Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

FAST—Fast Algorithm for the Scenario Technique

Algo Carè, Simone Garatti, Marco C. Campi

Operations Research
Vol. 62, No. 3, May–June 2014, pp. 662–671
ISSN 0030-364X (print) | ISSN 1526-5463 (online)

http://dx.doi.org/10.1287/opre.2014.1257
© 2014 INFORMS

FAST—Fast Algorithm for the Scenario Technique

Algo Carè
Department of Electrical and Electronic Engineering, The University of Melbourne, Parkville, VIC 3052, Australia,
algo.care@unimelb.edu.au

Simone Garatti
Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milano, Italia,
simone.garatti@polimi.it

Marco C. Campi
Dipartimento di Ingegneria dell'Informazione, Università di Brescia, 25123 Brescia, Italia,
marco.campi@ing.unibs.it

The scenario approach is a recently introduced method to obtain feasible solutions to chance-constrained optimization problems based on random sampling. It has been noted that the sample complexity of the scenario approach rapidly increases with the number of optimization variables and this may pose a hurdle to its applicability to medium- and large-scale problems. We here introduce the Fast Algorithm for the Scenario Technique, a variant of the scenario optimization algorithm with reduced sample complexity.

Subject classifications: stochastic programming; chance-constrained optimization; randomized algorithms; sample-based methods; scenario approach.
Area of review: Optimization.
History: Received March 2012; revisions received April 2013, September 2013; accepted December 2013. Published online in *Articles in Advance* April 28, 2014.

1. Introduction

Consider a cost function $f(x,\delta)$, where $x \in \mathcal{X} \subseteq \mathbb{R}^d$ is a decision variable and $\delta \in \Delta$ is a random variable, distributed according to $\mathbb{P}$, which describes uncertainty. Throughout, $f(x,\delta)$ is assumed to be *convex* in x, whereas its dependence on δ is arbitrary, and $\mathcal{X}$ is a *convex* set. This paper considers the following chance-constrained problem:

$$\begin{aligned} &\min_{x\in\mathcal{X}\subseteq\mathbb{R}^d,\, l\in\mathbb{R}} \quad l \\ &\text{subject to:}\quad \mathbb{P}\{f(x,\delta)\leqslant l\}\geqslant 1-\epsilon. \end{aligned} \tag{1}$$

In (1), an x has to be found so as to minimize l, which is an upper bound on function $f(x,\delta)$ that holds with probability $1-\epsilon$.

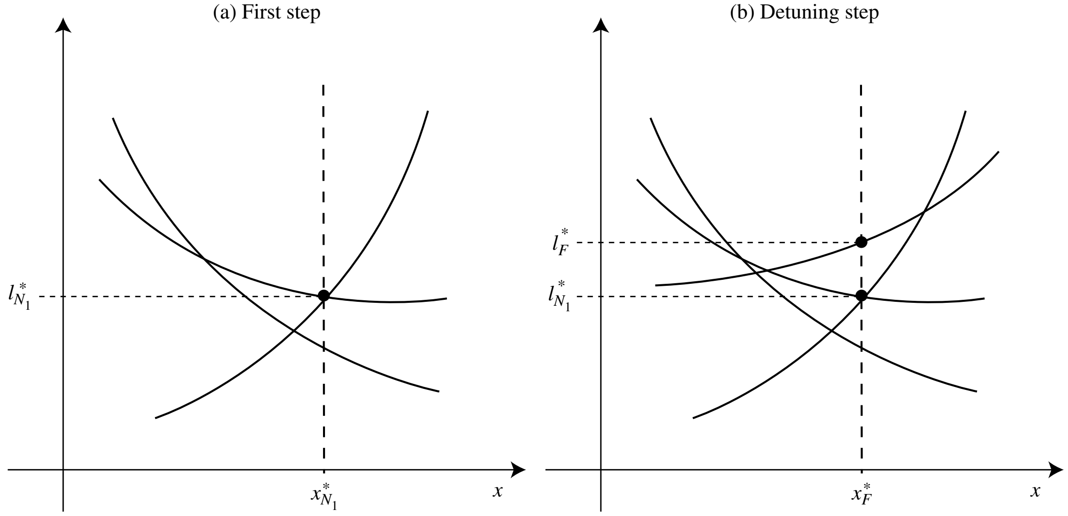
Chance-constrained problems are quite popular in stochastic optimization, see e.g., Prékopa (1995, 2003), Dentcheva (2006), Shapiro et al. (2009), and it is well known that they are in general difficult to solve. For this reason, sample-based approximations of chance-constrained problems have been considered. Letting $\delta^{(1)},\dots,\delta^{(N)}$ be N instances, or scenarios, of the uncertainty variable δ independently sampled according to the probability measure $\mathbb{P}$, a sample-based approximation to (1), the so-called "scenario program," can be written as

$$\begin{aligned} &\min_{x\in\mathcal{X}\subseteq\mathbb{R}^d,\, l\in\mathbb{R}} \quad l \\ &\text{subject to:}\quad f(x,\delta^{(i)})\leqslant l, \quad i=1,\dots,N. \end{aligned} \tag{2}$$

This program allows one to find a feasible solution to problem (1), while also heuristically pursuing the achievement of a satisfactory performance by optimizing over a finite sample of δ values. The feasibility of the solution of (2) with respect to the probabilistic constraint in (1) has been theoretically studied in Calafiore and Campi (2005, 2006), Campi and Garatti (2008), Luedtke and Ahmed (2008), Alamo et al. (2009, 2010), Campi and Garatti (2011), Garatti and Campi (2013). See Bertsimas and Brown (2009), Pagnoncelli et al. (2009), Chen et al. (2010), Hong et al. (2011), Pagnoncelli et al. (2012) for some applications of the scenario program. Programs incorporating a sample of the uncertainty variable also arise in data-driven optimization where optimization is based on historical data; see, e.g., Bertsimas and Thiele (2006), Delage and Ye (2010).

Letting x_N^* and l_N^* be the solution and the optimal value of (2), Theorem 1 in Campi and Garatti (2008) proves that, if N is suitably chosen, relation $f(x_N^*,\delta)\leqslant l_N^*$ holds with high probability $1-\epsilon$ with respect to δ, that is, (x_N^*, l_N^*) is feasible for problem (1). Thus, l_N^* is a guaranteed cost with probability $1-\epsilon$ when decision x_N^* is applied. It turns out that this "suitable N" is inversely proportional to ϵ and is proportional to d, the number of components in the decision variable x, i.e., N scales as $(1/\epsilon)\cdot d$. However, as noted in Nemirovski and Shapiro (2006), Oishi (2007), this dependence on ϵ and d may result in too many scenarios for large-scale problems where d takes large values. This fact represents a difficulty for the use of the method, and

Figure 1. Illustration of FAST.



Note. Each line represents a function $f(x, \delta^{(i)})$.

undermines its applicability when scenarios are observations and are therefore a limited and valuable resource. In the present paper a novel algorithm called FAST (Fast Algorithm for the Scenario Technique) is introduced that gets around this difficulty. FAST returns a solution (x^*_F, l^*_F), which is still feasible for problem (1), with a sample complexity N that exhibits a dependence on ϵ and d of the form $1/\epsilon + d$. Thereby, the FAST algorithm significantly reduces the sample complexity in large-scale optimization problems.

1.1. The Idea Behind FAST

FAST constructs a solution in two steps. First, a moderate number N_1 of scenarios $\delta^{(i)}$ are considered and problem (2) with $N = N_1$ is solved so generating a decision $x^*_{N_1}$ and an optimal value $l^*_{N_1}$; refer to Figure 1(a). This first step is carried out at a low computational effort because of the moderate number N_1 of scenarios involved. On the other hand, $f(x^*_{N_1}, \delta) \leqslant l^*_{N_1}$ is not guaranteed with the desired probability $1-\epsilon$ since N_1 is too low for this guarantee to hold. Then, a *detuning* step is started where N_2 additional scenarios are sampled and the smallest value l^*_F such that $f(x^*_{N_1}, \delta^{(i)}) \leqslant l^*_F$, $i = 1, \ldots, N_1 + N_2$, is computed; see Figure 1(b). The algorithm returns the solution $x^*_F = x^*_{N_1}$ and the value l^*_F. The theory in §3 shows that $f(x^*_F, \delta) \leqslant l^*_F$ holds with the desired probability $1-\epsilon$. In this construction, N_1 and N_2 scale as d and $1/\epsilon$, respectively, leading to an overall number of scenarios $N = N_1 + N_2$ that is typically much smaller than that required by the "classical" scenario approach. Moreover, choosing a small ϵ does not affect N_1 and only results in a large N_2 value, which corresponds to having many scenarios in the computationally low-demanding detuning step.

