STOCHASTIC POLYNOMIAL OPTIMIZATION

JIAWANG NIE, LIU YANG, AND SUHAN ZHONG

ABSTRACT. This paper studies stochastic optimization problems with polynomials. We propose an optimization model with sample averages and perturbations. The Lasserre type Moment-SOS relaxations are used to solve the sample average optimization. Properties of the optimization and its relaxations are studied. Numerical experiments are presented.

1. INTRODUCTION

Stochastic optimization is about functions that depend on random variables. A typical stochastic optimization problem is

$$(1.1) \qquad \min_{x \in K} \quad f(x) := \mathbb{E}[F(x, \xi)]$$

where $F : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}$ is a function in (x, ξ). The variable ξ is a random vector, and the decision variable $x \in \mathbb{R}^n$ is required to be contained in a set $K \subseteq \mathbb{R}^n$. In (1.1), the symbol $\mathbb{E}$ denotes the expectation of a function in the random vector ξ. Frequently used methods for solving stochastic optimization are often based on sample average approximation (SAA). We refer to [5, 8, 17, 24, 25, 33, 35, 36, 38, 39, 41] for related work on stochastic optimization. The SAA methods use sample averages to approximate the expectation function $f(x)$, transforming the stochastic optimization into deterministic optimization. Many classical SAA methods assume the objective functions are convex and are based on evaluations of gradients or subgradients. They can also be applied to nonconvex problems, however, the global optimality may not be guaranteed. There exists relatively less work on nonconvex stochastic optimization [1, 9, 10]. Generally, nonconvex stochastic optimization problems are computationally challenging, because the deterministic case is already difficult.

This paper discusses the special case of stochastic polynomial optimization, i.e., $F(x, \xi)$ is a polynomial function in x and K is a semialgebraic set defined by polynomials. When F does not depend on ξ, this is the case of polynomial optimization. The Lasserre type Moment-SOS relaxations are efficient and reliable for solving polynomial optimization [20, 23]. When F depends on the random vector ξ, the objective $f(x)$ is the expectation of $F(x, \xi)$ with respect to ξ. Hence, $f(x)$ is still a polynomial. When the distribution of ξ is known explicitly (e.g, Gaussian, Poisson, etc), the objective f can be expressed by integral formula. However, when the distribution of ξ is not known exactly, or its density function is too complicated for evaluating the expectation, it is not practical to get an explicit formula for f. In most methods for stochastic optimization, the objective f is approximated by

sample averages. In this article, we discuss how to use the efficient Moment-SOS relaxation for solving stochastic optimization with sample average approximation.

In stochastic polynomial optimization, we assume that

$$F(x, \xi) := \sum_{\alpha = (\alpha_1, \ldots, \alpha_n)} c_\alpha(\xi) x_1^{\alpha_1} \cdots x_n^{\alpha_n}$$

is a polynomial in $x \in \mathbb{R}^n$. Here, each coefficient $c_\alpha(\xi)$ is a measurable function in ξ. The feasible set K is assumed to be in the form

$$(1.2) \qquad K := \{x \in \mathbb{R}^n : g_1(x) \geq 0, \ldots, g_m(x) \geq 0\},$$

for given polynomials $g_1, \ldots, g_m$ in $x \in \mathbb{R}^n$. For each x, $F(x, \xi)$ is a measurable function in ξ. So the objective $f(x) := \mathbb{E}[F(x, \xi)]$ is also a polynomial. The stochastic optimization can be expressed as

$$(1.3) \qquad \begin{cases} \min & f(x) := \mathbb{E}\big[F(x, \xi)\big] \\ \text{s.t.} & g_1(x) \geq 0, \ldots, g_m(x) \geq 0. \end{cases}$$

The coefficients of the polynomial f are typically not known explicitly, because the true distribution of ξ is usually not known exactly. However, they can be estimated by sample average approximation. In applications, we can generate samples of ξ, say, N random samples $\xi^{(1)}, \xi^{(2)}, \ldots, \xi^{(N)}$. The expectation function $f(x) = \mathbb{E}[F(x, \xi)]$ can be approximated by the sample average

$$f_N(x) := \frac{1}{N} \sum_{k=1}^{N} F(x, \xi^{(k)}).$$

If each sample $\xi^{(k)}$ obeys the same distribution of ξ, then $\mathbb{E}[f_N(x)] = f(x)$. Furthermore, when all $\xi^{(k)}$ are independently identically distributed (i.i.d.), the Law of Large Numbers (LLN) (see [15]) implies that

$$f_N(x) \to f(x) \qquad \text{as } N \to \infty,$$

with probability one and under some regularity conditions. The resulting sample average approximation for (1.3) is

$$(1.4) \qquad \begin{cases} \min & f_N(x) \\ \text{s.t.} & g_1(x) \geq 0, \ldots, g_m(x) \geq 0. \end{cases}$$

This is also a polynomial optimization problem. It can be solved globally by the Lasserre type Moment-SOS hierarchy of relaxations [18].

Sample average approximation methods have good statistical properties. For convenience, denote by ϑ^*, ϑ_N the optimal values of (1.3) and (1.4) respectively, and denote by S, S_N their optimizer sets respectively. Assume that: i) f is continuous and S is nonempty; ii) there is a compact set $C \subseteq \mathbb{R}^n$ such that $S \subseteq C$ and f_N converges uniformly to f on C, with probability one; iii) for all N large enough, $\emptyset \neq S_N \subseteq C$. Then, it can be shown (see [39]) that $\vartheta_N \longrightarrow \vartheta^*$ and $\text{dist}(S_N, S) \longrightarrow 0$ [1] as $N \to \infty$, with probability one. Moreover, if $\xi^{(1)}, \xi^{(2)}, \ldots, \xi^{(N)}$ are independently identically distributed as ξ is, then $\mathbb{E}[\vartheta_N] \leq \mathbb{E}[\vartheta_{N+1}] \leq v^*$. That is, as the sample size Z increases, the sample average optimization gives better approximation for

---

[1]For sets $A, B \subseteq \mathbb{R}^n$, their distance is defined as

$$\text{dist}(A, B) := \max \left\{ \sup_{x \in A} \inf_{y \in B} \|x - y\|, \ \sup_{x \in B} \inf_{y \in A} \|x - y\| \right\}.$$

(1.3). We refer to [39] for more details about the convergence of the sample average optimization.

When $F(x, \xi)$ is a polynomial in x, the sample average approximation (1.4) is also a polynomial optimization problem. The Lasserre type Moment-SOS relaxations can be applied to solve it. However, the following concerns need to be addressed:

- For given samples $\xi^{(1)}, \ldots, \xi^{(N)}$, the optimizer set S_N of (1.4) may (or may not) be far away from the optimizer set S of (1.3). For instance, it is possible that (1.3) is bounded from below and has a global minimizer, while (1.4) is unbounded from below and has no global minimizers.
- The sample average $f_N(x)$ is only an approximation for the objective $f(x)$. Usually, the optimizer sets of (1.3) and (1.4) are not exactly same. Therefore, the sample average approximation (1.4) does not need to be solved exactly. However, we still expect that the optimizer set S_N of (1.4) (if it is nonempty) is a good approximation for the optimizer set S of (1.3), though they might be very different. Generally, the optimizer set S can not be determined exactly, unless the objective $f(x)$ can be determined exactly.

To address the above concerns, we propose the following perturbation sample average approximation (PSAA) model

$$
(1.5) \qquad \begin{cases} \min & f_N(x) + \epsilon \|[x]_{2d}\| \\ \text{s.t.} & g_1(x) \geq 0, \ldots, g_m(x) \geq 0, \end{cases}
$$

where d is the smallest integer such that

$$
2d \geq \max\{\deg(f_N), \deg(g_1), \ldots, \deg(g_m)\}
$$

and $[x]_{2d}$ is the vector of monomials in x and with degrees at most $2d$ (see (2.1)). The norm $\|\cdot\|$ is the standard Euclidean norm. We use the Lasserre type Moment-SOS relaxation of degree $2d$ for solving (1.5). A small parameter $\epsilon > 0$ is often selected, for (1.5) to approximate (1.3) well. In this article, we discuss properties of (1.5), as well as its Moment-SOS relaxations. The perturbation term $\epsilon\|[x]_{2d}\|$ plays an important role in sample average approximation. The paper is organized as follows. We review some basics for polynomial optimization in Section 2. The properties of the perturbed sample average optimization (1.5) are discussed in Section 3. The numerical experiments are given in Section 4.

