Annals of Operations Research
https://doi.org/10.1007/s10479-019-03143-z

S.I.: STOCHASTIC MODELING AND OPTIMIZATION, IN MEMORY OF ANDRÁS PRÉKOPA

A randomized method for handling a difficult function in a convex optimization problem, motivated by probabilistic programming

Csaba I. Fábián[1] · Edit Csizmás[1] · Rajmund Drenyovszki[1] · Tibor Vajnai[1] · Lóránt Kovács[1] · Tamás Szántai[2]

Abstract

We propose a randomized gradient method for handling a convex function whose gradient computation is demanding. The method bears a resemblance to the stochastic approximation family. But in contrast to stochastic approximation, the present method builds a model problem. The approach is adapted to probability maximization and probabilistic constrained problems. We discuss simulation procedures for gradient estimation.

Keywords Convex optimization · Stochastic optimization · Probabilistic problems

1 Introduction

We deal with approximate methods for the solution of smooth convex programming problems. First, we consider minimization over a polyhedron:

$$\min \ \phi(T\boldsymbol{x}) \quad \text{subject to} \quad A\boldsymbol{x} \leq \boldsymbol{b}, \tag{1}$$

where $\phi : \mathbb{R}^n \to \mathbb{R}$ is a convex function whose gradient computation is demanding. The vectors are $\boldsymbol{x} \in \mathbb{R}^m$, $\boldsymbol{b} \in \mathbb{R}^r$, and the matrices T and A are of sizes $n \times m$ and $r \times m$, respectively. For the sake of simplicity we assume that the feasible domain is not empty and is bounded. We then consider the minimization of a linear cost function subject to a difficult convex constraint:

This paper is dedicated to András Prékopa.

This research is supported by EFOP-3.6.1-16-2016-00006 Project titled 'The development and enhancement of the research potential at John von Neumann University'. The Project is supported by the Hungarian Government and co-financed by the European Social Fund.

✉ Csaba I. Fábián
fabian.csaba@gamf.uni-neumann.hu

Extended author information available on the last page of the article

Published online: 21 January 2019

$$\min\ \boldsymbol{c}^T\boldsymbol{x} \quad \text{subject to} \quad \breve{A}\boldsymbol{x} \le \breve{\boldsymbol{b}}, \quad \phi(T\boldsymbol{x}) \le \pi, \tag{2}$$

where the vectors $\boldsymbol{c}$, $\breve{\boldsymbol{b}}$ and the matrix $\breve{A}$ have compatible sizes, and π is a given number. The approach we discuss is easily extended to convex functions from the linear ones.

A motivation for the above forms are the classic probability maximization and probabilistic constrained problems, where $\phi(z) = -\log F(z)$ with a logconcave distribution function $F(z)$. We briefly overview a couple of closely related probabilistic programming approaches. For a broader survey, see Fábián et al. (2018). Given a distribution and a number p $(0 < p < 1)$, a probabilistic constraint confines search to the level set $\mathcal{L}(F, p) = \{\, z \mid F(z) \ge p \,\}$ of the distribution function $F(z)$. Prékopa (1990) initiated a novel solution approach by introducing the concept of p-efficient points. The point z is p-efficient if $F(z) \ge p$ and there exists no z' such that $z' \le z,\ z' \ne z,\ F(z') \ge p$. Prékopa et al. (1998) considered problems with random parameters having a discrete finite distribution. They began with enumerating p-efficient points and based on them, built a convex relaxation of the problem.

Dentcheva et al. (2000) formulated the probabilistic constraint in a split form: $T\boldsymbol{x} = z$ with $z \in \mathcal{L}(F, p)$; and constructed a Lagrangian dual by relaxing the constraint $T\boldsymbol{x} = z$. The resulting dual functional is the sum of the respective optimal objective values of two simpler problems. The first auxiliary problem is a linear programming problem, and the second one is the minimization of a linear function over the level set $\mathcal{L}(F, p)$. Based on this decomposition, the authors developed a method, called cone generation, that finds new p-efficient points in the course of the optimization process.

