obtaining a solution from a problem whose solutions are always representable by k observations. That is, simple questions are more guaranteed than simple answers.

Central to this paper is the non-degeneracy Assumption 2. This assumption isolates and singles out the essential property that must hold for the theory of this paper to

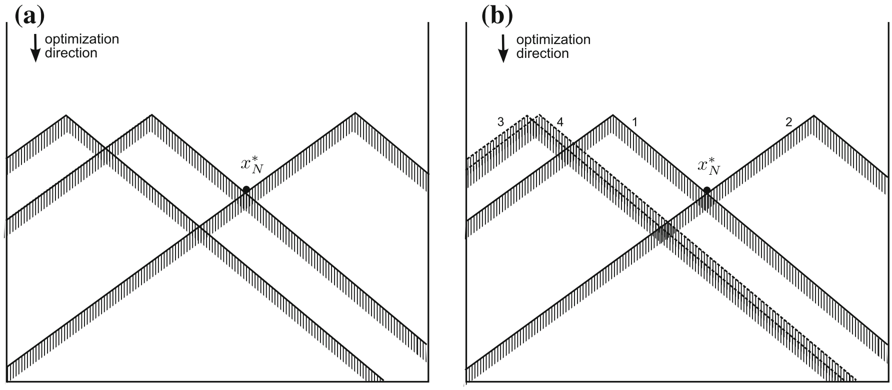

Fig. 8 Non convex problems. **a** All three constraints are of support; **b** constraints 3 and 4 are not of support

be valid. Here, we wish to provide comments and further insight on this assumption. The first part of this paper studies convex problems in finite-dimensional spaces. The fact that the space has dimension d implies that the maximum number of support constraints is d because the problem is convex. Importantly, in dimension d a non-convex problem can have more than d support constraints, and Fig. 8a displays a non-convex program in dimension $d = 2$ where all 3 constraints are of support.

In the first part of the paper, the only use we have made of convexity is to establish the bound d on the number of support constraints. However, we feel that it is important to clarify that convexity also plays a significant indirect role, and it is that Assumption 2 is normally satisfied under convexity. In convex problems, a support constraint need be active, a fact that fails to be true for non-convex problems, refer again to Fig. 8a. Hence, for convex problems, degeneracy is an anomalous condition requiring that more than d constraints meet at the solution point, which shows that Assumption 2 is quite mild. Moreover, for convex problems, one can consider to extend the theory of this paper to the cases where Assumption 2 is not satisfied by a heating and cooling procedure as it has been done in Sect. 3 of [12] in the context of an a-priori evaluation of the robustness properties. Hence, this paper virtually sets a final word for the convex finite-dimensional case.

In the general setup of Sect. 6, instead, applicability of the non-degeneracy Assumption 2 is much more delicate, and, while the main achievement of this paper is that it points to the non-degeneracy Assumption 2 as the key property for the theory to hold, the issue of identifying the classes of problems for which Assumption 2 is satisfied certainly demands more research. For reasons similar to the finite dimensional case, Assumption 2 is mild for convex infinite-dimensional problems, which is an important class in itself. Problems with convex constraints and a quasi-convex cost can be studied similarly to the convex case. Moreover, many isolated problems satisfy Assumption 2, and we have one such example in Sect. 6 where we constructed the convex hull of a set of points. On the other hand, missing to satisfy Assumption 2 cannot be seen as a pathological situation for non-convex problems. The reason lies in that support constraints need not be active for non-convex problems and thereby an anomalous concentration of constraints is not required for a non-convex problem to

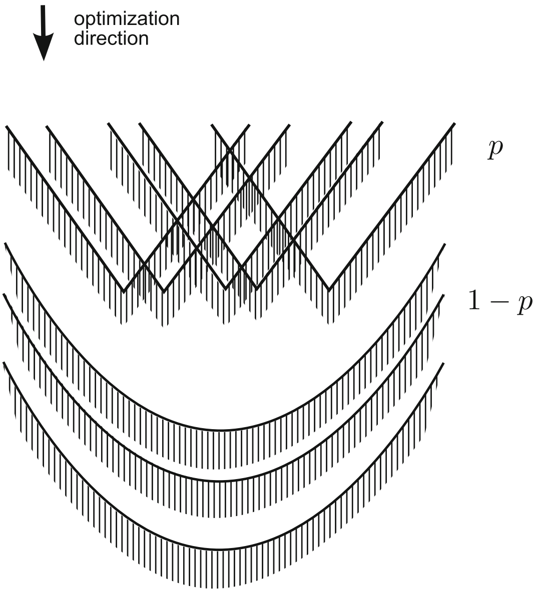

Fig. 9 A problem with V-shaped constraints and U-shaped constraints. The probability of V-shaped constraints is p and that of U-shaped constraints is $1 - p$. V-shaped constraints are above U-shaped constraints. When 1 support constraint is seen either all constraints are of the U-type or at most 1 constraint is of the V-type. In both cases, all V-shaped constraints, with at most the exception of one V-shaped constraint, are violated, and violation is at least p