2. Preliminaries

Notation. The symbols $\mathbb{N}, \mathbb{R}$ denote the set of nonnegative integers and real numbers, respectively. For given $x \in \mathbb{R}^n$ and a real scalar $r > 0$, $B(x, r)$ denotes the closed ball in $\mathbb{R}^n$ centered at x with radius r, under the standard Euclidean norm. For a real symmetric matrix X, we write $X \succeq 0$ (resp., $X \succ 0$) by meaning that X is positive semidefinite (resp., positive definite). The symbol $\mathbb{R}[x] := \mathbb{R}[x_1, \ldots, x_n]$ denotes the ring of polynomials with real coefficients and in $x := (x_1, \ldots, x_n)$. For a polynomial f, $\deg(f)$ refers to its total degree. For a tuple of polynomials $p = (p_1, p_2, \ldots, p_m)$, $\deg(p)$ refers to the maximum of the degrees of p_i. For a degree d, $\mathbb{R}[x]_d$ stands for the space of all real polynomials in x and of degrees no more than d. For a nonnegative integer vector $\alpha = (\alpha_1, ..., \alpha_n) \in \mathbb{N}^n$, denote

$$
|\alpha| := \alpha_1 + \cdots + \alpha_n, \quad x^\alpha := x_1^{\alpha_1} \cdots x_n^{\alpha_n}.
$$

For convenience, denote the monomial power set

$$\mathbb{N}_d^n := \{\alpha \in \mathbb{N}^n : |\alpha| \le d\}.$$

For a degree k, $[x]_k$ denotes the vector of all monomials of degrees at most k, ordered in the graded lexicographic ordering, i.e.,

(2.1) $$[x]_k := \begin{bmatrix} 1 & x_1 & \cdots & x_n & x_1^2 & x_1 x_2 & \cdots & x_n^k \end{bmatrix}^T.$$

(The superscript ^{T} denotes the transpose of a vector or matrix.) For $t \in \mathbb{R}$, $\lceil t \rceil$ denotes the smallest integer greater than or equal to t. For a function $p(\xi)$ in a random vector ξ, $\mathbb{E}[p(\xi)]$ stands for the expectation of $p(\xi)$, with respect to the distribution of ξ.

A polynomial σ is said to be a sum of squares (SOS) if $\sigma = s_1^2 + s_2^2 + \cdots + s_k^2$, for some $k \in \mathbb{N}$ and polynomials $s_1, s_2, ..., s_k \in \mathbb{R}[x]$. Clearly, if σ is SOS and has degree $2d$, then each s_i must have degree at most d. We use $\Sigma[x]$ to denote the cone of all SOS polynomials, and $\Sigma[x]_{2d}$ to denote the truncation of SOS polynomials in $\mathbb{R}[x]_{2d}$. Checking whether a polynomial is SOS or not can be done by solving a semidefintie program [18, 31].

For a tuple $g := (g_1, \ldots, g_m)$ of polynomials in $\mathbb{R}[x]$, its quadratic module is the set

$$Q(g) := \Sigma[x] + g_1 \cdot \Sigma[x] + \cdots + g_m \cdot \Sigma[x].$$

The $2d$-th truncation of $Q(g)$ is

$$Q(g)_{2d} := \Sigma[x]_{2d} + g_1 \cdot \Sigma[x]_{2d - \deg(g_1)} + \cdots + g_m \cdot \Sigma[x]_{2d - \deg(g_m)}.$$

It clearly holds the nesting relation of containment

$$\cdots \subseteq Q(g)_{2d} \subseteq Q(g)_{2d+2} \subseteq \cdots Q(g).$$

Indeed, each $Q(g)_{2d}$ is a convex cone of the space $\mathbb{R}[x]_{2d}$. The tuple g determines the semialgebraic set in (1.2). Obviously, if $f \in Q(g)$, then $f \ge 0$ on K. To ensure $f \in Q(g)$, we often require $f > 0$ on K. The quadratic module $Q(g)$ is said to be *archimedean* if there exists a single polynomial $p \in Q(g)$ such that the inequality $p(x) \ge 0$ defines a compact set in $\mathbb{R}^n$. If $Q(g)$ is archimedean, then the set K must be compact. The converse is not necessarily true. However, if K is compact (say, $K \subseteq B(0, R)$ for some radius R), one can always enforce $Q(g)$ to be archimedean by adding the redundant polynomial $R^2 - \|x\|^2$ to the tuple g. When $Q(g)$ is archimedean, if $f > 0$ on K, then we must have $f \in Q(g)$. This conclusion is referred to Putinar's Positivstellensatz, which was shown in [34]. Interestingly, when $f \ge 0$ on K, we still have $f \in Q(g)$, under some optimality conditions [28].

For a given dimension n and degree d, denote by $\mathbb{R}^{\mathbb{N}_d^n}$ the space of real vectors that are indexed by $\alpha \in \mathbb{N}_d^n$, i.e.,

$$\mathbb{R}^{\mathbb{N}_d^n} := \{y = (y_\alpha)_{\alpha \in \mathbb{N}_d^n} : y_\alpha \in \mathbb{R}\}.$$

Each vector in $\mathbb{R}^{\mathbb{N}_d^n}$ is called a truncated multi-sequence (tms) of degree d. A tms $y \in \mathbb{R}^{\mathbb{N}_d^n}$ gives the linear functional $\mathscr{R}_y$ acting on $\mathbb{R}[x]_d$ as

(2.2) $$\mathscr{R}_y\Big(\sum_{\alpha \in \mathbb{N}_d^n} f_\alpha x^\alpha\Big) := \sum_{\alpha \in \mathbb{N}_d^n} f_\alpha y_\alpha.$$

The $\mathscr{R}_y$ is called a Riesz functional. For $f \in \mathbb{R}[x]_d$ and $y \in \mathbb{R}^{\mathbb{N}_d^n}$, we denote

(2.3) $$\langle f, y \rangle := \mathscr{R}_y(f).$$

The tms $y \in \mathbb{R}^{\mathbb{N}_d^n}$ is said to admit a representing measure supported in a set $T \subseteq \mathbb{R}^n$ if there exists a Borel measure μ, supported in T, such that $y_\alpha = \int x^\alpha \mathrm{d}\mu$ for all $\alpha \in \mathbb{N}_d^n$. This is equivalent to that $\langle f, y \rangle = \int f(x) \mathrm{d}\mu$ for all $f \in \mathbb{R}[x]_d$. Such μ is called a T-representing measure for y. We refer to [7, 12, 29] for recent work on truncated moment problems.

For a polynomial $p \in \mathbb{R}[x]_{2d}$, the dth *localizing matrix* of p associated to a tms $y \in \mathbb{R}^{\mathbb{N}_{2d}^n}$, is the symmetric matrix $L_p^{(d)}[y]$ such that

$$(2.4) \qquad vec(a)^T \left(L_p^{(d)}[y] \right) vec(b) = \mathscr{R}_y(pab)$$

for all polynomials $a, b \in \mathbb{R}[x]_t$, with $t = d - \lceil \deg(p)/2 \rceil$. In the above, the $vec(a)$ denotes the coefficient vector of the polynomial a. For instance, when $n = 3$ and $p = x_1 x_2 - x_3^3$, for $y \in \mathbb{R}^{\mathbb{N}_6^3}$, we have

$$L_p^{(3)}[y] = \begin{bmatrix} y_{110} - y_{003} & y_{210} - y_{103} & y_{120} - y_{013} & y_{111} - y_{004} \\ y_{210} - y_{103} & y_{310} - y_{203} & y_{220} - y_{113} & y_{211} - y_{104} \\ y_{120} - y_{013} & y_{220} - y_{113} & y_{130} - y_{023} & y_{121} - y_{014} \\ y_{111} - y_{004} & y_{211} - y_{104} & y_{121} - y_{014} & y_{112} - y_{005} \end{bmatrix}.$$