As minimization over the level set $\mathcal{L}(F, p)$ entails a substantial computational effort, the master part of the decomposition framework should succeed with as few p-efficient points as possible. Efficient solution methods were developed by Dentcheva et al. (2004) and Dentcheva and Martinez (2013); the latter applies regularization to the master problem. Approximate minimization over the level set $\mathcal{L}(F, p)$ is another enhancement. Dentcheva et al. (2004) constructed approximate p-efficient points through approximating the original distribution by a discrete one. More recently, van Ackooij et al. (2017) employed a special bundle-type method for the solution of the master problem, based on the on-demand accuracy approach of de Oliveira and Sagastizábal (2014). This means working with inexact data and regulating accuracy in the course of the optimization. Approximate p-efficient points with on-demand accuracy were generated employing the integer programming approach of Luedtke et al. (2010).

Our former paper Fábián et al. (2018) focussed on probability maximization, and proposed a polyhedral approximation of the epigraph of the probabilistic function. This approach is analogous to the use of p-efficient points (has actually been motivated by that concept). The dual function is constructed and decomposed in the manner of Dentcheva et al. (2000), but the nonlinear subproblem is easier. In Dentcheva et al. (2000), finding a new p-efficient point amounts to minimization over the level set $\mathcal{L}(F, p)$. In contrast, a new approximation point in Fábián et al. (2018) is found by unconstrained minimization, with considerably less computational effort. Moreover, a practical approximation scheme was developed in the latter paper: instead of exactly solving an unconstrained subproblem occurring during the process, just a single line search is sufficient. The approach is easy to implement and endures noise in gradient computation.

In the present paper, we extend the inner approximation approach of Fábián et al. (2018) to a randomized method handling gradient estimates. The motivation is our experience reported in that former paper: when solving probability maximization problems, most computational efforts were spent on computing gradients. (Computing a single component of the gradient vector required an effort comparable to that of computing a distribution function value). We conclude that easily computable estimates for the gradients are well worth using, even if the iteration count increases due to estimation errors.

The paper is organized as follows. In Sect. 2 we work in an idealized setting, under the following assumptions:

Assumption 1 The function $\phi(z)$ is twice continuously differentiable, and real numbers α, ω $(0 < \alpha \leq \omega)$ are known such that

$$\alpha I \preceq \nabla^2 \phi(z) \preceq \omega I \quad (z \in \mathrm{I\!R}^n).$$

Here $\nabla^2 \phi(z)$ is the Hessian matrix, I is the identity matrix, and the relation $U \preceq V$ between matrices means that $V - U$ is positive semidefinite.

Assumption 2 Given $z \in \mathrm{I\!R}^n$, the function value $\phi(z)$ and the gradient vector $\nabla \phi(z)$ can be computed exactly.

We present a brief overview of the models and of the column generation approach proposed in Fábián et al. (2018) to the unconstrained problem (1). The epigraph of the convex function $\phi(z)$ is approximated by a convex combination of finitely many points (obtained by evaluating the function value in the known iterates.) New points (columns in a model problem) are generated by unconstrained minimization of a probabilistic function. The column generation problem is solved with a gradient descent method. Due to Assumption 1, an approximate solution is sufficient, taking a limited number of descent steps.

In Sect. 3 we extend the method to gradient estimates, replacing Assumption 2 with

Assumption 3 Given $z, \boldsymbol{u} \in \mathrm{I\!R}^n$, the function value $\phi(z)$ can be computed exactly, and the norm $\|\nabla \phi(z) - \boldsymbol{u}\|$ can be estimated with a pre-defined relative accuracy. Moreover, realizations of an unbiased stochastic estimate $\boldsymbol{G}$ of the gradient vector $\nabla \phi(z)$ can be constructed such that $\mathrm{E}(\|\boldsymbol{G} - \nabla \phi(z)\|^2)$ remains below a pre-defined tolerance. (Higher accuracy in case of norm estimation, and tighter tolerance on variance entail larger computational effort.)

We develop a randomized version of the column generation method, and present reliability considerations based on Assumption 1.

In Sect. 4 we deal with the convex constrained problem (2), still in the idealized setting of Assumption 1. We consider a parametric version of an unconstrained problem of the form (1). We present an approximation scheme for the constrained problem that requires the approximate solution of a short sequence of unconstrained problems. Initial problems in this sequence are solved with a large stopping tolerance, and the accuracy is gradually increased. This approximation scheme is first developed in a deterministic form (based on the deterministic method of Sect. 2), and then extended to admit the randomized method of Sect. 3. Reliability considerations are presented for the randomized scheme.