1.2. Structure of the Paper

The remainder of the paper is organized as follows. Section 2 provides background material on the classical scenario approach. In §3, the FAST algorithm is formally presented and its main theoretical properties are given, followed by a discussion on the practical use of the algorithm. Section 4 extends FAST to a setup with general convex constraints, instead of constraints of the form $f(x, \delta) \leqslant l$ as in (1). A simulation example is presented in §5, and all proofs are in §6.

2. Background Material on the Scenario Approach

Throughout, we assume that problem (2) has a unique solution for any N and any values of $\delta^{(1)}, \ldots, \delta^{(N)}$. Although this assumption can be relaxed, see, e.g., the discussion in section 2.1 of Campi and Garatti (2008), it is here made to streamline the presentation.

Definition 1 (Violation Probability). The *violation probability* of a given $(x, l) \in \mathscr{X} \times \mathbb{R}$ is defined as

$$V(x, l) = \mathbb{P}\{\delta\colon f(x, \delta) > l\}.$$

In words, $V(x, l)$ is the probability with which x attains a cost larger than l.

The violation probability $V(x^*_N, l^*_N)$, where (x^*_N, l^*_N) is the solution of (2), has been studied in Campi and Garatti (2008). The variables x^*_N and l^*_N are random since they depend on $\delta^{(1)}, \ldots, \delta^{(N)}$. Thus, a statement like $V(x^*_N, l^*_N) > \epsilon$ has a probabilistic nature, i.e., $V(x^*_N, l^*_N) > \epsilon$ holds with a certain probability. An exact quantification of the probability that $V(x^*_N, l^*_N) > \epsilon$ is given in Theorem 1 of Campi and Garatti (2008), where the following result is proved:

$$\mathbb{P}^N\{\delta^{(1)}, \ldots, \delta^{(N)}\colon V(x^*_N, l^*_N) > \epsilon\} \leqslant \sum_{i=0}^{d} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i}. \tag{3}$$

Equation (3) shows that the probability of seeing a "bad" sample $\delta^{(1)}, \ldots, \delta^{(N)}$ such that $V(x_N^*, l_N^*) > \epsilon$ is no more than $\sum_{i=0}^{d} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i}$. A truly remarkable fact is that the right-hand side of (3) is a bound valid irrespective of $\mathbb{P}$, so that an application of the result in (3) does not require knowledge of probability $\mathbb{P}$. Moreover, result (3) is not improvable since the inequality $\leqslant$ in (3) becomes an equality $=$ for a whole class of problems, the so-called *fully-supported* problems; see Definition 3 in Campi and Garatti (2008). See Campi and Garatti (2008) for more details, and a discussion on the use of (3).

The right-hand side of (3) is the so-called incomplete beta function ratio; see, e.g., Gupta and Nadarajah (2004). For brevity, in the sequel we shall use the notation

$$B_\epsilon^{N,d} = \sum_{i=0}^{d} \binom{N}{i} \epsilon^i (1-\epsilon)^{N-i}. \qquad (4)$$

When using (3), one fixes a (very small) confidence parameter β and finds the smallest integer N such that $B_\epsilon^{N,d} \leqslant \beta$. Because of (3), this N entails that $\mathbb{P}^N\{V(x_N^*, l_N^*) > \epsilon\} \leqslant \beta$, so that solving (2) returns a solution such that $V(x_N^*, l_N^*) \leqslant \epsilon$ holds with (very high) confidence $1-\beta$.

In Alamo et al. (2010), an explicit formula for N so that $B_\epsilon^{N,d} \leqslant \beta$ is shown to be

$$N \geqslant \frac{e}{e-1} \frac{1}{\epsilon} \left(d + \ln \frac{1}{\beta} \right). \qquad (5)$$

This N scales logarithmically in $1/\beta$, so that β can be made so small, say 10^{-9}, that it can be neglected in practice without increasing N too much. On the other hand, the dependence on d and ϵ is of the form $(1/\epsilon) \cdot d$.

3. FAST

The FAST algorithm is given in §3.1. In §3.2 theoretical results are presented and a discussion about the practical use of FAST follows in §3.3.

3.1. The FAST Algorithm

- INPUT:
  - $\epsilon \in]0, 1[$, violation parameter;
  - $\beta \in]0, 1[$, confidence parameter;
  - N_1, an integer such that $N_1 \geqslant d + 1$.

1. Compute the smallest integer N_2 such that

$$N_2 \geqslant \frac{\ln \beta - \ln B_\epsilon^{N_1, d}}{\ln(1-\epsilon)}, \qquad (6)$$

where $B_\epsilon^{N_1,d}$ is as in Equation (4).

2. Sample $N_1 + N_2$ independent constraints $\delta^{(1)}, \ldots, \delta^{(N_1)}, \delta^{(N_1+1)}, \ldots, \delta^{(N_1+N_2)}$, according to $\mathbb{P}$.

3. Solve problem (2) with $N = N_1$; let $(x_{N_1}^*, l_{N_1}^*)$ be the solution.

4. (Detuning step) Compute

$$l_F^* = \max_{i=1,\ldots,N_1+N_2} f(x_{N_1}^*, \delta^{(i)}).$$

- OUTPUT:
  - $(x_F^*, l_F^*) := (x_{N_1}^*, l_F^*)$.

3.2. Theoretical Results

The violation probability $V(x_F^*, l_F^*)$ is a random variable that depends on the sample $\delta^{(1)}, \ldots, \delta^{(N_1+N_2)}$. The following theorem bounds the probability that $V(x_F^*, l_F^*) > \epsilon$.

THEOREM 1. *The following relation holds*

$$\mathbb{P}^{N_1+N_2}\{V(x_F^*, l_F^*) > \epsilon\} \leqslant (1-\epsilon)^{N_2} \cdot B_\epsilon^{N_1,d}. \qquad (7)$$

The proof of Theorem 1 is given in §6.

It is a fact that the bound on the right-hand side of (7) is not improvable. Indeed, relation (7) holds with equality for the whole class $\mathscr{C}$ of optimization problems specified in the following definition.

DEFINITION 2. Problem (1) is said to be in class $\mathscr{C}$ if

(i) its sample-based approximation (2) is fully supported according to Definition 3 in Campi and Garatti (2008) with probability 1 for all $N \geqslant d+1$;

(ii) $\forall x \in \mathscr{X}, \forall l \in \mathbb{R},\ \mathbb{P}\{\delta\colon f(x,\delta) = l\} = 0$.

THEOREM 2. *Relation*

$$\mathbb{P}^{N_1+N_2}\{V(x_F^*, l_F^*) > \epsilon\} = (1-\epsilon)^{N_2} \cdot B_\epsilon^{N_1,d} \qquad (8)$$

holds whenever (1) *is in class* $\mathscr{C}$.

For a proof see §6.

Note now that Equation (6) is equivalent to

$$(1-\epsilon)^{N_2} \cdot B_\epsilon^{N_1,d} \leqslant \beta,$$

so that Theorem 1 implies that

$$\mathbb{P}^{N_1+N_2}\{V(x_F^*, l_F^*) > \epsilon\} \leqslant \beta.$$

On the other hand, since N_2 is the smallest integer such that (6) holds, any $N_2' < N_2$ gives

$$(1-\epsilon)^{N_2'} \cdot B_\epsilon^{N_1,d} > \beta,$$

and, in light of Theorem 2, this implies that

$$\mathbb{P}^{N_1+N_2'}\{V(x_F^*, l_F^*) > \epsilon\} > \beta$$

whenever (1) is in class $\mathscr{C}$. This discussion establishes the following main theorem.