For the special case of constant one polynomial $p = 1$, $L_1^{(d)}[y]$ is reduced to the so-called *moment matrix*

$$(2.5) \qquad M_d[y] := L_1^{(d)}[y].$$

The columns and rows of $L_p^{(d)}[y]$, as well as $M_d[y]$, are labelled by $\alpha \in \mathbb{N}^n$ with $2|\alpha| + \deg(p) \leq 2d$. For instance, for $n = 3$ and $y \in \mathbb{R}^{\mathbb{N}_4^3}$, we have

$$M_2[y] = \begin{bmatrix} y_{000} & y_{100} & y_{010} & y_{001} & y_{200} & y_{110} & y_{101} & y_{020} & y_{011} & y_{002} \\ y_{100} & y_{200} & y_{110} & y_{101} & y_{300} & y_{210} & y_{201} & y_{120} & y_{111} & y_{102} \\ y_{010} & y_{110} & y_{020} & y_{011} & y_{210} & y_{120} & y_{111} & y_{030} & y_{021} & y_{012} \\ y_{001} & y_{101} & y_{011} & y_{002} & y_{201} & y_{111} & y_{102} & y_{021} & y_{012} & y_{003} \\ y_{200} & y_{300} & y_{210} & y_{201} & y_{400} & y_{310} & y_{301} & y_{220} & y_{211} & y_{202} \\ y_{110} & y_{210} & y_{120} & y_{111} & y_{310} & y_{220} & y_{211} & y_{130} & y_{121} & y_{112} \\ y_{101} & y_{201} & y_{111} & y_{102} & y_{301} & y_{211} & y_{202} & y_{121} & y_{112} & y_{103} \\ y_{020} & y_{120} & y_{030} & y_{021} & y_{220} & y_{130} & y_{121} & y_{040} & y_{031} & y_{022} \\ y_{011} & y_{111} & y_{021} & y_{012} & y_{211} & y_{121} & y_{112} & y_{031} & y_{022} & y_{013} \\ y_{002} & y_{102} & y_{012} & y_{003} & y_{202} & y_{112} & y_{103} & y_{022} & y_{013} & y_{004} \end{bmatrix}.$$

Suppose $g := (g_1, \ldots, g_m)$ is a tuple of polynomials in $\mathbb{R}[x]_{2d}$. Consider the cone of tms of degree $2d$

$$(2.6) \qquad \mathscr{S}(g)_{2d} := \left\{ y \in \mathbb{R}^{\mathbb{N}_{2d}^n} \,\middle|\, M_d[y] \succeq 0, \, L_{g_1}^{(d)}[y] \succeq 0, \ldots, L_{g_m}^{(d)}[y] \succeq 0 \right\}.$$

It is a closed convex cone in $\mathbb{R}^{\mathbb{N}_{2d}^n}$. Consider the projection map:

$$(2.7) \qquad \pi : \mathbb{R}^{\mathbb{N}_{2d}^n} \to \mathbb{R}^n, \quad y \mapsto u = (y_{e_1}, \ldots, y_{e_n}).$$

Let K be the semialgebraic set as in (1.2), then

$$(2.8) \qquad K \subseteq \pi\left(\mathscr{S}(g)_{2d} \cap \{y_0 = 1\} \right).$$

This is because for each $u \in K$, the tms $y := [u]_{2d}$ belongs to $\mathscr{S}(g)_{2d}$ and $\pi(y) = u$. Therefore, the linear section $\{y_0 = 1\}$ of the cone $\mathscr{S}(g)_{2d}$ is a lifted convex relaxation of the set K. The cone $\mathscr{S}(g)_{2d}$ and the quadratic module $Q(g)_{2d}$ are dual to each other. This is because $\langle f, y \rangle \geq 0$ for all $f \in Q_{2d}(g)$ and $y \in \mathscr{S}(g)_{2d}$.

We refer to [20, 23, 30] for these basic properties. Interestingly, the containment in (2.8) is an equality when the polynomials g_i are SOS-concave [11].

There exists much work for polynomial optimization. The Lasserre type Moment-SOS hierarchy of relaxations were introduced in [18]. The Moment-SOS hierarchy was proved to have finite convergence under some optimality conditions [16, 19, 28]. The flat extension or flat truncation condition can be used to certify its convergence [7, 13, 26]. For unconstrained optimization, the performance of the standard SOS relaxations was studied in [31, 32]. For the special case of finite feasible sets, the convergence was studied in [21, 22, 27]. We refer to [3, 20, 23, 37] for more detailed introductions to polynomial optimization.

3. The sample average optimization

Let $\xi^{(1)}, \ldots, \xi^{(N)}$ be given samples for the random vector ξ. Consider the sample average function

$$f_N(x) := \frac{1}{N} \sum_{k=1}^{N} F(x, \xi^{(k)}).$$

When $F(x, \xi)$ is a polynomial in $x \in \mathbb{R}^n$, $f_N(x)$ is also a polynomial in x. We assume that the feasible set K is given as in (1.2), for a tuple $g := (g_1, \ldots, g_m)$ of polynomials. Let d be the degree:

$$(3.1) \qquad d = \left\lceil \frac{1}{2} \max\{\deg(f_N), \deg(g_1), \ldots, \deg(g_m)\} \right\rceil.$$

Instead of solving (1.4) directly, we propose to solve the sample average optimization with perturbation

$$(3.2) \qquad \left\{ \begin{array}{ll} \min & f_N(x) + \epsilon \|[x]_{2d}\| \\ \text{s.t.} & g_1(x) \geq 0, \ldots, g_m(x) \geq 0, \end{array} \right.$$

for a small parameter $\epsilon > 0$. The Lasserre type moment relaxation can be applied to solve (3.2). Recall the notation $[x]_{2d}$, $M_d[y]$, $L_{g_i}^{(d)}[y]$ as in Section 2. Observe that

$$f_N(x) = \langle f_N, [x]_{2d} \rangle, \quad M_d\big[[x]_{2d}\big] \succeq 0, \quad L_{g_i}^{(d)}\big[[x]_{2d}\big] \succeq 0$$

for all $x \in K$ and all $i = 1, \ldots, m$. If we replace the monomial vector $[x]_{2d}$ by a tms $y \in \mathbb{R}^{\mathbb{N}_{2d}^n}$, then (3.2) is relaxed to the following convex optimization

$$(3.3) \qquad \left\{ \begin{array}{ll} \min & \langle f_N, y \rangle + \epsilon \|y\| \\ \text{s.t.} & M_d[y] \succeq 0, \ L_{g_i}^{(d)}[y] \succeq 0 \ (i = 1, \ldots, m), \\ & y_0 = 1, \ y \in \mathbb{R}^{\mathbb{N}_{2d}^n}. \end{array} \right.$$

It is a semidefinite program, with a norm function in the objective. The relaxation (3.3) is said to be *tight* if its optimal value is the same as that of (3.2). In this paper, we choose $\| \cdot \|$ to be the standard Euclidean norm, but any other kind of vector norms can also be used. The equality constraint $y_0 = 1$ means that the first entry of y is equal to one. The set of all y satisfying linear matrix inequalities in (3.3) is just the cone $\mathscr{S}(g)_{2d}$, defined as in (2.6). The cone $\mathscr{S}(g)_{2d}$ and the truncated quadratic module $Q(g)_{2d}$ are dual to each other. Therefore, the Lagrange function for (3.3) is

$$\begin{array}{rcl} \mathcal{L}(y, q, \gamma) & = & \langle f_N, y \rangle + \epsilon \|y\| - \langle q, y \rangle - \gamma(y_0 - 1) \\ & = & \langle f_N - q - \gamma, y \rangle + \epsilon \|y\| + \gamma, \end{array}$$

for dual variables $q \in Q(g)_{2d}$ and $\gamma \in \mathbb{R}$. The function $\mathcal{L}(y, q, \gamma)$ has a finite minimum value for $y \in \mathbb{R}^{\mathbb{N}_{2d}^n}$ if and only if

$$\|vec(f_N - q - \gamma)\| \le \epsilon,$$

for which case the minimum value is γ. (The $vec(p)$ denotes the coefficient vector of p.) Therefore, the dual optimization problem of (3.3) is

(3.4)
$$\begin{cases} \max & \gamma \\ \text{s.t.} & f_N - p - \gamma \in Q(g)_{2d}, \\ & \|vec(p)\| \le \epsilon, p \in \mathbb{R}[x]_{2d}. \end{cases}$$

Because the sample average $f_N(x)$ is only an approximation for $f(x)$, it is possible that there is no scalar γ such that $f_N - \gamma \in Q(g)_{2d}$. The perturbation term $\epsilon\|y\|$ in (3.3) motivates us to find the maximum γ such that $f_N - p - \gamma \in Q(g)_{2d}$, for some polynomial p whose coefficient vector has a small norm. This leads to the following algorithm.