The approach is adapted to probabilistic programming problems in Sect. 5. Here we consider $\phi(z) = -\log F(z)$ with a nondegenerate n-dimensional standard normal distribution function $F(z)$. Assumption 1 obviously does not hold for every $z \in \mathrm{I\!R}^n$ with a probabilistic $\phi(z)$. However, as illustrated in Fábián et al. (2018), Assumption 1 holds for the points of a bounded ball around the origin. (The ratio $\frac{\alpha}{\omega}$ decreases as the radius of the ball increases.) Owing to the specialities of the probabilistic function, the column generation process can be guaranteed to remain in a ball of sufficiently large radius. Such a procedure was sketched in Fábián et al. (2018). That construction provides a theoretical justification for limiting our investigations to a bounded ball, but it does not yield usable estimates for the values α and ω. While efficiency considerations of the previous sections are inherited to probabilistic problems, reliability considerations cannot be based on Assumption 1. The quality of a model is measured by different means, based on special features of the probabilistic function.

Section 6 contains an overview of algorithms for the estimation of multivariate normal probability distribution function values and gradients. We discuss the numerical integration algorithm of Genz (1992), and the variance reduction Monte Carlo simulation algorithms of Deák (1980, 1986), Szántai (1976, 1985, 1988) and Ambartzumian et al. (1998), mentioning related works. These variance reduction Monte Carlo simulation algorithms have originally been developed to be used in primal-type methods for probabilistic constrained problems. An abundant stream of research in this direction has been initiated by the models, methods and applications pioneered by Prékopa and his school. Based on these algorithms, a gradient estimate satisfying Assumption 3 can be constructed by a two-stage sampling procedure, as mentioned in Sect. 6.5.

Section 7 describes a computational experiment. The aim is to demonstrate the workability of the randomized column generation scheme of Sect. 3, in case of probabilistic problems.

2 Column generation in an idealized setting

In this section we work in the idealized setting of Assumptions 1 and 2. We formulate the dual problem and construct polyhedral models of the primal and dual problems. We follow the construction in Fábián et al. (2018), though monotonicity of the probabilistic objective was exploited there, and variable splitting was based on $z \le Tx$. In the present idealized setting, we apply the traditional form of variable splitting: problem (1) is written as

$$\min\ \phi(z) \quad \text{subject to} \quad A\boldsymbol{x} - \boldsymbol{b} \le \boldsymbol{0}, \quad z - T\boldsymbol{x} = \boldsymbol{0}. \tag{3}$$

This problem has an optimal solution because the feasible domain of (1) is nonempty and bounded, by assumption. Introducing the multiplier vector $-\boldsymbol{y} \in \mathrm{I\!R}^r$, $-\boldsymbol{y} \ge \boldsymbol{0}$ to the constraint $A\boldsymbol{x} - \boldsymbol{b} \le \boldsymbol{0}$, and $-\boldsymbol{u} \in \mathrm{I\!R}^n$ to the constraint $z - T\boldsymbol{x} = \boldsymbol{0}$, the Lagrangian dual of (3) can be written as

$$\max\ \left\{\boldsymbol{y}^T\boldsymbol{b} - \phi^\star(\boldsymbol{u})\right\} \quad \text{subject to} \quad (\boldsymbol{y}, \boldsymbol{u}) \in \mathcal{D}, \tag{4}$$

where

$$\mathcal{D} := \left\{(\boldsymbol{y}, \boldsymbol{u}) \in \mathrm{I\!R}^{r+n} \mid \boldsymbol{y} \le \boldsymbol{0}, \ \ T^T\boldsymbol{u} = A^T\boldsymbol{y}\right\}. \tag{5}$$

According to the theory of convex duality, this problem has an optimal solution. For a recent treatise on Lagrangian duality, see, e.g., Chapter 4 of Ruszczyński (2006).