be degenerate. Figure 8b shows a degenerate situation: only constraints 1 and 2 are of support, but removing constraints 3 and 4 simultaneously changes the solution. As a program for future research we indicate (i) identifying general classes of non-convex problems for which Assumption 2 hold; and (ii) developing a theory for non-convex problems for when Assumption 2 fails, which is not an easy goal since the heating and cooling approach is not effective in this context because the constraints need not group anomalously, and therefore they cannot be scattered by a heating procedure.

Finally, we want to warn the reader on a completely different issue so as to avoid a misinterpretation of our results: the results of this paper do not have a conditional validity. For example, Theorem 1 permits one to keep under control the probability $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\}$, not the conditional probability $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)|s_N^* = k\}$. Bounding $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)|s_N^* = k\}$ to a value less than 1 is in general impossible. For example, given an arbitrary $\epsilon(1) < 1$, Fig. 9 illustrates a situation where seeing 1 support constraint leads systematically to a violation of at least p, so that $\mathbb{P}^N\{V(x_N^*) > \epsilon(1)|s_N^* = 1\} = 1$ if $p > \epsilon(1)$. On the other hand, by relation $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*) \wedge s_N^* = k\} = \mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)|s_N^* = k\} \cdot \mathbb{P}^N\{s_N^* = k\}$ we see that setting $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)\}$ to a very small value, say 10^{-6}, implies that the left-hand side $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*) \wedge s_N^* = k\}$ is no more than 10^{-6} so that a large value of $\mathbb{P}^N\{V(x_N^*) > \epsilon(s_N^*)|s_N^* = k\}$ is only possible if there is a very low probability of seeing $s_N^* = k$.

Appendix 1: The guarantee valid in dimension 1 is unattainable when 1 support constraint is seen in dimension 2

Consider the optimization problem illustrated in Fig. 9 where the probabilistic mass of the V-shaped constraints is p and that of the U-shaped constraints is $1 - p$. For a given

$\epsilon(1) < p$, the event $\left\{V(x_N^*) > \epsilon(1) \ \wedge \ s_N^* = 1\right\}$ is met when all N constraints are U-shaped or when only 1 constraint is V-shaped. This event has probability $(1-p)^N + Np(1-p)^{N-1}$. The sup of this probability over the p values that satisfy $p > \epsilon(1)$ is $(1-\epsilon(1))^N + N\epsilon(1)(1-\epsilon(1))^{N-1}$. According to (3), this is the probability that $V(x_N^*) > \epsilon(1)$ in a fully-supported problem in dimension 2. This probability is bigger than the probability $(1-\epsilon(1))^N$, let us call it β, of $V(x_N^*) > \epsilon(1)$ in a fully-supported problem in dimension 1. Hence, for the problem in dimension 2 given in Fig. 9 one has to increase $\epsilon(1)$ to the value $\epsilon'(1)$ such that $(1-\epsilon'(1))^N + N\epsilon'(1)(1-\epsilon'(1))^{N-1} = \beta$ to obtain that the probability of the event $\left\{V(x_N^*) > \epsilon'(1) \ \wedge \ s_N^* = 1\right\}$ is bounded by β.

Next we show that this example constitutes a counterexample to the validity of Eq. (7). To see this, take a generic $\epsilon(1)$ and $\epsilon(0) = \epsilon(2) = 1$. The right-hand side of (7) becomes $(1-\epsilon(1))^N$, which is smaller than

$$
\begin{aligned}
&(1-\epsilon(1))^N + N\epsilon(1)(1-\epsilon(1))^{N-1} \\
&\quad = \sup_p \mathbb{P}^N\left\{V(x_N^*) > \epsilon(1) \ \wedge \ s_N^* = 1\right\} \\
&\quad = \sup_p \Big[\mathbb{P}^N\left\{V(x_N^*) > 1 \ \wedge \ s_N^* = 0\right\} \\
&\qquad + \mathbb{P}^N\left\{V(x_N^*) > \epsilon(1) \ \wedge \ s_N^* = 1\right\} + \mathbb{P}^N\left\{V(x_N^*) > 1 \ \wedge \ s_N^* = 2\right\}\Big] \\
&\quad = \sup_p \mathbb{P}^N\left\{V(x_N^*) > \epsilon(s_N^*)\right\}.
\end{aligned}
$$

Since $\mathbb{P}^N\left\{V(x_N^*) > \epsilon(s_N^*)\right\}$ is the left-hand side of (7), this contradicts (7).