Algorithm 3.1. *Generate samples $\xi^{(1)}, \ldots, \xi^{(N)}$, according to the distribution of ξ. Choose a small perturbation parameter $\epsilon > 0$.*

Step 1 *Compute the sample average $f_N = N^{-1}\sum_{k=1}^{N} F(x, \xi^{(k)})$.*
Step 2 *Solve the semidefinite relaxation problem (3.3). If (3.3) is infeasible, increase the value of ϵ (e.g., let $\epsilon := 2\epsilon$), until (3.3) has a minimizer, which we denote as y^*.*
Step 3 *Let $u = \pi(y^*)$, where π is the projection map in (2.7), or equivalently, let*

$$u = (y_{e_1}^*, \ldots, y_{e_n}^*).$$

Output u as a candidate minimizer for the sample average optimization with perturbation (3.2), and stop.

For $\epsilon > 0$, the minimizer of the relaxation (3.3) is always unique (if it exists), because its objective is strictly convex. Our numerical experiments demonstrate that Algorithm 3.1 is efficient for solving (3.2).

Theorem 3.2. *Assume that u^* is a minimizer of (3.2) and y^* is a minimizer of (3.3). Then, for $\epsilon > 0$, the relaxation (3.3) is tight if and only if $rank\, M_d[y^*] = 1$. In particular, for the case $rank\, M_d[y^*] = 1$, the point $u = \pi(y^*)$ is a minimizer of (3.2).*

Proof. Let ϑ_1, ϑ_2 be optimal values of (3.2) and (3.3) respectively.

"$\Leftarrow$" It is clear that $\vartheta_1 \ge \vartheta_2$. If $rank M_d[y^*] = 1$, then for $u = \pi(y^*)$ one can show that $M_d[y^*] = [u]_d([u]_d)^T$. Hence, $y^* = [u]_{2d}$, $\langle f_N, y^* \rangle = f_N(u)$, and each $g_i(u) \ge 0$ (see [13, 26]). So, u is a feasible point of (3.2) and

$$\vartheta_1 \le f_N(u) + \epsilon\|[u]_{2d}\| = \langle f_N, y^* \rangle + \epsilon\|y^*\| = \vartheta_2.$$

Therefore, $\vartheta_1 = \vartheta_2$, u is a minimizer of (3.2), and the relaxation (3.3) is tight.

"$\Rightarrow$" Let $\tilde{y} := [u^*]_{2d}$, then $f_N(u^*) = \langle f_N, \tilde{y} \rangle$ and $\|[u^*]_{2d}\| = \|\tilde{y}\|$. If the relaxation (3.3) is tight, then $\vartheta_1 = \vartheta_2$ and $\tilde{y}$ is a minimizer of (3.3). For $\epsilon > 0$, the objective of (3.3) is strictly convex, so its minimizer must be unique. Hence, $\tilde{y} = y^*$ and

$$M_d[y^*] = M_d[\tilde{y}] = [u^*]_d([u^*]_d)^T.$$

Therefore, $\text{rank}\, M_d[y^*] = \text{rank}\, M_d[\tilde{y}] = 1$. $\qquad\square$

When the sample average $f_N(x)$ is unbounded from below on the feasible set K, the moment relaxation (3.3) might still be unbounded from below if $\epsilon > 0$ is small. However, if $\epsilon > 0$ is big, then (3.3) must be feasible and has a minimizer. Indeed, we have the following theorem.

Theorem 3.3. *Suppose the feasible set K has nonempty interior. If $\epsilon > 0$ is big, both (3.3) and (3.4) have optimizers and their optimal values are the same.*

Proof. When K has nonempty interior, the quadratic module $Q(g)_{2d}$ is a closed cone (see [23, Theorem 3.49]) and the cone $\mathscr{S}(g)_{2d}$ has nonempty interior. For instance, let ν be the Gaussian measure, then the tms

$$\hat{y} := \frac{1}{\nu(K)} \int_K [x]_{2d} d\nu(x)$$

is an interior point of the cone $\mathscr{S}(g)_{2d}$. In other words, $M_d[\hat{y}] \succ 0$ and all $L_{g_i}^{(d)}[\hat{y}] \succ 0$. This is because $\int_K p^2 d\nu > 0$ and $\int_K g_i p^2 d\nu > 0$ for all nonzero polynomials p. Moreover, $\hat{y}_0 = 1$. The convex relaxation (3.3) is strictly feasible (i.e., there is a feasible y such that each matrix in (3.3) is positive definite). When $\epsilon > 0$ is big, the SOS relaxation (3.4) is also strictly feasible. For instance, for the choice

$$\epsilon > \|f_N - [x]_d^T[x]_d\|, \ \hat{p} = f_N - [x]_d^T[x]_d, \ \hat{\gamma} = 0,$$

we have that

$$f_N - \hat{p} - \hat{\gamma} = [x]_d^T[x]_d \in \text{int}\left(\Sigma[x]_{2d}\right) \subseteq \text{int}\left(Q(g)_{2d}\right).$$

In the above, int denotes the interior of a set. Therefore, for big $\epsilon > 0$, both (3.3) and (3.4) have strictly feasible points. By the strong duality theorem (see [2, 4]), they have the same optimal value and they both achieve the optimal value, i.e., they have optimizers. □

In applications, however, we often choose a small $\epsilon > 0$, because we expect that (3.2) is a good approximation for (1.4). In (3.3), the value of ϵ affects the performance of (3.3). When $\epsilon > 0$ is too small, (3.3) might be unbounded from below and has no minimizers. If $\epsilon > 0$ is big, (3.3) might give a loose approximation for (1.4). For efficiency, we often anticipate the smallest value of ϵ such that (3.3) is bounded from below and has a minimizer. When K has nonempty interior, the relaxation (3.3) is strictly feasible, i.e., there exists $\hat{y}$ such that all the matrices $M_d[\hat{y}]$ and $L_{g_i}^{(d)}[\hat{y}]$ are positive definite. Therefore, the strong duality holds between (3.3) and (3.4). To ensure that (3.3) is solvable (i.e., it has a minimizer), the dual optimization problem (3.4) needs to be feasible. Consider the optimization problem

$$(3.5) \qquad \begin{cases} \epsilon^* := \min & \|vec(p)\| \\ \text{s.t.} & f_N - p - \gamma \in Q(g)_{2d}, \\ & \gamma \in \mathbb{R}, \ p \in \mathbb{R}[x]_{2d}. \end{cases}$$

The above is a convex optimization problem with semidefinite constraints. In computational practice, we often choose $\epsilon > 0$ in a heuristic way, e.g., $\epsilon = 10^{-2}$. If such ϵ is not enough, we can increase its value until (3.3) performs well.

4. NUMERICAL EXPERIMENTS

This section gives numerical experiments of applying Algorithm 3.1 to solve stochastic polynomial optimization. The computation is implemented in MATLAB R2018a, in a Laptop with CPU 8th Generation Intel Core i5-8250U and RAM 16 GB. The moment relaxation (3.3) is solved by the software `GloptiPoly 3` [14], which calls the semidefinite program solver `SeDuMi` [40]. The computational results are displayed with four decimal digits. We use "PSAA" to denote the perturbation sample average approximation model (3.2). Its relaxation (3.3) is said to be *solvable* if it has a minimizer y^*. For such a case, let u be the point given in Step 3 of Algorithm 3.1, i.e., $u = \pi(y^*)$. Otherwise, (3.3) is said to be *not solvable* and we use "n.a." to indicate the relevant values are not available. We use f_{min} and v^* to denote the optimal value and the minimizer of (1.3) respectively. The symbol $\bar{\xi}$ stands for the sample average of the random vector $\xi \in \mathbb{R}^r$, while $\bar{\xi}_i$ refers to its ith entry:

$$\bar{\xi} = \frac{1}{N} \sum_{k=1}^{N} \xi^{(k)}, \quad \bar{\xi}_i = \frac{1}{N} \sum_{k=1}^{N} \xi_i^{(k)}, \ i = 1, \cdots, r.$$

In our numerical examples, we use the following classical distributions for random variables (see [6]; let δ_a denote the Dirac function supported at a):

- $Ber(p)$ denotes the *Bernoullian distribution* with success probability p, whose density function is $(1-p) \cdot \delta_0 + p \cdot \delta_1$.
- $Geo(p)$ denotes the *geometric distribution* with success probability p, whose density function is $\sum_{n=0}^{\infty}(1-p)^n p \cdot \delta_n$.
- $\mathcal{P}(\lambda)$ denotes the *Poisson distribution* with parameter $\lambda > 0$, whose density function is $\sum_{n=0}^{\infty} e^{-\lambda} \frac{\lambda^n}{n!} \cdot \delta_n$.
- $\mathcal{U}(a, b)$, with $a < b$, denotes the *uniform distribution* on $[a, b]$.
- $\mathcal{N}(\mu, P)$ denotes the *normal distribution* with the expectation μ and co-variance matrix P.