2.1 Polyhedral models

Suppose we have evaluated the function $\phi(z)$ at points z_i $(i = 0, 1, \ldots, k)$; we introduce the notation $\phi_i = \phi(z_i)$ for respective objective values. An inner approximation of $\phi(\cdot)$ is

$$\phi_k(z) = \min \sum_{i=0}^{k} \lambda_i \phi_i \quad \text{such that} \quad \lambda_i \ge 0 \ (i = 0, \ldots, k), \quad \sum_{i=0}^{k} \lambda_i = 1, \quad \sum_{i=0}^{k} \lambda_i z_i = z. \tag{6}$$

If $z \notin \mathrm{Conv}(z_0, \ldots, z_k)$, then let $\phi_k(z) := +\infty$. A polyhedral model of problem (3) is

$$\min\ \phi_k(z) \quad \text{subject to} \quad A\boldsymbol{x} - \boldsymbol{b} \le \boldsymbol{0}, \quad z - T\boldsymbol{x} = \boldsymbol{0}. \tag{7}$$

We assume that (7) is feasible, i.e., its optimum is finite. This can be ensured by proper selection of the initial $z_0, \dots, z_k$ points. The convex conjugate of $\phi_k(z)$ is

$$\phi_k^\star(\boldsymbol{u}) = \max_{0 \le i \le k} \left\{ \boldsymbol{u}^T z_i - \phi_i \right\}. \tag{8}$$

As $\phi_k^\star(\cdot)$ is a cutting-plane model of $\phi^\star(\cdot)$, the following problem is a polyhedral model of problem (4):

$$\max \left\{ \boldsymbol{y}^T \boldsymbol{b} - \phi_k^\star(\boldsymbol{u}) \right\} \quad \text{subject to} \quad (\boldsymbol{y}, \boldsymbol{u}) \in \mathcal{D}. \tag{9}$$

2.2 Linear programming formulations

The primal model problem (6)–(7) will be formulated as

$$\begin{array}{ll} \min & \sum\limits_{i=0}^{k} \phi_i \lambda_i \\ \text{such that} & \lambda_i \ge 0 \quad (i = 0, \dots, k), \\ & \sum\limits_{i=0}^{k} \lambda_i = 1, \\ & \sum\limits_{i=0}^{k} \lambda_i z_i - T\boldsymbol{x} = \boldsymbol{0}, \\ & A\boldsymbol{x} \le \boldsymbol{b}. \end{array} \tag{10}$$

The dual model problem (8)–(9), formulated as a linear programming problem, is just the LP dual of (10):

$$\begin{array}{l} \max \vartheta + \boldsymbol{b}^T \boldsymbol{y} \\ \text{such that } \boldsymbol{y} \le \boldsymbol{0}, \\ \vartheta + z_i^T \boldsymbol{u} \le \phi_i \quad (i = 0, \dots, k), \\ -T^T \boldsymbol{u} + A^T \boldsymbol{y} = \boldsymbol{0}. \end{array} \tag{11}$$

Let $(\overline{\lambda}_0, \dots, \overline{\lambda}_k, \overline{\boldsymbol{x}})$ and $(\overline{\vartheta}, \overline{\boldsymbol{u}}, \overline{\boldsymbol{y}})$ denote respective optimal solutions of the problems (10) and (11)—both existing due to our assumption concerning the feasibility of (7) and hence (10). Let moreover

$$\overline{z} = \sum_{i=0}^{k} \overline{\lambda}_i z_i. \tag{12}$$

Observation 4 *We have*

(a) $\phi_k(\overline{z}) = \sum_{i=0}^{k} \phi_i \overline{\lambda}_i = \overline{\vartheta} + \overline{\boldsymbol{u}}^T \overline{z}$,
(b) $\overline{\vartheta} = -\phi_k^\star(\overline{\boldsymbol{u}})$,
(c) $\phi_k(\overline{z}) + \phi_k^\star(\overline{\boldsymbol{u}}) = \overline{\boldsymbol{u}}^T \overline{z}$ *and hence* $\overline{\boldsymbol{u}} \in \partial \phi_k(\overline{z})$.

Sketch of proof

(a) The first equality follows from the equivalence of (10) on the one hand, and (6)–(7) on the other hand. The second equality is a straight consequence of complementarity.

(b) follows from the equivalence between (11) on the one hand and (8)–(9) on the other hand.

(c) The equality is a consequence of *(a)* and *(b)*. This is Fenchel's equality between $\overline{\boldsymbol{u}}$ and $\overline{z}$, with respect to the model function $\phi_k(\cdot)$. On $\overline{\boldsymbol{u}}$ being a subgradient, see, e.g., Section 23 in Rockafellar (1970).

