2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

Robust and Chance-Constrained Optimization under Polynomial Uncertainty

F. Dabbene, C. Feng and C. M. Lagoa

***Abstract*— A chance-constrained optimization problem, induced from a robust design problem with polynomial dependence on the uncertainties, is, in general, non-convex and difficult to solve. By introducing a novel concept – the *kinship function* – an easily computable convex relaxation of this problem is proposed. In particular, *optimal polynomial* kinship functions, which can be computed a priori and once for all, are introduced and used to bound the probability of constraint violation. Moreover, it is proven that the solution of the relaxed problem converges to that of the original robust optimization problem as the degree of the polynomial kinship function increases. Finally, by relying on quadrature formulae for computation of integrals of polynomials, it is shown that the computational complexity of the proposed approach is polynomial on the number of uncertainty parameters.**

I. Introduction

In this paper, we provide a new approach aimed at determining approximate solutions of uncertain optimization problems which often arise in systems and control. In particular, deterministic convex relaxations of robust and chance-constrained optimization problems are developed and shown to be solvable in *polynomial time*.

More precisely, consider a convex uncertain optimization problem

$$\begin{aligned} \min_{x\in\mathcal{X}} \quad & c^\top x \\ \text{subject to} \quad & f(x,q)\le 0 \text{ for all } q\in\mathcal{Q} \end{aligned} \tag{1}$$

where $f(x,q)$ is convex in x for fixed q and polynomial in q for fixed x. Given this robust formulation, one can define its probabilistic counterpart. That is, assume that the uncertainty q is random with known probability density function $\mu(q)$ and that a fixed level of violation probability ϵ is given. Then, one can formulate a probabilistic relaxation of the robust problem above, the so-called *chance-constrained* problem, as

$$\begin{aligned} \min_{x\in\mathcal{X}} \quad & c^\top x \\ \text{subject to} \quad & \text{Prob}\{q\in\mathcal{Q} \ : \ f(x,q)>0\}\le\epsilon. \end{aligned} \tag{2}$$

Motivations for considering a probabilistic approach are numerous, from philosophical to computational ones; e.g., see [8], [21].

However, both these problems are known to be in general very hard. The first one is usually referred to as a "semi-infinite" optimization problem, since it has an infinite number of constraints, one for every q in Q. It is known that exact solutions can be computed by tractable algorithms only for very specific problem instances. For example, if one has affine or multi-affine dependence on the uncertain parameters and the support set for the uncertainty is an hypercube, then one can solve the problem by using only the extreme points; e.g., see [4], [7]. In general, however, this problem is known to be NP-hard [2]. Therefore, various relaxations have been proposed to find approximate solutions, which can be very conservative. In [2], an approximate solution is found by projecting the "optimizer" in an augmented feasible set of higher dimension. In the case of linear matrix inequality (LMI) constraints, a solution has been proposed for the case when the parameters enter in a so-called linear fractional transformation form; see [13].

Since a robust solution is usually conservative, it is desired to have a tradeoff between the robustness and the risk of failure in many practical problems. Therefore, the so-called *chance-constrained* optimization problem has been proposed. This problem, as formulated above, is, in general, non-convex. Randomized algorithms have been widely used for finding approximate solutions to this problem. It was shown in [9], [19] that one can use stochastic approximation algorithms to address this problem. Also, by randomly sampling the uncertainty, one can get a "large number" of constraints (rather than infinite many) to obtain an approximate solution with given confidence [6]. However, the results obtained by these techniques are only shown to hold with a given degree of confidence; i.e. given the stochastic nature of the algorithms, the solutions always entail a positive (though arbitrary small) probability of failure.

In this paper, we take a different approach. By introducing the concept of *kinship function*, we are able to extend the results in [1], [10], and provide algorithms that not only estimate the probability of performance violation but also address the chance-constrained *design* problem (2) via a convex *deterministic* relaxation. The proposed relaxation enjoys the following interesting properties: i) it is solvable in *polynomial time*; ii) has a provable probability of constraint violation; and iii) its solution can be made arbitrarily close to the solution of the robust optimization problem (1).

One crucial characteristic of kinship functions is that they can be computed *a priori and once for all*. Moreover, if polynomial kinship functions are chosen, particular combinations of quadrature formulae, the so-called *Smolyak formulae*, can be used to ensure *polynomial-time* complexity of the proposed algorithm.

This work was supported by the National Science Foundation under grants CNS–0931511, ECCS–0731224 and ECCS–0501166.

F. Dabbene is with IEIIT-CNR, Politecnico di Torino, Italy (email: fabrizio.dabbene@polito.it).

C. Feng and C. M. Lagoa are with the Penn State University, University Park, PA 16802, USA (email: feng@psu.edu, lagoa@engr.psu.edu).