Example 4.1. Consider the stochastic optimization problem

$$(4.1) \qquad \begin{aligned} \min_{x \in \mathbb{R}^4} \quad & f(x) := \mathbb{E}[G(x) + H(x, \xi)] \\ \text{s.t.} \quad & x_1 - 1 \geq 0, \ x_2 - 1/2 \geq 0, \ x_3 - 1/3 \geq 0, \ x_4 - 1/4 \geq 0, \end{aligned}$$

where

$$G(x) = (x_1^2 - 2x_2^2)^2 + x_3(2x_3^2 - 3x_1 x_2 + x_4^2)(x_4^2 - 3x_1 x_2) - x_4 x_3^3 (2x_1^3 - x_3^3),$$
$$H(x, \xi) = \xi_1 x_3^5 + \xi_2 x_1^6 x_4, \ \xi_1 \sim \mathcal{U}(0, 2), \ \xi_2 \sim \mathcal{U}(0, 2).$$

We have $\mathbb{E}(\xi_1) = \mathbb{E}(\xi_2) = 1$. The constraining polynomial tuple is

$$g = (x_1 - 1, x_2 - 1/2, x_3 - 1/3, x_4 - 1/4).$$

The feasible set K is a polyhedron and the exact objective is

$$f = (x_1^2 - 2x_2^2)^2 + x_3(x_3^2 - 3x_1 x_2 + x_4^2)^2 + x_4(x_1^3 - x_3^3)^2 \in Q(g)_8.$$

For the optimization (1.3), the optimal value and the minimizer are

$$f_{min} = 8.4455e - 07, \quad v^* = (1.0031, 0.7093, 1.0031, 1.0622).$$

Since the approximation polynomial f_N is uniquely determined by $\bar{\xi}_1$ and $\bar{\xi}_2$, we can pick some typical sample averages to explore the performance of our optimization

model (3.2) and its relaxation (3.3). For instance, we consider the samples of ξ such that
$$|\bar{\xi}_1 - \mathbb{E}(\xi_1)| = |\bar{\xi}_2 - \mathbb{E}(\xi_2)| = 10^{-2}.$$
There are four cases of the signs:
$$\text{I}: \bar{\xi}_1 - \mathbb{E}(\xi_1) > 0, \bar{\xi}_2 - \mathbb{E}(\xi_2) > 0, \quad \text{II}: \bar{\xi}_1 - \mathbb{E}(\xi_1) < 0, \bar{\xi}_2 - \mathbb{E}(\xi_2) > 0,$$
$$\text{III}: \bar{\xi}_1 - \mathbb{E}(\xi_1) < 0, \bar{\xi}_2 - \mathbb{E}(\xi_2) < 0, \quad \text{IV}: \bar{\xi}_1 - \mathbb{E}(\xi_1) > 0, \bar{\xi}_2 - \mathbb{E}(\xi_2) < 0.$$

We solve the optimization model (3.2) for each case. The computational results are reported in Table 1. The PSAA model (3.2) has clear advantages for cases III and IV, when they are compared to the unperturbed case (i.e, $\epsilon = 0$). It gives reliable optimizers, while the classical SAA model (1.4) does not return good ones.

TABLE 1. Performance of PSAA for Example 4.1

	ϵ	0	10^{-4}	10^{-3}	10^{-2}	10^{-1}
I	solvable?	yes	yes	yes	yes	yes
	time(sec.)	1.13	1.34	1.23	1.23	1.23
	$|\langle f_N, y^* \rangle - f_N(u)|$	1.29e-06	6.41e-07	2.38e-07	1.30e-07	4.75e-08
	$|\langle f_N, y^* \rangle - f_{min}|$	2.05e-02	2.05e-02	2.07e-02	3.41e-02	3.33e-01
II	solvable?	yes	yes	yes	yes	yes
	time(sec.)	1.12	1.29	1.23	1.21	1.22
	$|\langle f_N, y^* \rangle - f_N(u)|$	9.15e-07	6.41e-07	2.06e-07	1.27e-07	1.53e-08
	$|\langle f_N, y^* \rangle - f_{min}|$	4.84e-04	4.86e-04	6.91e-04	1.38e-02	3.13e-01
III	solvable?	no	yes	yes	yes	yes
	time(sec.)	2.14	2.78	1.40	1.16	1.15
	$|\langle f_N, y^* \rangle - f_N(u)|$	n.a.	1.13e+03	7.50e-08	1.25e-07	8.22e-09
	$|\langle f_N, y^* \rangle - f_{min}|$	n.a.	7.14e+05	3.85e-02	7.00e-03	2.94e-01
IV	solvable?	no	yes	yes	yes	yes
	time(sec.)	2.19	2.79	1.40	1.14	1.16
	$|\langle f_N, y \rangle - f_N(u)|$	n.a.	1.35e+03	1.89e-07	1.26e-07	1.54e-08
	$|\langle f_N, y \rangle - f_{min}|$	n.a.	6.52e+05	4.73e-04	1.32e-02	3.14e-01

Example 4.2. Consider the stochastic polynomial optimization

$$\min_{x \in \mathbb{R}^4} \quad f(x) := \mathbb{E}[F(x) + H(x, \xi)]$$

(4.2)
$$\text{s.t.} \quad x_1 x_3 + 1 \geq x_2^2 + x_4^2, \ x_2 x_3 - x_1 x_4 + 2 \geq 0,$$
$$x_1^3 + x_2^3 + x_3^3 + x_4^3 \leq 8,$$
$$x_1 \geq 0, \ x_2 \geq 0, \ x_3 \geq 0, \ x_4 \geq 0,$$

where

$$G(x) = x_1^2 x_2^2 + x_2^2 x_3^2 + (1 - x_2 x_3)^2 + (3 - x_1 x_4)^2 + x_1 x_2 x_3 x_4,$$
$$H(x, \xi) = \xi_1 x_1 x_2^2 x_3 + \xi_2 x_2^2 x_4^2,$$
$$\xi \sim \mathcal{N}\left(\begin{bmatrix} -0.41 \\ -2.50 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right).$$

If $f(x)$ is evaluated exactly, the optimal value and the minimizer of (1.3) are

$$f_{min} = 1.0655, \quad v^* = (1.5829, 0.6427, 0.9316, 1.4358).$$

As in Example 4.1, we explore the performance of (3.2)-(3.3) for the following cases of samples:

$$\text{I} : \bar{\xi}_1 = \mathbb{E}(\xi_1) - 10^{-2}, \ \bar{\xi}_2 = \mathbb{E}(\xi_2) - 10^{-2},$$
$$\text{II} : \bar{\xi}_1 = \mathbb{E}(\xi_1) - 10^{-2}, \ \bar{\xi}_2 = \mathbb{E}(\xi_2),$$
$$\text{III} : \bar{\xi}_1 = \mathbb{E}(\xi_1), \ \bar{\xi}_2 = \mathbb{E}(\xi_2) - 10^{-2}.$$

The computational results are reported in Table 2. The PSAA model (3.2) performs better than the classical SAA model (1.4) (i.e. $\epsilon = 0$). For all these cases, (3.2) gives more reliable optimizers. Moreover, solving the relaxation (3.3) costs less computational time for cases I and II.