THEOREM 3. *It holds that*

$$\mathbb{P}^{N_1+N_2}\{V(x_F^*, l_F^*) > \epsilon\} \leqslant \beta. \qquad (9)$$

Moreover, the value N_2 *given in step* 1 *of the FAST algorithm cannot be improved in the sense that there are problems for which no* N_2 *smaller than that given in step* 1 *of the FAST algorithm makes* (9) *true.*

3.3. Discussion

In the FAST algorithm, the user solves problem (2) with N_1 constraints, and computes N_2 through (6). The number N_1 is decided by the user, whereas N_2 depends on N_1, ϵ, and β. In this section, guidelines on how to select N_1, and a handier formula for N_2, are provided. Additional discussion on some advantages of using FAST is also given.

Selection of N_1. Computational reasons suggest that N_1 should be chosen as small as possible, for, otherwise, step 3 in the FAST algorithm becomes expensive, so losing the advantages of using FAST. On the other hand, if N_1 is too small, $x^*_{N_1}$ is poorly selected, and this in turn leads to a large cost value l^*_F after that the detuning step 4 in FAST is carried out. As a rule of thumb out of empirical experience, we suggest to take $N_1 = 20d$. Notice that the theoretical results in Theorem 3 remain valid for any choice of N_1.

A Handier Formula for N_2. In step 1 of the FAST algorithm, Equation (6) can be substituted by the handier formula

$$N_2 \geqslant \frac{1}{\epsilon}\ln\frac{1}{\beta}. \tag{10}$$

In fact,

$$\frac{\ln\beta - \ln B_\epsilon^{N_1,d}}{\ln(1-\epsilon)} \leqslant \frac{\ln\beta}{\ln(1-\epsilon)} \leqslant \frac{1}{\epsilon}\ln\frac{1}{\beta},$$

showing that an N_2 satisfying (10) also satisfies (6). (10) is easier to apply than (6) since (6) also involves computing the incomplete beta function ratio $B_\epsilon^{N_1,d}$.

Advantages with Using FAST.

Reduced sample size requirements. The FAST algorithm provides a cheaper way to find solutions to medium- and large-scale problems than the classical scenario approach. Indeed, one can choose $N_1 = Kd$, where K is a user-selected number normally set to 20, while, using (10), N_2 can be taken as the first integer bigger than or equal to $(1/\epsilon)\ln(1/\beta)$. Hence, a simple formula to estimate the overall number of scenarios needed with FAST is

$$Kd + \frac{1}{\epsilon}\ln\frac{1}{\beta}.$$

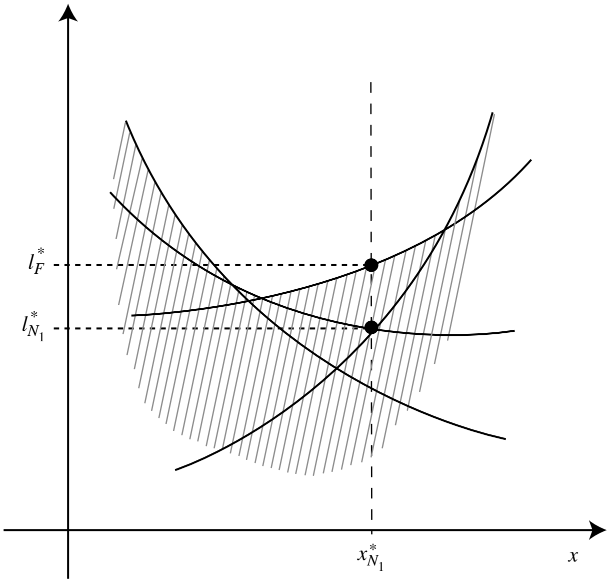
A comparison with the evaluation in (5)

$$\frac{e}{e-1}\frac{1}{\epsilon}\left(d + \ln\frac{1}{\beta}\right)$$

applicable to the classical scenario approach shows the key point that, with FAST, the critical multiplicative dependence on $(1/\epsilon)\cdot d$ is replaced by an additive dependence on $1/\epsilon$ and d.

Possibility to reduce ϵ to small values. The detuning step 4 of FAST is a simple one-dimensional maximization problem. Therefore, running step 4 with a large N_2 can be done at low computational effort so that ϵ can be reduced to values much smaller than with the classical scenario approach.

Figure 2. Comparison between FAST and the classical scenario approach.



Suboptimality of FAST. Figure 2 represents the solution obtained using FAST. In the figure, $l^*_{N_1}$ is the cost value for the problem with N_1 scenarios, and l^*_F is the cost value after the introduction of N_2 extra scenarios in the detuning step. In white is the region above all cost functions $f(x,\delta^{(i)})$, $i = 1,\dots,N_1+N_2$. An inspection of Figure 2 reveals that the white region contains a part that outperforms l^*_F. Although the classical approach introduces additional scenarios beyond N_1+N_2 to achieve the same level of violation as FAST, so that the number of scenarios it uses is $N \geqslant N_1+N_2$, still its solution can fall in the part of the white region that outperforms l^*_F. However, letting l^*_N be the cost value of the classical scenario approach, it certainly holds that $l^*_F - l^*_N < l^*_F - l^*_{N_1}$. Consequently, the user has a simple way to evaluate the maximum possible suboptimality of FAST compared with the classical scenario approach as given by $l^*_F - l^*_{N_1}$. Empirical evidence shows that l^*_F and l^*_N are often close to each other so that suboptimality is negligible.

A Comparison with Validation Set Methods. The approach of this paper of using a second set of N_2 constraints bears similarities with validation set methods, where a validation set is used to evaluate the feasibility level of a solution. Validation sets are often employed in sequential algorithms for solving convex and nonconvex optimization problems; see, e.g., Koltchinskii et al. (2000), Oishi (2007), Wada and Fujisaki (2007), Alamo et al. (2009), Calafiore et al. (2011). However, two differences between validation set methods and FAST must be highlighted. First, with FAST, the optimal value is updated based on the new N_2 constraints, as opposed to simply validating a given solution. Second, Theorem 2 combines the feasibility of the original solution based on N_1 constraints with the additional information carried by the extra N_2 constraints, rather than more simply validating the solution with the additional N_2 constraints.

4. A More General Set-Up

In previous sections, FAST was applied to problems with constraints in the specific form $f(x,\delta)\leqslant l$. Here, more general constraints are considered. Given a constant vector $c\in\mathbb{R}^{d+1}$, a convex and closed set $\mathcal{S}\subseteq\mathbb{R}^{d+1}$, and a family of convex and closed sets $\mathcal{Z}_\delta$ parameterized in the uncertainty variable δ, consider the following constrained convex scenario program:

$$\begin{aligned}&\min_{z\in\mathcal{S}\subseteq\mathbb{R}^{d+1}} \quad c^Tz\\ &\text{subject to:}\quad z\in\bigcap_{i=1,\dots,N}\mathcal{Z}_{\delta^{(i)}},\end{aligned}\tag{11}$$

where $\delta^{(1)},\dots,\delta^{(N)}$ are instances of δ independently sampled according to the probability measure $\mathbb{P}$. Problem (11) is meant as a sample-based approximation to the chance-constrained problem:

$$\begin{aligned}&\min_{z\in\mathcal{S}\subseteq\mathbb{R}^{d+1}} \quad c^Tz\\ &\text{subject to:}\quad \mathbb{P}\{z\in\mathcal{Z}_\delta\}\geqslant 1-\epsilon.\end{aligned}$$

Since every convex program can be rewritten so as it has a linear objective, see, e.g., Boyd and Vandenberghe (2004), linearity of the objective function in (11) is without loss of generality. Also, note that (2) is a particular case of (11) with $z=(x,l)$, $\mathcal{S}=\mathcal{X}\times\mathbb{R}$, $\mathcal{Z}_\delta=\{(x,l)\colon f(x,\delta)\leqslant l\}$, and $c^T=(0,0,\dots,0,1)$. The notion of violation probability of (x,l) given in Definition 1 is extended to the present context as follows.

Definition 3 (Violation Probability). The *violation probability* of a given point $z\in\mathcal{S}$ is defined as

$$V(z)=\mathbb{P}\{\delta\colon z\notin\mathcal{Z}_\delta\}.$$

In the following, we assume that, for any N and any values of $\delta^{(1)},\dots,\delta^{(N)}$, problem (11) is feasible, its feasibility domain has nonempty interior, and the solution of (11) exists and is unique. Moreover, it is assumed that the user knows a "robustly feasible" point.