II. Notation and Problem Formulation

A. Notation

Given a vector x, $\|x\|_p$ denotes its ℓ_p norm. For a symmetric matrix F, $F \succeq 0$ denotes semi-positive definiteness of F. We denote by $\lceil x \rceil$ the smallest integer larger or equal than $x \in \mathbb{R}$, and $\lfloor x \rfloor$ the largest integer smaller or equal than x. We define the space of one dimensional polynomials of degree ν as

$$\mathbb{P}_\nu \doteq \left\{ p(x) : \mathbb{R} \to \mathbb{R} \,|\, p(x) = \sum_{k=0}^{\nu} a_k x^k \right\}.$$

Similarly, $\mathbb{P}_\nu^d$ defines the space of all polynomials in d variables of total degree (i.e. the sum of the degrees with respect to the individual variables[1]) at most ν.

B. Problem Formulation

With this notation introduced, we are now ready to precisely state the problem addressed in the paper: We aim at finding a solution of problem (2) for the case where the following formal assumptions are satisfied:

Assumption 1 (Assumptions on Problem (2)):

(i) the set $\mathcal{X} \subset \mathbb{R}^{n_x}$ is a bounded convex set,
(ii) for fixed $q \in \mathcal{Q}$, the function $f(\cdot, q)$ is convex in x,
(iii) for fixed $x \in \mathcal{X}$, the function $f(x, \cdot)$ is a polynomial of total degree σ in x,
(iv) the function $f(\cdot, \cdot)$ is bounded from below in $\mathcal{X} \times \mathcal{Q}$,
(v) for fixed $q = \bar{q}$, a (sub)gradient $\partial_f(x, \bar{q})$ of $f(x, \bar{q})$ is available.

Since a rescaling of f is alway possible, without loss of generality, we assume that the lower bound on the function f is one; i.e.,

$$f : \mathcal{X} \times \mathcal{Q} \to [-1, \infty). \tag{3}$$

As for the uncertainty, the only assumption made is that the entries of q are independently distributed. More precisely, one makes the following assumption:

Assumption 2: The probability density function of the uncertainty q is of the form

$$\mu(q) = \mu_1(q_1)\mu_2(q_2)\cdots\mu_d(q_d) \tag{4}$$

and the uncertainty support set $\mathcal{Q}$ can be written as the cartesian product of (possibly unbounded) intervals

$$\mathcal{Q} = \mathcal{Q}_1 \times \mathcal{Q}_2 \times \cdots \times \mathcal{Q}_d. \tag{5}$$

III. Convex relaxation based on kinship function

In this section, we define the central concept of this paper and show how to use it to construct a convex relaxation of problem (2). To this end, first consider the following definition of a *kinship function*.

Definition 1: A function $\kappa : [-1, \infty) \to \mathbb{R}$ is said to be a *kinship function* if

(a) $\kappa(0) = 1$,
(b) $\kappa(y)$ is a convex, nonnegative and nondecreasing function for $y \in [-1, \infty)$.

Then we have the following key properties regarding the kinship function.

Theorem 1: Let $\kappa(\cdot)$ be a kinship function, define

$$V_\kappa(x) \doteq \int_{\mathcal{Q}} \kappa[f(x,q)]\mu(q)\mathrm{d}q, \tag{6}$$

then

$$\text{Prob}\{q \in \mathcal{Q} \ : \ f(x,q) > 0\} \le V_\kappa(x).$$

Proof. By definition, the kinship function $\kappa[f(x,q)]$ is nonnegative in $[-1, \infty)$ and greater than 1 if $f(x,q) \ge 0$. Thus, for any probability measure μ on $\mathcal{Q}$,

$$\begin{aligned} V_\kappa(x) &\ge \int_{\{q \in \mathcal{Q} : f(x,q) > 0\}} \mu(q)\mathrm{d}q \\ &= \text{Prob}\{q \in \mathcal{Q} \ : \ f(x,q) > 0\}. \end{aligned}$$

□

In words, Theorem 1 indicates that, given a kinship function κ satisfying Definition 1, one can define a *relaxation* of the chance constrained problem (2) as:

$$\begin{aligned} \min_{x \in \mathcal{X}} \quad & c^\top x \\ \text{subject to} \quad & V_\kappa(x) \le \epsilon. \end{aligned} \tag{7}$$

Theorem 2 (Convexity of the relaxation): The relaxed optimization problem (7) is convex.