Appendix 2: MATLAB code

The following MATLAB code returns $\epsilon(k)$, $k = 0, 1, \ldots, d$, for user assigned d, N, and β.

```
function out = epsilon(d,N,bet)
out = zeros(d+1,1);
for k = 0:d
  m = [k:1:N];
  aux1 = sum(triu(log(ones(N-k+1,1)*m),1),2);
  aux2 = sum(triu(log(ones(N-k+1,1)*(m-k)),1),2);
  coeffs = aux2-aux1;
  t1 = 0;
  t2 = 1;
  while t2-t1 > 1e-10
    t = (t1+t2)/2;
    val = 1 - bet/(N+1)*sum( exp(coeffs-(N-m')*log(t)) );
    if val >= 0
      t2 = t;
```

```
    else
      t1 = t;
    end
  end
  out(k+1) = 1-t1;
end
```

Appendix 3: Examples of solutions of Eq. (18)

Fully-supported problems

One can easily verify that the following F_k's satisfy Eq. (18):

$$F_k(v) = \begin{cases} 0, & v < 1 \\ 1, & v \geq 1 \end{cases}, \quad k = 0, 1, \ldots, d-1, \tag{40}$$

$$F_d(v) = \begin{cases} 0, & v < 0 \\ v^d, & 0 \leq v \leq 1 \\ 1, & v > 1. \end{cases} \tag{41}$$

These are the F_k's of fully-supported problems. The fact that for fully-supported problems $F_d(v)$ is as in (41) is proven in [12]. To show instead the validity of (40), argue as follows. For $0 \leq k < d$, relation

$$V(x_k^*) = 1 \tag{42}$$

holds with probability one since, otherwise, complementing the set of k constraints with $d - k + 1$ other constraints that are satisfied by x_k^*, which would be an event with nonzero probability, leads to a total set of $d + 1$ constraints among which fewer than d are of support, so contradicting the fully-supportedness assumption. Thus, the measures corresponding to $F_k, k = 0, 1, \ldots, d-1$, concentrate in 1. Further, (40) claims that the mass in 1 is equal to 1, which corresponds to say that the number of support constraints is equal to k with probability one. For $k = 0$ this is obvious. For $0 < k < d$, this is also true because, if less than k constraints were of support, then at least one of the constraints would be satisfied by the solution generated when only the other constraints are in place, a fact that happens with probability zero since the violation of the solution generated when only the other constraints are in place is equal to 1 as shown in (42).

Two examples of problems in dimension $d = 2$

Consider the problem

$$\min_{x_1 \geq 0, x_2 \geq 0} x_2$$
$$\text{subject to: } |x_1 - \delta| \leq x_2,$$

where δ is uniform in [0, 1]. As it can be easily verified, this problem is fully-supported so that, according to (40),(41), its F_k's are

$$F_0(v) = F_1(v) = \begin{cases} 0, & v < 1 \\ 1, & v \geq 1 \end{cases}, \quad F_2(v) = \begin{cases} 0, & v < 0 \\ v^2, & 0 \leq v \leq 1 \\ 1, & v > 1 \end{cases}.$$

Consider instead problem

$$\min_{x_1 \geq 0, x_2 \geq 0} \; x_1 + x_2$$
$$\text{subject to: } x_{\delta_1} \geq \delta_2,$$

where δ_1 takes value 1 or 2 with probability 0.5 each, and δ_2 is independent of δ_1 and is uniformly distributed on [0, 1]. This problem is not fully-supported and a simple calculation shows that

$$F_0(v) = \begin{cases} 0, & v < 1 \\ 1, & v \geq 1 \end{cases}, \quad F_1(v) = \begin{cases} 0, & v < 0.5 \\ 2v - 1, & 0.5 \leq v \leq 1 \\ 1, & v > 1 \end{cases},$$
$$F_2(v) = \begin{cases} 0, & v < 0 \\ 0.5v^2, & 0 \leq v \leq 1 \\ 0.5, & v > 1 \end{cases}.$$

These F_k's are another solution of Eq. (18).