Assumption 1. *A point $\bar z\in(\bigcap_{\delta\in\Delta}\mathcal{Z}_\delta)\cap\mathcal{S}$ is known to the user.*[1]

In §4.1 the generalized FAST algorithm is given. The main theoretical result for the generalized FAST algorithm is presented in §4.2, followed by a brief discussion in §4.3.

4.1. Generalized FAST Algorithm

- INPUT:
  - $\epsilon\in\,]0,1[$, violation parameter;
  - $\beta\in\,]0,1[$, confidence parameter;
  - N_1, an integer such that $N_1\geqslant d+1$;
  - $\bar z\in(\bigcap_{\delta\in\Delta}\mathcal{Z}_\delta)\cap\mathcal{S}$, a robustly feasible point.

1. Compute the smallest integer N_2 such that

$$N_2\geqslant\frac{\ln\beta-\ln B_\epsilon^{N_1,d}}{\ln(1-\epsilon)},\tag{12}$$

where $B_\epsilon^{N_1,d}$ is as in Equation (4).

2. Sample N_1+N_2 independent constraints $\delta^{(1)},\dots,\delta^{(N_1)}$, $\delta^{(N_1+1)},\dots,\delta^{(N_1+N_2)}$, according to $\mathbb{P}$.

3. Solve problem (11) with $N=N_1$; let $z^*_{N_1}$ be the solution.

4. (Detuning step) Let $\hat z[\alpha]:=(1-\alpha)z^*_{N_1}+\alpha\bar z$, $\alpha\in[0,1]$, i.e., $\hat z[\alpha]$ describes the line segment connecting $z^*_{N_1}$ with $\bar z$. Compute the solution α^* to the problem

$$\begin{aligned}&\min_{\alpha\in[0,1]} \quad c^T\hat z[\alpha]\\ &\text{subject to:}\quad \hat z[\alpha]\in\bigcap_{i=N_1+1}^{N_1+N_2}\mathcal{Z}_{\delta^{(i)}}.\end{aligned}\tag{13}$$

- OUTPUT:
  - $z^*_F:=\hat z[\alpha^*]$.

4.2. Theoretical Results

The violation of the solution z^*_F obtained with the generalized FAST algorithm is given in the following theorem.

Theorem 4. *It holds that*

$$\mathbb{P}^{N_1+N_2}\{V(z^*_F)>\epsilon\}\leqslant\beta.\tag{14}$$

Moreover, the value N_2 given in step 1 of the generalized FAST algorithm cannot be improved in the sense that there are problems for which no N_2 smaller than that given in step 1 of the generalized FAST algorithm makes (14) true.

A proof is given in §6.

4.3. Discussion

The essential difference between the FAST algorithm of §3 and the generalized FAST algorithm of this section is in the detuning step: the idea of lifting $l^*_{N_1}$ in the FAST algorithm is replaced in the generalized FAST algorithm by the idea of moving $z^*_{N_1}$ toward $\bar z$. This operation can be performed at low computational effort since (13) is an optimization problem with a scalar decision variable α, so that (13) can be solved, e.g., by means of bisection. Moreover, all observations in the discussion §3.3 can be carried over *mutatis mutandis* to the context of the present section.

5. An Example

In this section, the classical scenario approach is compared with FAST on an instance of the well-known *weighted distribution problem*, see Ferguson et al. (1956), Dantzig (1998).

5.1. Problem Formulation

A company sells n products. The demand for the products over a given period is quantified through the "demand vector" $D = [d_1 \; d_2 \; \cdots \; d_n]$, where d_k is the demand for product k. The company owns m different machines. Each machine can be used to produce any of the n products, although the efficiency varies from product to product and from machine to machine. This is specified by the $m \times n$ "capacity matrix" P, whose entry p_{jk} is the quantity of product k that is produced in a time unit when the jth machine is allocated to that product. Moreover, each machine can only be used for a limited amount of time over a period as specified by the "availability vector" $A = [a_1 \; a_2 \; \cdots \; a_m]^T$. The production costs are given by matrix C, which is again a $m \times n$ matrix and whose entry c_{jk} gives the cost incurred when the jth machine is allocated to product k for a time unit. The entry c_{jk} takes into account the operating cost, the cost of raw materials that are processed in a time unit, etc.

Selling a unitary quantity of product k gives a revenue equal to u_k, and $U = [u_1 \; u_2 \; \cdots \; u_n]$ is the "revenue vector." The total revenue achieved from the production of a quantity q_k of product k is given by $u_k \cdot \min(q_k, d_k)$, where min is because the sold product is no more than the demand for that product. Moreover, overproduction generates an additional cost because of inventory holding. The inventory holding cost for a unitary quantity of product k is $\tilde{c}_k$ and $\tilde{C} = [\tilde{c}_1 \; \tilde{c}_2 \; \cdots \; \tilde{c}_n]$.

The company has to allocate the production over the available machines by choosing the $m \times n$ "allocation matrix" X whose entry x_{jk} represents the amount of time that the jth machine is allocated to product k. The objective is the minimization of the net cost, that is, the difference between the total cost and the total revenue in the time period, given the constraint posed by the availability vector A:

$$\min_{X \in \mathcal{X}} \left\{ \sum_{j=1}^{m} \sum_{k=1}^{n} c_{jk} x_{jk} + \sum_{k=1}^{n} \tilde{c}_k \left[\sum_{j=1}^{m} p_{jk} x_{jk} - d_k \right]_+ - \sum_{k=1}^{n} u_k \min\left(\sum_{j=1}^{m} p_{jk} x_{jk}, d_k \right) \right\}, \tag{15}$$

where

$$\mathcal{X} = \left\{ X\colon \sum_{k=1}^{n} x_{jk} \leqslant a_j,\, j = 1, \ldots, m,\, x_{jk} \geqslant 0,\; j = 1, \ldots, m,\, k = 1, \ldots, n \right\},$$

and $[\cdot]_+$ denotes positive part ($[a]_+ = a$ if $a \geqslant 0$, $[a]_+ = 0$ otherwise).

Formulation (15) is a deterministic model of the weighted distribution problem. In the sequel, more realistically, we shall view some quantities as random. Precisely, we shall treat as random the demand vector D, due to partial unpredictability of customers' behavior, and the capacity matrix P, due, e.g., to the need for human intervention. Hence, we shall adopt a chance-constrained formulation:

$$\min_{X \in \mathcal{X}, l} \; l$$

$$\text{subject to: } \mathbb{P}\left\{ \sum_{j=1}^{m} \sum_{k=1}^{n} c_{jk} x_{jk} + \sum_{k=1}^{n} \tilde{c}_k \left[\sum_{j=1}^{m} p_{jk} x_{jk} - d_k \right]_+ - \sum_{k=1}^{n} u_k \min\left(\sum_{j=1}^{m} p_{jk} x_{jk}, d_k \right) \leqslant l \right\} \geqslant 1 - \epsilon,$$

and the scenario approach will be applied.