Proof. As assumed, $f(x,q)$ is convex in x for any fixed q. Also, the kinship function $\kappa(\cdot)$ is non-decreasing and convex. The composite function $\kappa[f(x,q)]$ is thus convex. Hence, the integral $V_\kappa(x)$ is convex since μ is a probability (non-negative) measure on $\mathcal{Q}$. Note that non-negative weighted integration and function composition are standard operations that preserve convexity; see e.g. [5]. □

The class of all kinship functions is rather large. Hence, in the remainder of this paper, we concentrate on the subclass of polynomial kinship functions. It is shown that, as the degree of the kinship functions increases, the solution of the relaxed problem (7) converges to the solution of problem (1). Moreover, it is shown in Section V how to efficiently numerically solve the minimization problem (7).

A. Optimal Polynomial Kinship Functions

We now consider the case of designing "optimal" polynomial kinship functions. We stress again that, for our purposes, these functions are computed *a priori* and *once for all*.

Definition 2 (Optimal polynomial kinship function): The optimal polynomial kinship function of degree ϱ is defined as the solution of the following optimization problem

$$\begin{aligned} \min_{a_0,\ldots,a_\varrho} \quad & \textstyle\int_{-1}^{0} p(y)dy \\ \text{subject to} \quad & \text{(a)-(b) in Definition 1,} \\ & p(y) \in \mathbb{P}\varrho. \end{aligned} \tag{8}$$

The following lemma shows that, in the case of polynomial kinship functions, the constraints above can be substantially simplified. Due to space constraints, the proof is omitted.

[1]The total degree of a monomial $m(x) = x_1^{\varrho_1} x_2^{\varrho_2} \cdots x_d^{\varrho_d}$ is the sum of the exponents $deg(m(x)) = \varrho_1 + \varrho_2 + \cdots + \varrho_d$. The total degree of multivariate polynomial is defined as the maximum total degree of its monomials.

The reader is referred to [11] for an extended version of this paper.

Lemma 1: The optimal polynomial kinship function of degree ϱ, denoted as $\kappa_\varrho(y)$, is the solution of the following optimization problem

$$\min_{a_0,\dots,a_\varrho} \quad \int_{-1}^{0} p(y)dy \tag{9}$$
$$\text{subject to} \quad p(y) \in \mathbb{P}\varrho.$$
$$p(0) = 1, \tag{10}$$
$$p(-1) = 0, \tag{11}$$
$$p'(-1) = 0, \tag{12}$$
$$p''(y) \geq 0 \text{ for } y \in [-1,\, \infty). \tag{13}$$

□

It is obvious that, with respect to the coefficients $a_0, \dots, a_\varrho$, the objective function and the first three constraints (10), (11) and (12) are linear. Moreover, constraint (13) can be recasted as an LMI. To see this, following the reasoning in Section 3.2 in [15], define two series of Hankel matrices $H_{1,k} \in \mathcal{R}^{(n_1+1)\times(n_1+1)}$ and $H_{2,k} \in \mathcal{R}^{(n_2+1)\times(n_2+1)}$, with $n_1 = \lfloor \frac{\varrho-2}{2} \rfloor$ and $n_2 = \lfloor \frac{\varrho-3}{2} \rfloor$, as

$$H_{i,k}^{(rl)} = \begin{cases} 1, & \text{if } r+l = k+1, \\ 0, & \text{otherwise.} \end{cases}, i = 1, 2.$$

Then, the optimization problem in Lemma 1 can be reformulated as follows.

$$\begin{aligned} \min_{a_0,\dots,a_\varrho,Y_1,Y_2} \quad & \textstyle\sum_{i=0}^{\varrho} \frac{(-1)^i}{i+1} a_i \\ \text{subject to} \quad & a_0 = 1, \\ & \textstyle\sum_{i=0}^{\varrho} (-1)^i a_i = 0, \\ & \textstyle\sum_{i=1}^{\varrho} i(-1)^{i-1} a_i = 0, \\ & Y_1 H_{1,k} + Y_2 H_{2,k} - \\ & \quad - \textstyle\sum_{i=k+2}^{\varrho} \frac{i!(-1)^{i-k-2} a_i}{k!(i-k-2)!} = 0, \\ & \qquad \text{for } k = 0, \dots, \varrho - 2, \\ & Y_k \succeq 0, \text{ for } k = 1, 2. \end{aligned}$$

Remark 1 (Interpretation of optimality): The above optimization problem is indeed the one of seeking a polynomial that minimizes its $\mathcal{L}_1$ norm on $[-1, 0]$ among all possible polynomial kinship functions with degree ϱ. The function $\kappa(\cdot)$ here is acting as a weighting function for $f(x, q)$. In this way, one can expect that the subset $\{q \in \mathcal{Q} : f(x, q) < 0\}$ contributes less possible on calculating the integral $V_\kappa(x)$. In more general cases, an additional weighting function $w(\cdot)$ may be introduced, to minimize the weighted norm of $p(\cdot)$ on $[-1, 0]$: $\int_{-1}^{0} p(y)w(y)\mathrm{d}y$.