References

1. Alamo, T., Tempo, R., Camacho, E.: Randomized strategies for probabilistic solutions of uncertain feasibility and optimization problems. IEEE Trans. Autom. Control **54**(11), 2545–2559 (2009)
2. Alamo, T., Tempo, R., Luque, A., Ramirez, D.R.: Randomized methods for design of uncertain systems: sample complexity and sequential algorithms. Automatica **51**, 160–172 (2015)
3. Anderson, E., Nash, P.: Linear programming in infinite-dimensional spaces: theory and applications. Wiley, New York (1987)
4. Ben-Tal, A., Nemirovski, A.: On safe tractable approximations of chance-constrained linear matrix inequalities. Math. Oper. Res. **34**(1), 1–25 (2009)
5. Bertsimas, D., Thiele, A.: Robust and data-driven optimization: modern decision-making under uncertainty. In: Tutorials on Operations Research. INFORMS (2006)
6. Borwein, J., Lewis, A.: Convex analysis and nonlinear optimization. Springer, New York (2005)
7. Calafiore, G., Campi, M.: Uncertain convex programs: randomized solutions and confidence levels. Math. Program. **102**(1), 25–46 (2005). doi:10.1007/s10107-003-0499-y
8. Calafiore, G., Campi, M.: The scenario approach to robust control design. IEEE Trans. Autom. Control **51**(5), 742–753 (2006)
9. Campi, M.: Classification with guaranteed probability of error. Mach. Learn. **80**, 63–84 (2010)
10. Campi, M., Calafiore, G., Garatti, S.: Interval predictor models: identification and reliability. Automatica **45**(2), 382–392 (2009). doi:10.1016/j.automatica.2008.09.004
11. Campi, M., Carè, A.: Random convex programs with L1-regularization: sparsity and generalization. SIAM J. Control Optim. **51**(5), 3532–3557 (2013)
12. Campi, M., Garatti, S.: The exact feasibility of randomized solutions of uncertain convex programs. SIAM J. Optim. **19**(3), 1211–1230 (2008). doi:10.1137/07069821X

13. Campi, M., Garatti, S.: A sampling-and-discarding approach to chance-constrained optimization: feasibility and optimality. J. Optim. Theory Appl. **148**(2), 257–280 (2011)
14. Campi, M., Garatti, S., Prandini, M.: The scenario approach for systems and control design. Annu. Rev. Control **33**(2), 149–157 (2009). doi:10.1016/j.arcontrol.2009.07.001
15. Carè, A., Garatti, S., Campi, M.: FAST—fast algorithm for the scenario technique. Oper. Res. **62**(3), 662–671 (2014)
16. Carè, A., Garatti, S., Campi, M.: Scenario min–max optimization and the risk of empirical costs. SIAM J. Optim. **25**(4), 2061–2080 (2015)
17. Crespo, L., Giesy, D., Kenny, S.: Interval predictor models with a formal characterization of uncertainty and reliability. In: Proceedings of the 53rd IEEE Conference on Decision and Control (CDC), pp. 5991–5996. Los Angeles, CA, USA (2014)
18. Delage, E., Ye, Y.: Distributionally robust optimization under moment uncertainty with application to data-driven problems. Oper. Res. **58**(3), 596–612 (2010)
19. Dentcheva, D.: Optimization models with probabilistic constraints. In: Calafiore, G., Dabbene, F. (eds.) Probabilistic and Randomized Methods for Design Under Uncertainty. Springer, London (2006)
20. Esfahani, P., Sutter, T., Lygeros, J.: Performance bounds for the scenario approach and an extension to a class of non-convex programs. IEEE Trans. Autom. Control **60**(1), 46–58 (2015)
21. Farias, V., Jagabathula, S., Shah, D.: A nonparametric approach to modeling choice with limited data. Manag. Sci. **59**(2), 305–322 (2013)
22. Garatti, S., Campi, M.: Modulating robustness in control design: principles and algorithms. IEEE Control Syst. Mag. **33**(2), 36–51 (2013). doi:10.1109/MCS.2012.2234964
23. Grammatico, S., Zhang, X., Margellos, K., Goulart, P., Lygeros, J.: A scenario approach for non-convex control design. IEEE Trans. Autom. Control **61**(2), 334–345 (2016)
24. Grant, M., Boyd, S.: CVX: Matlab Software for Disciplined Convex Programming, Version 2.1. http://cvxr.com/cvx (2014)
25. Hong, L., Hu, Z., Liu, G.: Monte Carlo methods for value-at-risk and conditional value-at-risk: a review. ACM Trans. Model. Comput. Simul. **24**(4), 22:1–22:37 (2014)
26. Krishnamoorthy, K., Mathew, T.: Statistical Tolerance Regions. Wiley, Hoboken (2009)
27. Lasserre, J.: Moments, Positive Polynomials and Their Applications. Imperial College Press, London (2009)
28. Lei, J., Robins, J., Wasserman, L.: Distribution-free prediction sets. J. Am. Stat. Assoc. **108**(501), 278–287 (2013)
29. Lofberg, J.: Yalmip: a toolbox for modeling and optimization in matlab. In: Proceedings of the CACSD Conference, pp. 284–289. Taipei, Taiwan (2004)
30. Lucchetti, R.: Convexity and Well-Posed Problems. Springer, New York (2006)
31. Luedtke, J., Ahmed, S.: A sample approximation approach for optimization with probabilistic constraints. SIAM J. Optim. **19**, 674–699 (2008). doi:10.1137/070702928
32. Luedtke, J., Ahmed, S., Nemhauser, G.: An integer programming approach for linear programs with probabilistic constraints. Math. Program. **122**(2), 247–272 (2010)
33. Luenberger, D.: Optimization by Vector Space Methods. Wiley, New York (1969)
34. Margellos, K., Goulart, P., Lygeros, J.: On the road between robust optimization and the scenario approach for chance constrained optimization problems. IEEE Trans. Autom. Control **59**(8), 2258–2263 (2014)
35. Margellos, K., Prandini, M., Lygeros, J.: On the connection between compression learning and scenario based single-stage and cascading optimization problems. IEEE Trans. Autom. Control **60**(10), 2716–2721 (2015)
36. de Mello, T.H., Bayraksan, G.: Monte Carlo sampling-based methods for stochastic optimization. Surv. Oper. Res. Manag. Sci. **19**(1), 56–85 (2014)
37. Nemirovski, A.: On safe tractable approximations of chance constraints. Eur. J. Oper. Res. **219**, 707–718 (2012)
38. Nemirovski, A., Shapiro, A.: Convex approximations of chance constrained programs. SIAM J. Optim. **17**(4), 969–996 (2006). doi:10.1137/050622328
39. Nemirovski, A., Shapiro, A.: Scenario approximations of chance constraints. In: Calafiore, G., Dabbene, F. (eds.) Probabilistic and Randomized Methods for Design Under Uncertainty. Springer, London (2006)
40. Pagnoncelli, B., Ahmed, S., Shapiro, A.: Sample average approximation method for chance constrained programming: theory and applications. J. Optim. Theory Appl. **142**(2), 399–416 (2009)