5.2. FAST vs. Classical Scenario Approach

Take $m = 5$ and $n = 10$, and let

$$C = \begin{bmatrix} 1.8 & 2.2 & 1.5 & 2.2 & 2.6 & 2.1 & 2.2 & 1.7 & 2.8 & 1.9 \\ 1.6 & 1.9 & 1.3 & 1.9 & 2.3 & 1.9 & 2.0 & 1.5 & 2.5 & 1.7 \\ 1.2 & 1.5 & 1.0 & 1.5 & 1.9 & 1.4 & 1.6 & 1.1 & 2.0 & 1.3 \\ 1.3 & 1.6 & 1.1 & 1.6 & 2.0 & 1.5 & 1.7 & 1.2 & 2.2 & 1.4 \\ 1.2 & 1.5 & 1.0 & 1.6 & 1.9 & 1.5 & 1.6 & 1.1 & 2.1 & 1.3 \end{bmatrix}$$

$$A = \begin{bmatrix} 10 \\ 13 \\ 22 \\ 19 \\ 21 \end{bmatrix}$$

$$\tilde{C} = [1.3 \quad 1.3 \quad 1.3 \quad 1.3 \quad 1.3 \quad 1.3 \quad 1.3 \quad 1.3 \quad 1.3 \quad 1.3]$$

$$U = [1.5 \quad 1.8 \quad 1.2 \quad 1.9 \quad 2.2 \quad 1.8 \quad 1.9 \quad 1.4 \quad 2.4 \quad 1.6].$$

Vector D takes value according to a Dirichlet distribution $\mathrm{Dir}(25, 38, 18, 39, 60, 35, 41, 22, 74, 30)$ multiplied by 382. Since the sum of the components of a Dirichlet distributed vector adds up to 1, the total demand $\sum_{k=1}^{n} d_k$ is equal to 382, and is not subject to stochastic fluctuations. This models a market where products are varieties of the same good, and the preference for a product is to the detriment of the others (for instance, the amount of paint bought is constant, and customers can choose among various colors). As for the capacity matrix P, the p_{jk}'s are assumed to be independent of each other and uniformly distributed around nominal values $\bar{p}_{jk}$'s, as given by the matrix

$$\bar{P} = \begin{bmatrix} 5.0 & 7.6 & 3.6 & 7.8 & 12.0 & 7.0 & 8.2 & 4.4 & 14.8 & 6.0 \\ 3.8 & 5.8 & 2.8 & 6.0 & 9.2 & 5.4 & 6.3 & 3.4 & 11.4 & 4.6 \\ 2.3 & 3.5 & 1.6 & 3.5 & 5.5 & 3.2 & 3.7 & 2.0 & 6.7 & 2.7 \\ 2.6 & 4.0 & 1.9 & 4.1 & 6.3 & 3.7 & 4.3 & 2.3 & 7.8 & 3.2 \\ 2.4 & 3.6 & 1.7 & 3.7 & 5.7 & 3.3 & 3.9 & 2.1 & 7.0 & 2.9 \end{bmatrix},$$

with a variation of ±5% each.

The scenario program is

$$\min_{X \in \mathcal{X}, l} \; l$$

$$\text{subject to: } \sum_{j=1}^{m} \sum_{k=1}^{n} c_{jk} x_{jk} + \sum_{k=1}^{n} \tilde{c}_k \left[\sum_{j=1}^{m} p_{jk}^{(i)} x_{jk} - d_k^{(i)} \right]_+ - \sum_{k=1}^{n} u_k \min\left(\sum_{j=1}^{m} p_{jk}^{(i)} x_{jk}, d_k^{(i)} \right) \leqslant l, \quad i = 1, \ldots, N, \tag{16}$$

where $\delta^{(i)} = (d_1^{(i)}, \ldots, d_n^{(i)}, p_{11}^{(i)}, \ldots, p_{1n}^{(i)}, \ldots, p_{m1}^{(i)}, \ldots, p_{mn}^{(i)})$ is a random extraction of the elements of D and P.

We are interested in a solution with a violation probability no more than $\epsilon = 1\%$, with confidence $1-\beta = 1-10^{-9}$. In the classical scenario approach, letting $B_\epsilon^{N,d} \leqslant 10^{-9}$, see (4), yields $N = 10{,}580$. The scenario program (16) was solved on a Windows 7 system with an Intel Core i5 (2.53 GHz) and 4 GB of RAM. Using YALMIP, Lofberg (2004), with the IBM ILOG CPLEX optimizer, the obtained solution was

$$X_N^* = \left[\begin{matrix} 0 & 0 & 0 & 0.9579 & 4.6336 & 0 \\ 0 & 5.6199 & 0 & 4.4399 & 0 & 0.9391 \\ 0 & 0 & 0 & 0 & 0 & 3.0872 \\ 7.8225 & 0 & 0 & 0 & 0 & 4.7436 \\ 0 & 0 & 10.2830 & 0 & 0 & 0 \end{matrix}\right.$$

$$\left.\begin{matrix} 0 & 0 & 4.4086 & 0 \\ 2.0012 & 0 & 0 & 0 \\ 0 & 10.6160 & 0 & 0 \\ 6.4339 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10.6304 \end{matrix}\right]$$

and $l_N^* = -458.7238$, and the computation time was 7,760 seconds.

Next, we used FAST with $N_1 = 20 \cdot d = 1{,}000$, and, based on (12), $N_2 = 2{,}062$. Running (16) with $N = N_1 = 1{,}000$ we obtained an allocation matrix

$$X_{N_1}^* = \left[\begin{matrix} 0 & 4.9417 & 0 & 0 & 0 & 2.5074 \\ 0 & 0 & 0 & 5.6812 & 0 & 0 \\ 8.8018 & 0 & 5.5402 & 0 & 0 & 0 \\ 0.9443 & 0 & 3.3857 & 0 & 8.7939 & 3.5155 \\ 0 & 0 & 0.3239 & 0 & 0 & 0 \end{matrix}\right.$$

$$\left.\begin{matrix} 0 & 0 & 2.5509 & 0 \\ 4.6527 & 0 & 2.6661 & 0 \\ 0 & 0 & 0 & 0 \\ 2.3606 & 0 & 0 & 0 \\ 0 & 9.8523 & 0 & 10.8238 \end{matrix}\right]$$

with the cost value $l_{N_1}^* = -477.96$. Then, we solved the detuning step with $N_2 = 2{,}062$ additional scenarios, and the final result was $X_F^* = X_{N_1}^*$, and $l_F^* = -453.77$. The overall computation time was 95 seconds. Hence, a drastic reduction in computation time is obtained at the expense of a small increase of cost. Further experiments showed that the difference in time execution becomes rapidly larger as ϵ decreases, with little variation in the cost.

6. Proofs

Theorems 1 and 2 are proved in §6.1, and the proof of Theorem 4 is given in §6.2.

6.1. Proof of Theorems 1 and 2

Define, for brevity, $\boldsymbol{\delta}_m^n := (\delta^{(m)}, \delta^{(m+1)}, \ldots, \delta^{(n)})$, so that $\boldsymbol{\delta}_m^n \in \Delta^{n-m+1}$.

We want to compute the probability of set

$$H = \{\boldsymbol{\delta}_1^{N_1+N_2}\colon\ V(x_F^*, l_F^*) > \epsilon\}.$$

Given $x_{N_1}^*$, consider the set

$$L = \{l\colon\ V(x_{N_1}^*, l) > \epsilon\}.$$

Set L is a random set, depending on $\boldsymbol{\delta}_1^{N_1}$ through $x_{N_1}^*$. Once $x_{N_1}^*$ is fixed, $1 - V(x_{N_1}^*, l)$, as a function of l, is the cumulative distribution function of the random variable $f(x_{N_1}^*, \delta)$. Hence, $V(x_{N_1}^*, l)$ is right continuous and nonincreasing in $\mathbb{R}$, entailing that L can be written as $L =]-\infty, \bar{l}[$ for a suitable $\bar{l}$. The following property provides a useful characterization of set H.