B. Connection to Robust Optimization Problem

With these optimal polynomial kinship functions at hand, the intuition is that, for any given $x \in \mathcal{X}$, the set $\{q \in \mathcal{Q} : f(x, q) < 0\}$ contributes less and less to the integral $V_\kappa(x)$ as the degree ϱ increases. Moreover, the contribution of the set $\{q \in \mathcal{Q} : f(x, q) \geq 0\}$ increases with the degree of the optimal kinship function. Theorem 3 below shows that the reasoning above is indeed true. In this theorem, asymptotic properties of the relaxed problem (7) are provided for the case where optimal polynomial kinship functions are used. Due to space constraints, the proof is omitted. The reader is referred to [11] for an extended version of this paper.

Theorem 3: Assume that the uncertainty set $\mathcal{Q}$ is compact, and the probability measure μ is continuous and strictly positive on $\mathcal{Q}$. Also assume that the original robust problem (1) admits an *unique* solution x^*. Let x_ϱ^* be the solution of the optimization problem (7), with $\kappa_\varrho(\cdot)$ being the optimal polynomial kinship function of degree ϱ, i.e.

$$x_\varrho^* \doteq \arg\min_{x\in\mathcal{X}} c^\top x \tag{14}$$
$$\text{s.t.} \quad V_\kappa(x, \varrho) \leq \epsilon,$$

with

$$V_\kappa(x, \varrho) \doteq \int_{\mathcal{Q}} \kappa_\varrho[f(x, q)]\mu(q)\mathrm{d}q. \tag{15}$$

Then

$$\lim_{\varrho\to\infty} x_\varrho^* = x^*.$$

□

The theorem above formally guarantees that the solution of the relaxed problem (7) can be made arbitrarily close to the unique solution of the robust problem (1), as long as the degree ϱ is chosen large enough. In the next sections, we show how to compute $V_\kappa(x, \varrho)$ efficiently.

IV. A FINITE REPRESENTATION OF V_κ

The idea at the basis of the technique we propose for numerically solving the optimization problem (7) is to compute the integral in (15) using a *quadrature formula* (QF), as briefly recalled next.

A. Quadrature formulae

It is a well-known fact that, for a given function $g : \mathcal{Q}_j \to \mathbb{R}$, the one-dimensional definite weighted integral

$$I[g] = \int_{\mathcal{Q}_j} g(q_j)\mu_j(q)\mathrm{d}q_j$$

can be computed using an N-point quadrature formula of the form

$$\mathsf{Q}_{N,j}[g] \doteq \sum_{k=1}^{N} \omega_{k,j}\, g(\theta_{k,j}) \tag{16}$$

where $\theta_{k,j}$, $\omega_{k,j}$, $k = 1, \dots, N$, are, respectively, the nodes and weights of the quadrature formula[2]. Equation (16) corresponds to evaluating the function $g(\cdot)$ on the non-uniform *grid* $\Theta_j \doteq \{\theta_{1,j}, \dots, \theta_{N,j}\}$.

Associated to a given quadrature formula, we define the error, or residual as $R_N[g] \doteq I[g] - \mathsf{Q}_N[g]$. Then, the *degree of exactness* (DoE) of a QF, denoted by $\deg(\mathsf{Q}_N)$, is defined

[2]The notation with double suffix N, j is introduced to indicate that the quadrature rule is constructed over N nodes, and it is relative to the interval $\mathcal{Q}_j$ and weighting function μ_j (different quadrature rule are usually constructed depending on the different intervals and weighting functions). In the sequel, for the ease of readability and when clear from the context, the unnecessary suffixes will be dropped.

as the maximum integer s such that the QF is exact for all polynomials of degree less or equal than s, that is

$$R_N[p] = 0 \quad \forall p \in \mathbb{P}_s, \quad s \leq \deg(\mathsf{Q}_N)$$

and there exists at least a polynomial p in $\mathbb{P}_{s+1}$ such that $R_N[p] \neq 0$. In particular, when μ is the uniform distribution, if the nodes are chosen as the zeros of the N-th order Legendre orthogonal polynomial $P_N(x)$, and the weights are computed by integrating the associated Lagrange polynomials, then the QF (16) has the maximum achievable degree of exactness

$$\deg(\mathsf{Q}_N) = 2N - 1,$$

and is called a *Gauss formula*.

It is evident that the approach previously described can be extended in a straightforward way to the general multidimensional problem of computing the multivariate integral (15), for $d \geq 1$. Let us consider the computation of the generic multidimensional integral

$$I_d[g] = \int_{\mathcal{Q}} g(q)\mu(q)\mathrm{d}q \tag{17}$$

for $g : \mathcal{Q} = \mathcal{Q}_1 \times \cdots \times \mathcal{Q}_d \to \mathbb{R}$.