TABLE 2. Performance of PSAA for Example 4.2

	ϵ	0	10^{-4}	10^{-3}	10^{-2}
I	solvable?	yes	yes	yes	yes
	time(sec.)	0.37	0.29	0.15	0.26
	$\lvert\langle f_N, y^*\rangle - f_N(u)\rvert$	1.81e+05	1.29e-08	5.28e-09	9.53e-09
	$\lvert\langle f_N, y^*\rangle - f_{min}\rvert$	1.81e+05	1.48e-02	1.48e-02	1.47e-02
II	solvable?	yes	yes	yes	yes
	time(sec.)	0.23	0.12	0.14	0.12
	$\lvert\langle f_N, y^*\rangle - f_N(u)\rvert$	1.87e+05	1.29e-08	5.61e-09	9.64e-09
	$\lvert\langle f_N, y^*\rangle - f_{min}\rvert$	1.87e+05	6.13e-03	6.12e-03	6.12e-03
III	solvable?	yes	yes	yes	yes
	time(sec.)	0.12	0.13	0.10	0.09
	$\lvert\langle f_N, y^*\rangle - f_N(u)\rvert$	5.58e-02	1.29e-08	5.61e-09	9.70e-09
	$\lvert\langle f_N, y^*\rangle - f_{min}\rvert$	1.40e-02	8.56e-03	8.56e-03	8.55e-03

Example 4.3. Consider the stochastic optimization

$$(4.3) \qquad \begin{aligned} &\min_{x \in \mathbb{R}^2} \quad \mathbb{E}\big\{ F(x, \xi) = \xi_1 x_1^4 x_2^2 + \xi_2 x_1^2 x_2^4 - \xi_3 x_1 x_2^3 + \xi_4 x_1 x_2 \big\} \\ &\text{s.t.} \quad 0 \le x_1 \le 2, \ x_1 + x_2 \le 4, \ x_1 x_2 \le 8, \end{aligned}$$

where

$$\xi \sim \mathcal{N}\left(\begin{bmatrix} 1 \\ 1 \\ 3 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 & 0.1 & 0.3 & 0.2 \\ 0.1 & 1 & 0.4 & 0.3 \\ 0.3 & 0.4 & 1 & 0.2 \\ 0.2 & 0.3 & 0.2 & 1 \end{bmatrix} \right).$$

The exact objective $f(x) = \mathbb{E}[F(x, \xi)]$ can be evaluated as

$$f(x) = x_1^4 x_2^2 + x_1^2 x_2^4 - 3x_1 x_2^3 + x_1 x_2.$$

For the optimization (1.3), its optimal value and minimizer are

$$f_{min} = -27.8444, \quad v^* = (0.3442, \ 3.6558).$$

For convenience, we use "$\bar{\xi} = \mathbb{E}(\xi) + o(10^{-3})$" to denote a random sample average of size $N = 1000$ with error in the order of 10^{-3}. Consider two cases:

$$\text{I} : \bar{\xi} = \mathbb{E}(\xi), \quad \text{II} : \ \bar{\xi} = \mathbb{E}(\xi) + o(10^{-3}).$$

The numerical results are reported in Table 3. The PSAA model (3.2) gives a better optimizer than the unperturbed one, for both cases.

TABLE 3. Performance of PSAA for Example 4.3

	ϵ	0	10^{-4}	10^{-3}
I	solvable?	yes	yes	yes
	time(sec.)	0.17	0.13	0.12
	$\lvert\langle f_N, y^*\rangle - f_N(u)\rvert$	1.86e+03	4.18e-06	3.55e-06
	$\lvert\langle f_N, y^*\rangle - f_{min}\rvert$	3.44e+01	2.09e-04	2.30e-02
II	solvable?	yes	yes	yes
	time(sec.)	0.13	0.14	0.16
	$\lvert\langle f_N, y^*\rangle - f_N(u)\rvert$	1.85e+03	4.17e-06	3.56e-06
	$\lvert\langle f_N, y^*\rangle - f_{min}\rvert$	3.44e+01	1.29e-02	9.87e-03

Example 4.4. Consider the stochastic polynomial optimization

$$(4.4) \qquad \begin{aligned} \min_{x \in \mathbb{R}^3} \quad & f(x) := \mathbb{E}[G(x) + H(x, \xi)] \\ \text{s.t.} \quad & x_1 \geq 0, \; x_2 \geq 0, \; x_3 \geq 0, \; 1 - \mathbf{1}^T x \geq 0, \end{aligned}$$

where

$$G(x) = x_1^4 + x_1 x_2 x_3 + x_3(1 - x_1^2 - x_2^2),$$
$$H(x, \xi) = 2\xi_1 x_2^4 - 4\xi_1 x_1^2 x_2^2 - \xi_2 x_1 x_2,$$
$$\xi_1 \sim Ber(0.5), \quad \xi_2 \sim Geo(0.5).$$

The feasible set K is a simplex, which is compact and satisfies the archimedean condition. For all samples $\xi^{(i)}$, the sample average $f_N(x)$ is bounded from below on K and it has a minimizer. For this example, $\mathbb{E}(\xi_1) = 0.5$ and $\mathbb{E}(\xi_2) = 2$, so

$$f(x) = (x_1^2 - x_2^2)^2 + x_3(1 - x_1^2 - x_2^2) - x_1 x_2(2 - x_3).$$

The optimal value and minimizer of (1.3) are

$$f_{min} = -0.5, \quad v^* = (0.5, \, 0.5, \, 0).$$

We consider two cases of samples

$$\text{I}: \; \bar{\xi}_1 = \mathbb{E}(\xi_1) + 10^{-3}, \; \bar{\xi}_2 = \mathbb{E}(\xi_2), \; \epsilon^* \approx 0.001155;$$
$$\text{II}: \; \bar{\xi}_1 = \mathbb{E}(\xi_1), \; \bar{\xi}_2 = \mathbb{E}(\xi_2) + 10^{-3}, \; \epsilon^* \approx 7.5875 \times 10^{-9}.$$

In the above, ϵ^* is the minimum value of (3.5). The numerical results are reported

TABLE 4. Performance of PSAA for Example 4.4.

	ϵ	0	0.0012	0.004	0.008
I	solvable?	no	yes	yes	yes
	time(sec.)	0.12	0.10	0.07	0.09
	$\lvert\langle f_N, y^*\rangle - f_N(u)\rvert$	n.a.	5.31e-03	3.36e-04	1.09e-04
	$\lvert\langle f_N, y^*\rangle - f_{min}\rvert$	n.a.	5.44e-03	4.61e-04	2.34e-04
	ϵ	0	10^{-4}	10^{-3}	10^{-2}
II	solvable?	yes	yes	yes	yes
	time(sec.)	0.08	0.08	0.07	0.06
	$\lvert\langle f_N, y^*\rangle - f_N(u)\rvert$	5.03e-09	1.84e-09	1.61e-09	1.86e-09
	$\lvert\langle f_N, y^*\rangle - f_{min}\rvert$	2.50e-04	2.50e-04	2.50e-04	2.50e-04

in Table 4. The PSAA model (3.2) performs very well for both cases. Compared with the classical SAA model (1.4) (i.e., $\epsilon = 0$), it has quite clear advantages for case I. It successfully returned a good minimizer, while (1.4) is unbounded from below and does not return a minimizer.

Example 4.5. Consider the unconstrained stochastic optimization

$$(4.5) \qquad \min_{x \in \mathbb{R}^4} \ \mathbb{E}[G(x) + H(x, \xi)]$$

where

$$\xi \sim \mathcal{P}(2), \ G(x) = (x_3 - x_4)^4 + (x_1 + x_2)^4 + x_1^2 + x_2^2 + x_3^2 + x_4^2,$$
$$H(x, \xi) = \xi - (\xi^2 - 2\xi)(x_1 - x_4) - 2(\xi - 1)(x_3 - x_4)^2(x_1 + x_2)^2.$$

Evaluating the expectation, we get $\mathbb{E}(\xi) = 2, \mathbb{E}(\xi^2) = 6$ and

$$f(x) = [(x_3 - x_4)^2 - (x_1 + x_2)^2]^2 + (x_1 - 1)^2 + (1 + x_4)^2 + x_2^2 + x_3^2.$$

The optimal value and minimizer of (1.3) are

$$f_{min} = 0, \quad v^* = (1, 0, 0, -1).$$

Approximate ξ, ξ^2 by their sample averages $\bar{\xi}, s_2$, i.e., $s_2 = \frac{1}{N} \sum_{k=1}^{N} (\xi^{(k)})^2$. We make samples of different sizes and compute ϵ^* in (3.5) for each case. We focus on

TABLE 5. The values of ϵ^* for Example 4.5.