Property 1. *$\boldsymbol{\delta}_1^{N_1+N_2} \in H$ if and only if $V(x_{N_1}^*, l_{N_1}^*) > \epsilon$ and $f(x_{N_1}^*, \delta^{(i)}) \in L$, $\forall i \in \{N_1+1, \ldots N_1+N_2\}$.*

Proof. At the detuning step 4, the FAST algorithm computes $l_F^* = \max_{i=1,\ldots,N_1+N_2} f(x_{N_1}^*, \delta^{(i)})$, i.e., $l_F^* = \max\{l_{N_1}^*, \max_{i=N_1+1,\ldots,N_1+N_2} f(x_{N_1}^*, \delta^{(i)})\}$. If $V(x_{N_1}^*, l_{N_1}^*) > \epsilon$, we have $l_{N_1}^* < \bar{l}$. If $f(x_{N_1}^*, \delta^{(i)}) \in L$, $\forall i \in \{N_1+1, \ldots, N_1+N_2\}$, we have $\max_{i=N_1+1,\ldots,N_1+N_2} f(x_{N_1}^*, \delta^{(i)}) < \bar{l}$. Thus, we have $l_F^* < \bar{l}$, i.e., $l_F^* \in L$, when both conditions hold true simultaneously, yielding $\boldsymbol{\delta}_1^{N_1+N_2} \in H$. Vice versa, if $V(x_{N_1}^*, l_{N_1}^*) \leqslant \epsilon$ we have $l_{N_1}^* > \bar{l}$, so that $l_F^* \geqslant l_{N_1}^* > \bar{l}$, i.e., $l_F^* \notin L$ and $\boldsymbol{\delta}_1^{N_1+N_2}$ is not in H; on the other hand, if $f(x_{N_1}^*, \delta^{(\bar{i})}) \notin L$ for some $\bar{i} \in \{N_1+1, \ldots, N_1+N_2\}$ we have $f(x_{N_1}^*, \delta^{(\bar{i})}) > \bar{l}$, so that $l_F^* \geqslant f(x_{N_1}^*, \delta^{(\bar{i})}) > \bar{l}$, i.e., $l_F^* \notin L$ and $\boldsymbol{\delta}_1^{N_1+N_2}$ is not in H. □

Based on Property 1 we proceed now to evaluate the probability of H:

$$\begin{aligned}
&\mathbb{P}^{N_1+N_2}\{H\} \\
&\quad= [\mathbb{1}\{A\} = \text{indicator function of set } A] \\
&\quad= \int_{\Delta^{N_1+N_2}} \mathbb{1}\{V(x_{N_1}^*, l_{N_1}^*) > \epsilon \text{ and } f(x_{N_1}^*, \delta^{(i)}) \in L, \\
&\qquad\qquad \forall i \in \{N_1+1, \ldots, N_1+N_2\}\}\, \mathbb{P}^{N_1+N_2}\{\mathrm{d}\boldsymbol{\delta}_1^{N_1+N_2}\} \\
&\quad= \int_{\Delta^{N_1+N_2}} \mathbb{1}\{V(x_{N_1}^*, l_{N_1}^*) > \epsilon\} \cdot \mathbb{1}\{f(x_{N_1}^*, \delta^{(i)}) \in L, \\
&\qquad\qquad \forall i \in \{N_1+1, \ldots, N_1+N_2\}\}\, \mathbb{P}^{N_1}\{\mathrm{d}\boldsymbol{\delta}_1^{N_1}\}\, \mathbb{P}^{N_2}\{\mathrm{d}\boldsymbol{\delta}_{N_1+1}^{N_1+N_2}\} \\
&\quad= \int_{\Delta^{N_1}} \mathbb{1}\{V(x_{N_1}^*, l_{N_1}^*) > \epsilon\} \qquad [\text{using Fubini's theorem}] \\
&\qquad \cdot \Bigg[\int_{\Delta^{N_2}} \mathbb{1}\{f(x_{N_1}^*, \delta^{(i)}) \in L, \forall i \in \{N_1+1, \ldots, N_1+N_2\}\} \\
&\qquad\qquad \mathbb{P}^{N_2}\{\mathrm{d}\boldsymbol{\delta}_{N_1+1}^{N_1+N_2}\}\Bigg]\, \mathbb{P}^{N_1}\{\mathrm{d}\boldsymbol{\delta}_1^{N_1}\}.
\end{aligned} \tag{17}$$

As we show below in this proof, the inner integral in the square brackets is upper bounded by $(1-\epsilon)^{N_2}$ for any $\boldsymbol{\delta}_1^{N_1}$, and it is exactly equal to $(1-\epsilon)^{N_2}$ when problem (1) is in class $\mathscr{C}$, according to Definition 2. Whence, $\mathbb{P}^{N_1+N_2}\{H\}$ is upper bounded as follows:

$$\mathbb{P}^{N_1+N_2}\{H\} \\ \leqslant (1-\epsilon)^{N_2}\int_{\Delta^{N_1}} \mathbb{1}\{V(x^*_{N_1}, l^*_{N_1}) > \epsilon\}\mathbb{P}^{N_1}\{d\boldsymbol{\delta}_1^{N_1}\}. \tag{18}$$

The integral in (18) is $\mathbb{P}^{N_1}\{V(x^*_{N_1}, l^*_{N_1}) > \epsilon\}$, a quantity that, according to the classical theory of the scenario approach reminded in §2, is upper bounded by $B_\epsilon^{N_1,d}$, while it is exactly equal to $B_\epsilon^{N_1,d}$ whenever (1) is in class $\mathscr{C}$. Thus, from (18) we conclude that

$$\mathbb{P}^{N_1+N_2}\{H\} \leqslant (1-\epsilon)^{N_2}\cdot B_\epsilon^{N_1,d},$$

which is the statement of Theorem 1, and, if (1) is in $\mathscr{C}$, we have equality, i.e.,

$$\mathbb{P}^{N_1+N_2}\{H\} = (1-\epsilon)^{N_2}\cdot B_\epsilon^{N_1,d},$$

and Theorem 2 is proved.

To complete the proof we have to evaluate the inner integral in (17).

In what follows, we take a fixed $\boldsymbol{\delta}_1^{N_1}$—so that $x^*_{N_1}$ is fixed—and the result is proved by working conditionally with respect to $\boldsymbol{\delta}_1^{N_1}$.

By the independence of $\delta^{(1)}, \dots, \delta^{(N_1)}, \delta^{(N_1+1)}, \dots, \delta^{(N_1+N_2)}$,

$$\begin{aligned}\int_{\Delta^{N_2}} &\mathbb{1}\{f(x^*_{N_1}, \delta^{(i)}) \in L, \forall i \in \{N_1+1, \dots, N_1+N_2\}\}\mathbb{P}^{N_2}\{d\boldsymbol{\delta}_{N_1+1}^{N_1+N_2}\} \\ &= \left(\int_\Delta \mathbb{1}\{f(x^*_{N_1}, \delta) \in L\}\mathbb{P}\{d\delta\}\right)^{N_2} \\ &= \left(\mathbb{P}\{f(x^*_{N_1}, \delta) \in L\}\right)^{N_2} \\ &= \left(\mathbb{P}\{f(x^*_{N_1}, \delta) < \bar{l}\}\right)^{N_2}.\end{aligned} \tag{19}$$

When problem (1) belongs to class $\mathscr{C}$, $\mathbb{P}\{f(x^*_{N_1}, \delta) = l\} = 0$ so that $V(x^*_{N_1}, l) = \mathbb{P}\{f(x^*_{N_1}, \delta) > l\}$ is a continuous function of l. Since $\bar{l}$ is the extreme point of the set where $V(x^*_{N_1}, l) > \epsilon$, by continuity it follows that $V(x^*_{N_1}, \bar{l}) = \epsilon$. Hence, $\mathbb{P}\{f(x^*_{N_1}, \delta) < \bar{l}\} = \mathbb{P}\{f(x^*_{N_1}, \delta) \leqslant \bar{l}\} = 1 - \mathbb{P}\{f(x^*_{N_1}, \delta) > \bar{l}\} = 1 - V(x^*_N, \bar{l}) = 1-\epsilon$ and the right-hand side of (19) equals $(1-\epsilon)^{N_2}$. If (1) is not in $\mathscr{C}$, we prove that $\mathbb{P}\{f(x^*_{N_1}, \delta) < \bar{l}\} \leqslant 1-\epsilon$. To this end, define the sets $L_n :=]-\infty, \bar{l} - 1/n]$ for $n > 1$. Clearly, $L_n \subseteq L$, and $\mathbb{P}\{f(x^*_{N_1}, \delta) \in L_n\} = 1 - V(x^*_{N_1}, \bar{l} - 1/n) < 1 - \epsilon$. Applying the σ-additivity of $\mathbb{P}$, we conclude that

$$\begin{aligned}\mathbb{P}\{f(x^*_{N_1}, \delta) \in L\} &= \mathbb{P}\left\{f(x^*_{N_1}, \delta) \in \bigcup_{n=1}^{\infty} L_n\right\} \\ &= \lim_{n\to\infty}\left[1 - V\left(x^*_{N_1}, \bar{l} - \frac{1}{n}\right)\right] \leqslant 1-\epsilon\end{aligned}$$

and the right-hand of (19) is upper bounded by $(1-\epsilon)^{N_2}$. □

6.2. Proof of Theorem 4

The proof of Theorem 4 follows the same line of reasoning as that of Theorems 1 and 2.