41. Pagnoncelli, B., Reich, D., Campi, M.: Risk-return trade-off with the scenario approach in practice: a case study in portfolio selection. J. Optim. Theory Appl. **155**(2), 707–722 (2012)
42. Pagnoncelli, B., Vanduffel, S.: A provisioning problem with stochastic payments. Eur. J. Oper. Res. **221**(2), 445–453 (2012)
43. Petersen, I.R., Tempo, R.: Robust control of uncertain systems: classical results and recent developments. Automatica **50**, 1315–1335 (2014)
44. Prèkopa, A.: Stochastic Programming. Kluwer, Boston (1995)
45. Schildbach, G., Fagiano, L., Morari, M.: Randomized solutions to convex programs with multiple chance constraints. SIAM J. Optim. **23**(4), 2479–2501 (2013)
46. Shapiro, A., Dentcheva, D., Ruszczyński, A.: Lectures on Stochastic Programming: Modeling and Theory. MPS-SIAM, Philadelphia (2009)
47. Shiryaev, A.: Probability. Springer, New York (1996)
48. Tempo, R., Calafiore, G., Dabbene, F.: Randomized Algorithms for Analysis and Control of Uncertain Systems, 2nd edn. Springer, London (2013)
49. Thiele, A.: Robust stochastic programming with uncertain probabilities. IMA J. Manag. Math. **19**(3), 289–321 (2008). doi:10.1093/imaman/dpm011
50. Vayanos, P., Kuhn, D., Rustem, B.: A constraint sampling approach for multistage robust optimization. Automatica **48**(3), 459–471 (2012)
51. Welsh, J., Kong, H.: Robust experiment design through randomisation with chance constraints. In: Proceedings of the 18th IFAC World Congress. Milan, Italy (2011)
52. Welsh, J., Rojas, C.: A scenario based approach to robust experiment design. In: Proceedings of the 15th IFAC Symposium on System Identification. Saint-Malo, France (2009)
53. Zhang, X., Grammatico, S., Schildbach, G., Goulart, P., Lygeros, J.: On the sample size of random convex programs with structured dependence on the uncertainty. Automatica **60**, 182–188 (2015)
54. Zymler, S., Kuhn, D., Rustem, B.: Distributionally robust joint chance constraints with second-order moment information. Math. Program. **137**(1–2), 167–198 (2013)