In particular, let the function $g(q)$ have total degree ν in the variable q. Then, it may be easily verified that the integral (17) can be computed by means of the tensor product formula

$$\begin{aligned} \mathsf{T}^d[g] &\doteq \left(\mathsf{Q}_1 \otimes \cdots \otimes \mathsf{Q}_d\right)[g] \\ &\doteq \sum_{k_1=1}^{N_1} \cdots \sum_{k_d=1}^{N_d} (\omega_{k_1} \cdots \omega_{k_d}) g(\theta_{k_1} \cdots \theta_{k_d}) \end{aligned} \tag{18}$$

by choosing the number of nodes N_j such that the one-dimensional QFs have DoE ν, i.e. $\deg(\mathsf{Q}_j) \geq \nu$.

B. Computation of V_κ

From these above premises, it follows we can apply the tensor formula (18) to compute the integral in (15). Indeed, we notice that, from our assumptions, it follows that $\kappa_\varrho[f(x,q)]$ is a polynomial in q (for fixed x) of total degree less or equal that $\sigma\varrho$. Then, we can choose N such that $\deg(\mathsf{Q}_N) \geq \sigma\varrho$ and obtain

$$V_\kappa(x,\varrho) = \sum_{k=1}^{N} \omega_k \kappa_\varrho[f(x,\theta_k)].$$

Remark 2 (Complexity of the tensor quadrature formula): Unfortunately the multidimensional quadrature formula (18) may become untractable due to the exponential growth, with respect to the dimension d, of the number of functional evaluations. To see this, assume that $\mathcal{Q}_j$ is constructed on N_j nodes and Then, applying formula (18) corresponds to evaluating the function on the multidimensional grid

$$\mathsf{H}_d \doteq \left(\Theta_1 \times \cdots \times \Theta_d\right) \subset \mathcal{Q}^d,$$

which has cardinality $|\mathsf{H}_d| = \prod_{i=j}^{d} N_j$, which exhibits exponential dependence[3] on d.

[3]To see that, just consider the case when $N_j = N$ for each $j = 1, \ldots, d$.

To avoid this exponential behavior, we introduce next special combinations of low-order QF which exhibit a polynomial dependence on d.

C. Sequences of QF and Smolyak formulae

For every $j = 1, \ldots, d$, we consider a sequence of one-dimensional quadrature formulae of increasing precision

$$\mathsf{Q}_j^{(i)}[g] \doteq \mathsf{Q}_{N_i,j}[g] = \sum_{k=1}^{N_i} \omega_{k,j}^{(i)}\, g(\theta_{k,j}^{(i)}), \quad i = 1, 2, \ldots \tag{19}$$

with nodes $\Theta_j^{(i)} \doteq \left\{\theta_{1,j}^{(i)}, \ldots, \theta_{N_i,j}^{(i)}\right\}$ and weights $\omega_1^{(i)} \cdots \omega_{N_i}^{(i)}$. The number of nodes N_i in (19) is assumed to be a function of the *precision index* i. In order to apply our construction, we consider special sequences of quadrature formulae that satisfy particular requirements (see [10], [18] for a more formal description). The introduced sequences are at the basis of the formula proposed by Smolyak in 1963 [20] for the construction of particular cubature rules with low number of points. Formally, the Smolyak formula with *precision index* ℓ is defined as follows.

$$\begin{aligned} \mathsf{S}_d^{(\ell)}[g] &= \sum_{\ell+1 \leq \|\mathbf{i}\|_1 \leq \ell+d} (-1)^{\ell+d-\|\mathbf{i}\|_1} \binom{d-1}{\|\mathbf{i}\|_1 - \ell - 1} \\ &\quad \left(\mathsf{Q}_1^{(i_1)} \otimes \cdots \otimes \mathsf{Q}_d^{(i_d)}\right)[g], \end{aligned} \tag{20}$$

where $\mathbf{i} \doteq [i_1 \cdots i_d]^\top, \quad \mathbf{i} \in \mathbb{N}_+^d$ is the vector of precision indices for each dimension.

Looking at the formula, it is evident that the Smolyak cubature rule is a linear combination of product formulae involving only relatively-low precision quadrature formulae, chosen in such a way that the interpolation property for $d = 1$ is maintained for $d > 1$. The Smolyak formula gives rise to a highly non uniform grid, which is composed of a subset of the nodes of a full cartesian grid. Thus, we expect the number of nodes of a sparse grid $\mathsf{H}_d^{(\ell)}$ to increase much slower with respect to the dimension d. The characteristics of such formula are reported in the next two propositions, see [16] for formal proofs.