We want to compute the probability of set

$$H = \{\boldsymbol{\delta}_1^{N_1+N_2}\colon V(z^*_F) > \epsilon\}. \tag{20}$$

Given $z^*_{N_1}$, the solution z^*_F obtained by the generalized FAST algorithm lies on the half-line defined as $\hat{z}[\alpha] := (1-\alpha)z^*_{N_1} + \alpha\bar{z}$, $\alpha \in]-\infty, 1]$: this half-line extends the line segment at step 4 of the generalized FAST algorithm in §4.1 beyond point $z^*_{N_1}$. The set Z of points on this half-line with a violation probability bigger than ϵ is formally defined as

$$Z = \{\hat{z}[\alpha]\colon \alpha \in]-\infty, 1] \text{ and } V(\hat{z}[\alpha]) > \epsilon\}.$$

Since sets $\mathcal{Z}_\delta$ are convex and closed, $V(\hat{z}[\alpha])$ is right continuous and nonincreasing in $\alpha \in]-\infty, 1]$. Hence, Z is an open half-line. In formulas, by defining

$$\bar{\alpha} = \sup_{\alpha \in]-\infty, 1]} \{\alpha\colon V(\hat{z}[\alpha]) > \epsilon\}, \tag{21}$$

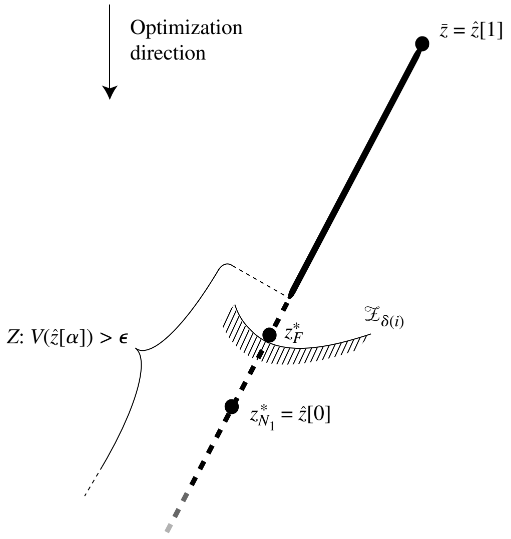
Z can then be rewritten as

$$Z = \{\hat{z}[\alpha]\colon \alpha \in]-\infty, \bar{\alpha}[\}.$$

The following property provides a useful characterization of set H.

Property 2. *$\boldsymbol{\delta}_1^{N_1+N_2} \in H$ if and only if $V(z^*_{N_1}) > \epsilon$ and $Z \cap \mathcal{Z}_{\delta^{(i)}} \neq \varnothing$, $\forall i \in \{N_1+1, \dots, N_1+N_2\}$.*

Figure 3. Optimization domain for problem (13) in step 4 of the generalized FAST algorithm.



Notes. The algorithm returns the point z^*_F closest to $z^*_{N_1}$ and such that $z^*_F \in \mathcal{Z}_{\delta(i)}, \forall i \in \{N_1+1, \dots, N_1+N_2\}$. In this figure, set $\mathcal{Z}_{\delta(i)}$ is the region above the shaded area, and $Z \cap \mathcal{Z}_{\delta(i)} \neq \varnothing$.

This Property 2 can be proved similarly to Property 1 in §6.1, by observing that Z has here the same role as L in §6.1. Refer to Figure 3 for a geometrical visualization of the various objects involved.

Based on Property 2 and mimicking (17), the probability of H can be written as

$$\mathbb{P}^{N_1+N_2}\{H\} = \int_{\Delta^{N_1}} \mathbb{1}\{V(z^*_{N_1})>\epsilon\}\left[\int_{\Delta^{N_2}} \mathbb{1}\{Z\cap\mathcal{Z}_{\delta^{(i)}}\neq\varnothing, \forall i\in\{N_1+1,\ldots,N_1+N_2\}\}\mathbb{P}^{N_2}\{\mathrm{d}\boldsymbol{\delta}_{N_1+1}^{N_1+N_2}\}\right]\mathbb{P}^{N_1}\{\mathrm{d}\boldsymbol{\delta}_1^{N_1}\}.$$

By the independence of $\delta^{(1)},\ldots,\delta^{(N_1)},\delta^{(N_1+1)},\ldots,\delta^{(N_1+N_2)}$, the inner integral in this latter equation can be written as

$$\int_{\Delta^{N_2}} \mathbb{1}\{Z\cap\mathcal{Z}_{\delta^{(i)}}\neq\varnothing, \forall i\in\{N_1+1,\ldots,N_1+N_2\}\}\mathbb{P}^{N_2}\{\mathrm{d}\boldsymbol{\delta}_{N_1+1}^{N_1+N_2}\} = (\mathbb{P}\{Z\cap\mathcal{Z}_{\delta}\neq\varnothing\})^{N_2},$$

which, as we shall show below in this proof, is upper bounded by $(1-\epsilon)^{N_2}$ for every $\boldsymbol{\delta}_1^{N_1}$. Thus, we conclude that

$$\mathbb{P}^{N_1+N_2}\{H\}\leqslant(1-\epsilon)^{N_2}\int_{\Delta^{N_1}}\mathbb{1}\{V(z^*_{N_1})>\epsilon\}\mathbb{P}^{N_1}\{\mathrm{d}\boldsymbol{\delta}_1^{N_1}\} \leqslant(1-\epsilon)^{N_2}B_{\epsilon}^{N_1,d}, \qquad (22)$$

where the last equality follows from the classical theory of the scenario approach; see Campi and Garatti (2008). Theorem 4 follows by substituting in (22) the expression for N_2 given in (12).

The fact that $(\mathbb{P}\{Z\cap\mathcal{Z}_{\delta}\neq\varnothing\})^{N_2}\leqslant(1-\epsilon)^{N_2}$ is now proved by working conditionally on a fixed $\boldsymbol{\delta}_1^{N_1}$, so that $\hat{z}[\alpha]$, $\alpha\in]-\infty,1]$ has to be thought of as a fixed half-line. Define the sets

$$Z_n=\{\hat{z}[\alpha]\colon\ \alpha\in]-\infty,\bar{\alpha}-1/n]\},$$

for $n>1$. Clearly, $\{\delta\in\Delta\colon\ Z_n\cap\mathcal{Z}_{\delta}\neq\varnothing\}=\{\delta\in\Delta\colon\ \hat{z}[\bar{\alpha}-1/n]\in\mathcal{Z}_{\delta}\}$, that is, for $Z_n\cap\mathcal{Z}_{\delta}$ to be nonempty, the extreme point $\hat{z}[\bar{\alpha}-1/n]$ of Z_n must be in $\mathcal{Z}_{\delta}$. Now, by the Definition 3 of violation probability, $\mathbb{P}\{\delta\in\Delta\colon\ \hat{z}[\bar{\alpha}-1/n]\in\mathcal{Z}_{\delta}\}=1-V(\hat{z}[\bar{\alpha}-1/n])$, and by the σ-additivity of $\mathbb{P}$ we have that

$$\mathbb{P}\{Z\cap\mathcal{Z}_{\delta}\neq\varnothing\}=\mathbb{P}\left\{\bigcup_{n=1}^{\infty}\{Z_n\cap\mathcal{Z}_{\delta}\neq\varnothing\}\right\} =\lim_{n\to\infty}[1-V(\hat{z}[\bar{\alpha}-1/n])]\leqslant1-\epsilon,$$

where the last inequality follows from the fact that $V(\hat{z}[\bar{\alpha}-1/n])>\epsilon$, $\forall n$; see (21). Thus, $(\mathbb{P}\{Z\cap\mathcal{Z}_{\delta}\neq\varnothing\})^{N_2}\leqslant(1-\epsilon)^{N_2}$, and the theorem is proved. □

Endnote

[1] This assumption is satisfied in many situations of interest. For example, in robust feedback controller synthesis one can take the $\bar{z}$ corresponding to the zero controller, Campi et al. (2009b). Similarly, a suitable $\bar{z}$ can be easily determined in applications as IPMs (interval predictor models) with bounded noise, Campi et al. (2009a), and robust Chebyshev FIR equalization, Mutapčić et al. (2007). One way to search for a robustly feasible $\bar{z}$ in more general contexts is by robust optimization techniques; see, e.g., Ben-Tal and Nemirovski (1998), El Ghaoui and Lebret (1998), Ben-Tal and Nemirovski (1999), El Ghaoui and Niculescu (2000), Bertsimas and Sim (2004), or by sequential randomized algorithms; see, e.g., Polyak and Tempo (2001), Fujisaki et al. (2003), Oishi (2007), Wada and Fujisaki (2007), Calafiore et al. (2011).