	I	II	III	IV
N	500	1000	5000	10000
$\bar{\xi}$	2.11	1.96	2.01	2.02
s_2	6.43	5.71	6.13	6.07
ϵ^*	0.807543	3.3618e-10	0.073413	0.146826

cases III and IV. The computational results are reported in Table 6. The pertur-

TABLE 6. Performance of PSAA for Example 4.5.

	III			IV		
ϵ	0	ϵ^*	0.1	0	ϵ^*	0.2
solvable?	no	yes	yes	no	yes	yes
time(sec.)	0.23	0.20	0.11	0.10	0.16	0.07
$\|\langle f_N, y^* \rangle - f_N(u)\|$	n.a.	5.30e+01	2.02e-01	n.a.	6.47e+01	2.28e-01
$\|\langle f_N, y \rangle - f_{min}\|$	n.a.	5.39e+01	3.90e-01	n.a.	6.49e+01	2.96e-01
$\|u - v^*\|$	n.a.	3.09e-02	1.28e-01	n.a.	5.86e-02	2.73e-01
u	n.a.	$\begin{bmatrix} 1.0216 \\ 0.0035 \\ 0.0035 \\ -1.0216 \end{bmatrix}$	$\begin{bmatrix} 0.9102 \\ 0.0071 \\ 0.0071 \\ -0.9102 \end{bmatrix}$	n.a.	$\begin{bmatrix} 0.9591 \\ 0.0059 \\ 0.0059 \\ -0.9591 \end{bmatrix}$	$\begin{bmatrix} 0.8070 \\ 0.0085 \\ 0.0085 \\ -0.8070 \end{bmatrix}$

bation term in the PSAA model (3.2) makes a big difference for computing reliable minimizers. The PSAA model returned minimizers that are close to the optimizer of (1.3), while the classical SAA model (i.e., $\epsilon = 0$) is unbounded from below and fails to return a minimizer.

Example 4.6. Consider the stochastic polynomial optimization

$$\min_{x \in \mathbb{R}^2} \quad f(x) := \mathbb{E}[G(x) + H(x, \xi)]$$

(4.6)

$$\text{s.t.} \quad x_1 - 1 \geq 0, \; x_2 \geq 0, \; 2 - \mathbf{1}^T x \geq 0$$

where

$$G(x) = x_1^4 + x_2^4 + x_1 x_2 - 2(x_1 + x_2) + 1,$$

$$H(x, \xi) = \xi_3 x_1 x_2 \big[\xi_1 x_1 + \xi_2 x_2 - (\xi_1 + \xi_2) x_1 x_2 \big],$$

$$\xi \sim \mathcal{N} \left(\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 3 \end{bmatrix} \right).$$

Note that $\mathbb{E}(\xi_1 \xi_3) = \mathbb{E}(\xi_2 \xi_3) = 1$ and the exact objective is

$$f = (x_1^2 - x_2^2)^2 + (x_1 x_2 - 1)(x_1 + x_2) + (x_1 - 1)(x_2 - 1).$$

Its optimal value and minimizer are

$$f_{min} = -0.2500, \quad v^* = (1.0000, \, 0.7071).$$

Denote $s_1 := \sum_{k=1}^N \xi_1^{(k)} \xi_3^{(k)}$, $s_2 := \sum_{k=1}^N \xi_2^{(k)} \xi_3^{(k)}$. We make samples of different sizes and compute ϵ^* in (3.5) for each case. The values of ϵ^* are given in Table 7. We focus on cases II and III, for which the numerical results are reported in Table 8.

TABLE 7. The values of ϵ^* for Example 4.6.

	I	II	III	IV
N	500	1000	5000	10000
s_1	0.98	1.08	1.01	0.97
s_2	1.06	0.96	1.02	0.96
ϵ^*	0.023094	0.023094	0.017321	1.1076e-09

While the optimization (1.4) is unbounded from below via the classical SAA model,

TABLE 8. Performance of PSAA for Example 4.6.

		II			III	
ϵ	0	ϵ^*	0.05	0	ϵ^*	0.05
solvable?	no	yes	yes	no	yes	yes
time(sec.)	0.06	0.21	0.06	0.06	0.13	0.05
$\lvert \langle f_N, y^* \rangle - f_N(u) \rvert$	n.a.	1.13e+02	1.36e-02	n.a.	5.65e+00	6.25e-03
$\lvert \langle f_N, y \rangle - f_{min} \rvert$	n.a.	1.13e+02	4.21e-03	n.a.	5.65e+00	2.80e-03
$\lVert u - v^* \rVert$	n.a.	1.20e-03	1.85e-02	n.a.	6.08e-03	2.58e-02
u	n.a.	$\begin{bmatrix} 1.0000 \\ 0.7059 \end{bmatrix}$	$\begin{bmatrix} 1.0000 \\ 0.6886 \end{bmatrix}$	n.a.	$\begin{bmatrix} 1.0000 \\ 0.7010 \end{bmatrix}$	$\begin{bmatrix} 1.0000 \\ 0.6813 \end{bmatrix}$

our PSAA model (3.2) provides good lower bounds and returns reliable minimizers for all cases.

Example 4.7. Consider the stochastic polynomial optimization

$$
\begin{aligned}
\min_{x \in \mathbb{R}^3} \quad & f(x) = \mathbb{E}[F(x,\xi)] \\
\text{s.t.} \quad & x_1 - 1 \geq 0, \ x_2 - 1 \geq 0, \ x_3 - 1 \geq 0, \\
& 8 - x_1 x_2 x_3 \geq 0
\end{aligned}
$$

(4.7)

where $\xi \sim \mathcal{U}(0,2)$ and

$$
F(x,\xi) = (-\xi^3 + 3\xi_2^2 - \xi)x_1 x_2 x_3(x_1 + x_2 + x_3) + (\xi^3 - \xi)(x_1 x_2 + x_2 x_3 + x_1 x_3).
$$

One can see that $\mathbb{E}(\xi) = 1, \mathbb{E}(\xi^2) = 4/3, \mathbb{E}(\xi^3) = 2$, and the exact objective

$$
f = x_1 x_2 x_3(x_1 + x_2 + x_3) + x_1 x_2 + x_2 x_3 + x_1 x_3.
$$

Its optimal value and minimizer are

$$
f_{min} = 6, \quad v^* = (1,\ 1,\ 1).
$$

As in Examples 4.5-4.6, we approximate ξ, ξ^2, ξ^3 by sample averages where $s_2 = \sum_{k=1}^{N} \left(\xi^{(k)}\right)^2$, $s_3 = \sum_{k=1}^{N} \left(\xi^{(k)}\right)^3$. Several samples of different sizes are made. The values of ϵ^* in (3.5) are given in Table 9. We apply the classical SAA model (i.e.,

TABLE 9. The values of ϵ^* for Example 4.7.

	I	II	III
N	500	1000	5000
$\bar{\xi}$	0.99	1.03	1.00
s_2	1.32	1.38	1.33
s_3	1.97	2.09	1.99
ϵ^*	0.508637	0.518810	0.508637

$\epsilon = 0$) and the PSAA model (i.e., $\epsilon = \epsilon^*$) to cases I, II and III. The computational

TABLE 10. Performance of PSAA for Example 4.7.

	I		II		III	
ϵ	0	ϵ^*	0	ϵ^*	0	ϵ^*
solvable?	yes	yes	yes	yes	yes	yes
time(sec.)	0.07	0.06	0.07	0.06	0.08	0.06
$\lvert \langle f_N, y^* \rangle - f_N(u) \rvert$	7.08e+01	1.07e-06	7.08e+01	8.44e-07	7.08e+01	1.07e-06
$\lvert \langle f_N, y \rangle - f_{min} \rvert$	2.07e+01	2.00e-02	2.07e+01	1.00e-02	2.07e+01	1.00e-02
$\lVert u - v^* \rVert$	1.67e+00	2.97e-08	1.67e+00	2.97e-08	1.67e+00	2.97e-08
u	$\begin{bmatrix} 1.9637 \\ 1.9637 \\ 1.9630 \end{bmatrix}$	$\begin{bmatrix} 1.0000 \\ 1.0000 \\ 1.0000 \end{bmatrix}$	$\begin{bmatrix} 1.9631 \\ 1.9631 \\ 1.9627 \end{bmatrix}$	$\begin{bmatrix} 1.0000 \\ 1.0000 \\ 1.0000 \end{bmatrix}$	$\begin{bmatrix} 1.9631 \\ 1.9631 \\ 1.9627 \end{bmatrix}$	$\begin{bmatrix} 1.0000 \\ 1.0000 \\ 1.0000 \end{bmatrix}$

results are reported in the Table 10. The PSAA model (3.2) performs much better than the classical SAA model (1.4), as (3.2) gives more reliable minimizers for all cases.