Proposition 1 (Polynomial complexity): For fixed ℓ, the Smolyak formula is polynomial in d, that is $\mathsf{N}_d^{(\ell)} = \mathcal{O}\left(d^\ell\right)$. In particular, we have

$$\mathsf{N}_d^{(\ell)} \approx \frac{2^\ell}{\ell!} d^\ell.$$

Proposition 2 (DoE of Smolyak formulae): The degree of exactness of the Smolyak formula is such that

$$\deg(\mathsf{S}_d^{(\ell)}) \geq 2\ell + 1.$$

The first result shows that the overall complexity of the Smolyak construction is *polynomial in* d, while the second shown that the degree of exactness of $\mathsf{S}_d^{(\ell)}$ is *at least* $2\ell + 1$ (Indeed, better results hold for specific sequences introduced in the literature, see e.g. [16], [18]).

Remark 3 (Independence from problem instance): An important point which should be remarked is that, for given ℓ and d, the $\mathsf{N}_d^{(\ell)}$ nodes of Smolyak cubature formula $\mathsf{S}_d^{(\ell)}[g]$ can be computed *once for all* and stored for successive

computations. Indeed, these quantities do not depend on the integrand, but only on ℓ, d. The procedure for actually computing the nodes and weights is the most time consuming part of the proposed computational scheme. For this reason, a repository of nodes and weights for different values of ℓ, d has been created and is available on request. Specific algorithms for fast construction of the coefficients of Smolyak formulae are provided in [17].

V. An efficient numerical algorithm

Define by Θ_k and w_k, $k = 1, \ldots, \mathsf{N}_d^{(\ell)}$, the nodes and weights of formula (20), then this formula can be written in $\mathsf{N}_d^{(\ell)}$-points cubature

$$\mathsf{S}_d^{(\ell)}[g] = \sum_{k=1}^{\mathsf{N}_d^{(\ell)}} \mathsf{w}_k g(\Theta_k).$$

Hence, as long as these nodes and weights are computed *a priori* for given ℓ, d, formula (20) represents a cubature rule with a number of nodes[4] which increases polynomially with respect to d. Hence, it is now possible to state the main result of this section, which provides an efficient formulation of the optimization problem (7).

Theorem 4: Consider problem (7) and let Θ_k, w_k, $k = 1, \ldots, \mathsf{N}_d^{(\ell)}$ be the nodes and weights of a Smolyak formula with precision index $\ell = \left\lceil \frac{\sigma\varrho - 1}{2} \right\rceil$. Then, the optimization problem (7) is equivalent to the following one

$$\begin{aligned} \min_{x \in \mathcal{X}} \quad & c^\top x \\ \text{subject to} \quad & \sum_{k=1}^{\mathsf{N}_d^{(\ell)}} \mathsf{w}_k \kappa_\varrho[f(x, \Theta_k)] \leq \epsilon. \end{aligned} \tag{21}$$

Moreover, for fixed ν, the number of points $\mathsf{N}_d^{(\ell)}$ is polynomial in d.

Proof. The theorem follows from a direct application of Propositions 1 and 2. □

Problem (21) is now a standard convex optimization problem, and standard tools as (sub)gradient descent or ellipsoidal/cutting plane localization methods can be applied for his solution. For completeness, we report next the (sub)gradient of V_κ, which can be easily verified by direct computation.

Proposition 3 (Subgradient of V_κ): For given $q \in \mathcal{Q}$, let $\partial_f(x, q)$ be a (sub)gradient of the function f with respect to x. Consider a polynomial kinship function of degree ϱ

$$\kappa_\varrho(y) = \sum_{j=0}^{\varrho} a_j y^j.$$

Then

$$\partial_{V_\kappa}(x) \doteq \sum_{k=1}^{\mathsf{N}_d^{(\ell)}} \mathsf{w}_k \partial_f(x, \Theta_k) \left(\sum_{j=0}^{\varrho} j\, a_j [f(x, \Theta_k)]^{j-1} \right) \tag{22}$$

is a (sub)gradient of $V_\kappa(x, \varrho)$.

[4]In computing the weights w_k, some of them may turn out to be zero, therefore leading to a formula with less than $\mathsf{N}_d^{(\ell)}$ nodes.