Acknowledgments

This work was supported by the Ministero dell'Istruzione, dell'Università e della Ricerca (MIUR), and by the European Union under project FP7 257005 "MoVeS: Modeling, Verification, and Control of Complex Systems."

References

Alamo T, Tempo R, Camacho EF (2009) Randomized strategies for probabilistic solutions of uncertain feasibility and optimization problems. *IEEE Trans. Automatic Control* 54(11):2545–2559.

Alamo T, Tempo R, Luque A (2010) On the sample complexity of probabilistic analysis and design methods. Willems JC, Hara S, Ohta Y, Fujioka H, eds. *Perspectives in Mathematical System Theory, Control and Signal Processing, Lecture Notes in Control and Information Sciences*, Vol. 398 (Springer, Berlin), 39–50.

Ben-Tal A, Nemirovski A (1998) Robust convex optimization. *Math. Oper. Res.* 23(4):769–805.

Ben-Tal A, Nemirovski A (1999) Robust solutions of uncertain linear programs. *Oper. Res. Lett.* 25(1):1–13.

Bertsimas D, Thiele A (2006) Robust and data-driven optimization: modern decision-making under uncertainty. *Tutorials Oper. Res. INFORMS*, 95–122.

Bertsimas D, Brown DB (2009) Constructing uncertainty sets for robust linear optimization. *Oper. Res.* 57(6):1483–1495.

Bertsimas D, Sim M (2004) The price of robustness. *Oper. Res.* 52(1):35–53.

Boyd S, Vandenberghe L (2004) *Convex Optimization* (Cambridge University Press, Cambridge, UK).

Calafiore GC, Campi MC (2005) Uncertain convex programs: Randomized solutions and confidence levels. *Math. Programming* 102(1):25–46.

Calafiore GC, Campi MC (2006) The scenario approach to robust control design. *IEEE Trans. Automatic Control* 51(5):742–753.

Calafiore GC, Dabbene F, Tempo R (2011) Research on probabilistic methods for control system design. *Automatica* 47(7):1279–1293.

Campi MC, Garatti S (2008) The exact feasibility of randomized solutions of uncertain convex programs. *SIAM J. Optim.* 19(3):1211–1230.

Campi MC, Garatti S (2011) A sampling-and-discarding approach to chance-constrained optimization: Feasibility and optimality. *J. Optim. Theory Appl.* 148(2):257–280.

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

Campi MC, Garatti S, Prandini M (2009b) The scenario approach for systems and control design. *Annual Rev. Control* 33(2):149–157.

Chen W, Sim M, Sun J, Teo CP (2010) From CVar to uncertainty set: Implications in joint chance-constrained optimization. *Oper. Res.* 58(2):470–485.

Dantzig GB (1998) *Linear Programming and Extensions*. Landmarks in Physics and Mathematics (Princeton University Press).

Delage E, Ye Y (2010) Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Oper. Res.* 58(3):595–612.

Dentcheva D (2006) Optimization models with probabilistic constraints. Calafiore G, Dabbene F, eds. *Probabilistic and Randomized Methods for Design Under Uncertainty* (Springer-Verlag, London), 49–97.

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

El Ghaoui L, Niculescu S-I (2000) Robust decision problems in engineering: A linear matrix inequality approach. El Ghaoui L, Niculescu S-I, eds. *Advances in Linear Matrix Inequality Methods in Control* (SIAM), 3–37.

Ferguson AR, Dantzig GB (1956) The allocation of aircraft to routes—An example of linear programming under uncertain demand. *Management Sci.* 3(1):45–73.

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

Garatti S, Campi MC (2013) Modulating robustness in control design: Principles and algorithms. *IEEE Control Systems Magazine* 33(2): 36–51.

Gupta AK, Nadarajah S, eds. (2004) *Handbook of Beta Distribution and Its Applications* (Mercel Dekker, New York), 3–31.

Hong LJ, Yang Y, Zhang L (2011) Sequential convex approximations to joint chance constrained programs: A Monte Carlo approach. *Oper. Res.* 59(3):617–630.

Koltchinskii V, Abdallah CT, Ariola M, Dorato P, Panchenko D (2000) Improved sample complexity estimates for statistical learning control of uncertain systems. *IEEE Trans. Automatic Control* 45(12): 2383–2388.

Lofberg J (2004) YALMIP: A toolbox for modeling and optimization in MATLAB. *Proc. CACSD Conf*. (IEEE, Piscataway, NJ).

Luedtke J, Ahmed S (2008) A sample approximation approach for optimization with probabilistic constraints. *SIAM J. Optim.* 19:674–699.

Mutapčić A, Kim SJ, Boyd SP (2007) Robust Chebyshev FIR equalization. *Proc. 50th IEEE Global Comm. Conf.* (*GLOBECOM '07*) (IEEE, Piscataway, NJ).

Nemirovski A, Shapiro A (2006) Convex approximations of chance constrained programs. *SIAM J. Optim.* 17(4):969–996.

Oishi Y (2007) Polynomial-time algorithms for probabilistic solutions of parameter-dependent linear matrix inequalities. *Automatica* 43(3): 538–545.

Pagnoncelli BK, Ahmed S, Shapiro A (2009) Sample average approximation method for chance constrained programming: theory and applications. *J. Optim. Theory Appl.* 142(2):399–416.

Pagnoncelli BK, Reich D, Campi M (2012) Risk-return trade-off with the scenario approach in practice: A case study in portfolio selection. *J. Optim. Theory Appl*. 155(2):707–722.

Polyak BT, Tempo R (2001) Probabilistic robust design with linear quadratic regulators. *Systems and Control Lett.* 43:343–353.

Prékopa A (1995) *Stochastic Programming* (Kluwer Academic Publishers, Boston).

Prékopa A (2003) Probabilistic programming. Ruszczyński A, Shapiro A, eds. *Stochastic Programming, Handbooks in Operations Research and Management Science*, Vol. 10 (Elsevier, London, UK), 267–351.

Shapiro A, Dentcheva D, Ruszczyński A (2009) *Lectures on Stochastic Programming: Modeling and Theory* (Society for Industrial and Applied Mathematics and Mathematical Programming Society, Philadelphia).

Wada T, Fujisaki Y (2007) Sequential randomized algorithms for robust optimization. *46th IEEE Conf. Decision and Control* (IEEE, Piscataway, NJ), 6190–6195.

Algo Carè is currently a research fellow at the Department of Electrical and Electronic Engineering, University of Melbourne, Australia. His research interests lie in stochastic optimization, system identification, and machine learning.

Simone Garatti is an assistant professor at the Dipartimento di Elettronica, Informazione e Bioingegneria of the Politecnico di Milano. He has been a visiting scholar at the Lund University of Technology, Lund, Sweden, at the University of California San Diego, and at the Massachusetts Institute of Technology and the Northeastern University, Boston. He is a member of the IEEE Technical Committee on Computational Aspects of Control System Design and of the IFAC Technical Committee on Modeling, Identification, and Signal Processing. His research interests include system identification and model quality assessment, identification of interval predictor models, and randomized optimization for problems in systems and control.

Marco C. Campi is a Professor of Automatic Control at the University of Brescia, Italy. He is the chair of the Technical Committee IFAC on Modeling, Identification, and Signal Processing (MISP), and has served as a distinguished lecturer of the Control Systems Society. In 2008, he received the IEEE CSS George S. Axelby outstanding paper award for the article "The Scenario Approach to Robust Control Design." He is a Fellow of IEEE, a member of IFAC, and a member of SIDRA. His research interests include randomized methods, robust convex optimization, system identification, robust control and estimation, and learning theory.