5. Conclusion

This paper proposes a sample average optimization model with a perturbation term for solving stochastic polynomial optimization. The perturbation optimization model performs better than the classical one without perturbations. The Lasserre type moment relaxations are used to solve the perturbation optimization. In particular, we show that the moment relaxation is tight if and only if the moment matrix of the minimizer is rank one. Numerical experiments demonstrated advantages of our perturbation optimization model.

References

[1] F. Bastin, C. Cirillo and P. Toint Convergence theory for nonconvex stochastic programming with an application to mixed logit. *Math. Program.*, 108(23), pp. 207–234, 2006.

[2] A. Ben-Tal and A. Nemirovski, *Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications*, MPS-SIAM Series on Optimization, SIAM, Philadelphia, 2001.

[3] G. Blekherman, P. Parrilo and R. Thomas (eds.), *Semidefinite optimization and convex algebraic geometry*, MOS-SIAM Series on Optimization, SIAM, 2013.

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

[5] M. Branda. Sample approximation technique for mixed-integer stochastic programming problems with several chance constraints. *Operations Research Letters*, 40(3), pp. 207–211, 2012.

[6] K. Chung, *A course in probability theory*, Academic press, 2001.

[7] R. Curto and L. Fialkow. Truncated K-moment problems in several variables. *Journal of Operator Theory*, 54(2005), pp. 189-226.

[8] M. Fu (eds.), *Handbook of simulation optimization*, volume 216, Springer, New York, 2015.

[9] S. Ghadimi and G. Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming, *SIAM J. Optim.*, 23(4), pp. 2341–2368, 2013.

[10] S. Ghadimi, G. Lan and H. Zhang. Mini-batch stochastic approximation methods for nonconvex stochastic composite optimization, *Math. Program.*, 155(1-2), pp. 267–305, 2016.

[11] J. Helton and J. Nie. Semidefinite representation of convex sets. *Math. Program., Series A*, Vol. 122, No.1, pp. 21-64, 2010.

[12] J. Helton and J. Nie, A semidefinite approach for truncated K-moment problem, *Found. Comput. Math.*, 12(6), pp. 851–881, 2012.

[13] D. Henrion and J. Lasserre. Detecting global optimality and extracting solutions in GloptiPoly. *Positive polynomials in control*, 293–310, Lecture Notes in Control and Inform. Sci., 312, Springer, Berlin, 2005.

[14] D. Henrion, J. Lasserre, and J. Löfberg. `Gloptipoly 3:` moments, optimization and semidefinite programming. *Optimization Methods & Software*, 24(4-5), pp. 761–779, 2009.

[15] K. Judd. The law of large numbers with a continuum of iid random variables. *Journal of Economic theory*, 35(1), pp. 19–25, 1985.

[16] E. Klerk and M. Laurent. On the Lasserre hierarchy of semidefinite programming relaxations of convex polynomial optimization problems, *SIAM J. Optim.*, 21, pp. 824-832, 2011.

[17] A. Kleywegt, A. Shapiro, and T. Mello. The sample average approximation method for stochastic discrete optimization. *SIAM J. Optim.*, 12(2), pp. 479–502, 2002.

[18] J. Lasserre, Global optimization with polynomials and the problem of moments, *SIAM J. Optim.*, 11, pp. 796–817, 2001.

[19] J. Lasserre. Convexity in semi-algebraic geometry and polynomial optimization. *SIAM J. Optim.*, 19 (2009), pp. 1995-2014.

[20] J. Lasserre. *Introduction to polynomial and semi-algebraic optimization*, Cambridge University Press, Cambridge, 2015.

[21] J. Lasserre, M. Laurent, and P. Rostalski. Semidefinite characterization and computation of zero-dimensional real radical ideals. *Found. Comput. Math.* 8, pp. 607–647, 2008.

[22] M. Laurent. Semidefinite representations for finite varieties. *Math. Program.*, Vol. 109, pp. 1–26, 2007.

[23] M. Laurent, Sums of squares, moment matrices and optimization over polynomials, *Emerging Applications of Algebraic Geometry of IMA Volumes in Mathematics and its Applications*, 149, pp. 157–270, Springer, 2009.

[24] J. Linderoth, A. Shapiro, and S. Wright. The empirical behavior of sampling methods for stochastic programming. *Annals of Operations Research*, 142(1):215–241, 2006.

[25] J. Luedtke and S. Ahmed. A sample approximation approach for optimization with probabilistic constraints. *SIAM J. Optim.*, 19(2), pp. 674–699, 2008.

[26] J. Nie. Certifying convergence of Lasserre's hierarchy via flat truncation. *Math. Program.*, Ser. A, 142 (2013), no. 1-2, pp. 485–510.

[27] J. Nie, Polynomial optimization with real varieties, *SIAM J. Optim.*, 23(3), pp. 1634–1646, 2013.

[28] J. Nie. Optimality conditions and finite convergence of Lasserre's hierarchy. *Math. Program.*, Ser. A, 146 (2014), no. 1-2, pp. 97–121.

[29] J. Nie, The $\mathcal{A}$-truncated K-moment problem, *Found. Comput. Math.*, 14(6), pp. 1243–1276, 2014.

[30] J. Nie. Linear optimization with cones of moments and nonnegative polynomials. *Math. Program.*, Ser. B, 153 (2015), no. 1, pp. 247–274.

[31] P. Parrilo. Semidefinite Programming relaxations for semialgebraic problems. *Math. Program., Ser. B* **96** (2003), No. 2, pp. 293–320.

[32] P. Parrilo and B. Sturmfels. Minimizing polynomial functions. *Algorithmic and quantitative real algebraic geometry (Piscataway, NJ, 2001)*, pp. 83–99, DIMACS Ser. Discrete Math. Theoret. Comput. Sci., 60, Amer. Math. Soc., Providence, RI, 2003.

[33] E. Plambeck, B. Fu, S. Robinson, and R. Suri. Sample-path optimization of convex stochastic performance functions. *Math. Program.*, 75(2), pp. 137–176, 1996.

[34] M. Putinar. Positive polynomials on compact semi-algebraic sets, *Ind. Univ. Math. J.* 42 (1993), pp. 969–984.

[35] S. Robinson. Analysis of sample-path optimization. *Mathematics of Operations Research*, 21(3), pp. 513–528, 1996.

[36] T. Santoso, S. Ahmed, M. Goetschalckx, and A. Shapiro. A stochastic programming approach for supply chain network design under uncertainty. *European Journal of Operational Research*, 167(1), pp. 96–115, 2005.

[37] C. Scheiderer. Positivity and sums of squares: A guide to recent results. *Emerging Applications of Algebraic Geometry (M. Putinar, S. Sullivant, eds.)*, IMA Volumes Math. Appl. 149, Springer, 2009, pp. 271–324.

[38] S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan. Stochastic convex optimization. In *COLT*, 2009.

[39] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on stochastic programming: modeling and theory.* SIAM, 2009.

[40] J. Sturm, SeDuMi 1.02: a MATLAB toolbox for optimization over symmetric cones, *Optimization Methods and Software*, 11&12, pp. 625–653, 1999. `http://sedumi.mcmaster.ca/`

[41] B. Verweij, S. Ahmed, A. Kleywegt, G. Nemhauser, and A. Shapiro. The sample average approximation method applied to stochastic routing problems: a computational study. *Computational Optimization and Applications*, 24(2-3):289–333, 2003.

Jiawang Nie, Suhan Zhong, Department of Mathematics, University of California San Diego, 9500 Gilman Drive, La Jolla, CA, USA, 92093.
   E-mail address: `njw@math.ucsd.edu,suzhong@ucsd.edu`

Liu Yang, school of Mathematics and Computational Sciences, Xiangtan University, Xiangtan, Hunan, China, 411105.
   E-mail address: `yangl410@xtu.edu.cn`