VI. Application example: the portfolio problem

In the example, we consider a one-period portfolio problem. Assume that there are n investment options. Denote x_i the amount of budget put in the i-th option and let r_i be the corresponding return. The general goal is to minimize the risk while maximizing the total return $r^T x$. Here we choose the factor model and adopt the Value at Risk (VaR) approach. Then, for a given *risk level* $\eta \in (0, 1)$, the portfolio selection problem can be stated as follows (see e.g. [3], [14]):

$$\begin{aligned} \min_{\gamma, x \in \mathcal{X}} \quad & \gamma \\ \text{subject to} \quad & h(x, q) \leq \gamma \text{ for all } q, \end{aligned}$$

where $h(x, q) = k(\eta)\sqrt{x^T(A(q)\Sigma A(q)^T + D)x} - x^T A(q)\hat{\phi}$. The variable γ is usually referred to as *loss level*, while $k(\eta)$ is a given function of η. The parameters $\hat{\phi}$, Σ and D are computed from real market data. In this example, the sensitivity matrix $A(q)$ is assumed to be an interval matrix of the form

$$A(q) = A_0 + \sum_{i,j} q_{ij} A_{ij},$$

where $\|q\|_\infty \leq q_{max}$ and A_{ij} has all elements equal to zero except the i,j-th one that $[A_{ij}]_{i,j} = [A_0]_{i,j}$. The allowable investment policy set $\mathcal{X}$ is assumed to be

$$\mathcal{X} := \left\{ x : \sum_{i=1}^{n} x_i = 1, x_i \geq 0, i = 1, 2, ..., n \right\}.$$

A. The Convex Feasibility Problem

Note that $h(x, q)$ is not polynomial in q. However, if we fix the loss level γ and aim at finding the optimal investment policy that is feasible, the constraint is equivalent to

$$f(x, q) \leq 0 \quad \text{for all } q \in \mathcal{Q},$$

where

$$\begin{aligned} f(x, q) = x^T \left(A(q) \left(k^2(\eta)\Sigma - \hat{\phi}\hat{\phi}^T \right) A(q)^T + k^2(\eta) D \right) x \\ -2\gamma w^T A(q)\hat{\phi} - \gamma^2. \end{aligned}$$

Hence one obtains an equivalent constraint which is polynomial in q.

Moreover, by assuming that

$$k^2(\eta)\Sigma - \hat{\phi}\hat{\phi}^T \succ 0, \tag{23}$$

the function $f(x, q)$ is also convex in x. Hence, the problem is ready to be handled in our framework. It worth to point out that assumption (23) is indeed satisfied in real market data for low risk level; e.g., see Table 2 in [12] for $\eta = 5\%$.

Finally, we notice that a lower bound of the function $f(x, q)$ can be easily calculated,

$$\begin{aligned} f(x, q) &\leq -2\gamma x^T A(q)\hat{\phi} - \gamma^2 \\ &\leq -2\gamma \max_q \{\|A(q)\hat{\phi}\|_\infty\} - \gamma^2. \end{aligned}$$

Hence, we can always pick a constant $\alpha > 0$ such that $\alpha f(x, q) \geq -1$ for all x and q.

B. Numerical results

In the numerical example, we considered a portfolio problem with number of assets $n = 3$ and number of factors $m = 2$. The risk level was chosen to be $\eta = 5\%$. The nominal sensitivity matrix A_0, the covariance matrices Σ and D, and the expect value of the factors $\hat{\phi}$ were randomly generated,

$$A_0 = \begin{pmatrix} 0.4666 & -0.6952 \\ 0.2447 & -0.5934 \\ 0.9796 & 0.6386 \end{pmatrix},$$

$$D = \text{diag}\{\begin{pmatrix} 0.1902 & 0.5995 & 0.2923 \end{pmatrix}^T\},$$

$$\Sigma = \begin{pmatrix} 0.2009 & 0.1791 \\ 0.1791 & 0.4489 \end{pmatrix}, \quad \hat{\phi} = \begin{pmatrix} 0.0584 \\ 0.5385 \end{pmatrix}.$$

To illustrate, we fix the uncertainty magnitude $q_{max} = 0.05$, to find an investment x (if any) such that the VaR is less than a given loss level γ. This is relaxed, in the proposed framework, to find a policy x that $\text{Prob}\{f(x,q) > 0\} \leq \epsilon$. The optimal kinship function is chosen to be of order $\varrho = 3$, which leads to a total degree $\nu = 6$. Then the integral $\int_{\mathcal{Q}} f(x,q)dq$ can be evaluated exactly by using Smolyak rules with $l = 3$ and $d = nm = 6$. The simulation was run for γ increasing from 2 to 4. Figure 1 shows for different γ, the estimates of the upper bound of the probability, or the volume percentage of the "bad" set in $\mathcal{Q}$.

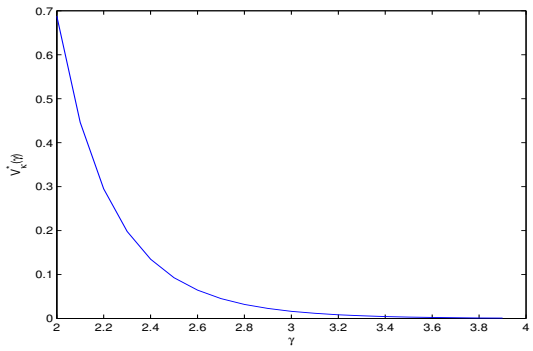

Fig. 1. $V^*_{\kappa_3}(\gamma)$ v.s. γ for $q_{max} = 0.05$.

At the point $\gamma = 2.6$, the best policy

$$x = \begin{pmatrix} 0.4518 & 0.1852 & 0.3630 \end{pmatrix}^T$$

leads to the estimate $V^*_{\varrho_3} = 0.06$. By sampling 10,000,000 points over the set $\mathcal{Q}$, we found actually that none of them violates the inequality $f(x,q) \leq 0$.

VII. Conclusions

In this paper, novel convex relaxations of *chance-constrained* problems are proposed. These problems are, in general, non-convex and difficult to solve. Hence, a novel concept, kinship function, is introduced and used to bound the probability of constraint violation. By using this bound, one is able to obtain a relaxation which is shown to be convex. Another concept introduced in the paper is the concept of optimal polynomial kinship functions. These members of the class of kinship functions can be efficiently computed *a priori* by using standard LMI solvers. Moreover, it is proven that, under "mild" conditions, as the degree of the optimal polynomial kinship function increases, the solution of the proposed relaxation converges to the solution of the robust optimization problem; i.e., the problem for which the probability of constraint violation is zero. To efficiently solve the relaxed problem, the so-called Smolyak quadrature formulae is used to compute the integral that bounds the probability of constraint violation. By using this approach, one is able to show that the computation complexity increases polynomially with uncertainty dimension and that the (sub)gradient of the constraints can be easily computed. Hence, one can use standard gradient descent algorithms to solve the proposed relaxed problem.

References

[1] B. R. Barmish and P. S. Shcherbakov. A dilation method for robustness problems with nonlinear parameter dependence. In *Proc. American Control Conference*, pages 3834–3839, 2003.

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

[3] O. Boni, A. Ben-Tal, and A. Nemirovski. Robust solutions to conic quadratic problems and their applications. *Optim. Eng.*, 9(1):1–18, 2008.

[4] S. Boyd, El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*. SIAM, Philadelphia, PA, 1994.

[5] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, 2004.

[6] G. Calafiore and M.C. Campi. The scenario approach to robust control design. *IEEE Trans. Automat. Contr.*, 51(5):742–753, 2006.

[7] G. Calafiore and F. Dabbene. Observer design with guaranteed RMS gain for linear parameter varying jump systems. In *Proc. American Control Conference*, pages 6073–6078, 2007.

[8] G. Calafiore and F. Dabbene (Eds.). *Probabilistic and Randomized Methods for Design under Uncertainty*. Springer-Verlag, London, 2006.

[9] G. Calafiore and B. T. Polyak. Stochastic algorithms for exact and approximate feasibility of robust LMIs. *IEEE Trans. Automat. Contr.*, 46(11):1755–1759, 2001.

[10] F. Dabbene, C. Feng, and C. Lagoa. Robust LMIs with polynomial dependence on the uncertainty. In *Proc. of the IEEE Conference on Decision and Control*, pages 5646–5651, 2007.

[11] F. Dabbene, C. Feng, and C. M. Lagoa. Robust and chance-constrained optimization under polynomial uncertainty. Technical report, 2008.

[12] E. F. Fama and K. R. French. Common risk factors in the returns on stocks and bonds. *J. Finan. Econ.*, 33(1):3–56, 1993.

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

[14] L. El Ghaoui, M. Oks, and F. Oustry. Worst-case value-at-risk and robust portfolio optimization: A conic programming approach. *Operations Research*, 51(4):543–556, 2003.

[15] Y. Nesterov. *High Performance Optimization*, chapter 17, pages 405–440. Kluwer Academic Publishers, The Netherlands, 2000.

[16] E. Novak and K. Ritter. Simple cubature formulas wigh high polynomial exactness. *Constructive Approximation*, 15:499–522, 1999.

[17] K. Petras. Fast calculation of coefficients in the Smolyak algorithm. *Numerical Algorithms*, 26:93–109, 2001.

[18] K. Petras. Smolyak cubature of given polynomial degree with few nodes for increasing dimension. *Numerische Mathematik*, 93:729–753, 2003.

[19] B. T. Polyak and R. Tempo. Probabilistic robust design with linear quadratic regulators. *Systems Control Lett.*, 43(5):343–353, 2001.

[20] S.A. Smolyak. Quadrature and interpolation formulas for tensor productsof certain classes of functions. *Doklady Akademii Nauk SSSR*, 4:240–243, 1963.

[21] R. Tempo, G. Calafiore, and F. Dabbene. *Randomized Algorithms for Analysis and Control of Uncertain Systems*. Communications and Control Engineering Series. Springer-Verlag, London, 2005.