2.3 A column generation procedure

We give a brief overview of the approximation scheme of Fábián et al. (2018). An optimal dual solution (i.e., shadow price vector) of the current model problem is $(\,\overline{\vartheta},\ \overline{\boldsymbol{u}},\ \overline{\boldsymbol{y}}\,)$. Given a vector $z \in \mathrm{I\!R}^n$, we can add a new column in (10), corresponding to $z_{k+1} = z$. This is an improving column if its reduced cost

$$\overline{\rho}(z) \ := \ \overline{\vartheta} \ + \ \overline{\boldsymbol{u}}^T z - \phi(z) \tag{13}$$

is positive. It is easily seen that the reduced cost of $\overline{z}$ is non-negative. Indeed,

$$\overline{\rho}(\overline{z}) \ \geq \ \overline{\vartheta} \ + \ \overline{\boldsymbol{u}}^T \overline{z} - \phi_k(\overline{z}) \ = 0 \tag{14}$$

follows from $\phi_k(\cdot) \geq \phi(\cdot)$ and Observation 4(a).

In the context of the simplex method, the Markowitz column-selection rule is widely used. The Markowitz rule selects the vector with the largest reduced cost. Coming back to the present problem (10), let

$$\overline{\mathcal{R}} := \max_z \overline{\rho}(z). \tag{15}$$

The column with the largest reduced cost can, in theory, be found by a steepest descent method applied to the function $-\overline{\rho}(z)$. In a practical approach, only a limited number of line search steps are performed, starting from $\overline{z}$. The efficiency of this practical approach can be estimated on the basis of the following well-known theorem:

Theorem 5 *Let Assumption 1 hold for the function $f : \mathrm{I\!R}^n \to \mathrm{I\!R}$. We minimize $f(z)$ over $\mathrm{I\!R}^n$ using a steepest descent method, starting from a point z^0. Let $z^1, \ldots, z^j, \ldots$ denote the iterates obtained by applying exact line search at each step. Then we have*

$$f\left(z^j\right) - \mathcal{F} \ \leq \ \left(1 - \frac{\alpha}{\omega}\right)^j \left[\ f\left(z^0\right) - \mathcal{F}\ \right], \tag{16}$$

where $\mathcal{F} = \min_z f(z)$.

This theorem can be found e.g., in Chapter 8.6 of Luenberger and Ye (2008). Ruszczyński (2006) in Chapter 5.3.5, Theorem 5.7 presents a slightly different form. The following corollary was obtained in Fábián et al. (2018):

Corollary 6 *Let β $(0 < \beta \ll 1)$ be given. In $O\,(-\log\beta)$ steps with the steepest descent method, we find a vector $\widehat{z}$ such that*

$$\overline{\rho}\,(\widehat{z}) \ \geq \ (1-\beta)\,\overline{\mathcal{R}}. \tag{17}$$

This can be shown by substituting $f(z) = -\overline{\rho}(z),\ \ z^0 = \overline{z}$ in (16), and applying (14). The objective function $-\overline{\rho}(z)$ inherits Assumption 1 from $\phi(z)$. Performing j steps with j such that $(1-\frac{\alpha}{\omega})^j \leq \beta$ yields an appropriate $\widehat{z} = z^j$.

In view of the Markowitz rule mentioned above, the vector $\widehat{z}$ in Corollary 6 is a fairly good improving vector in the column generation scheme.

To check near-optimality of the current solution, we use the usual LP stopping rule: the reduced cost of any candidate vector should be below a fixed optimality tolerance. Of course we do not know $\overline{\mathcal{R}}$, but let

$$\overline{\mathcal{B}} := \frac{1}{1-\beta}\overline{\rho}\left(\widehat{z}\right), \tag{18}$$

with the β and $\widehat{z}$ of Corollary 6. We stop the column generation procedure when $\overline{\mathcal{B}}$ falls below the optimality tolerance. When applying this bound, we work with a fixed β throughout the process, e.g., let $\beta = 0.5$. For the present special linear programming problem (10), this is not just a heuristic rule:

Observation 7 *$\overline{\mathcal{R}}$ (and hence $\overline{\mathcal{B}}$) is an upper bound on the gap between the respective optima of the model problem* (10) *and the original convex problem* (3).

Proof We have

$$\overline{\mathcal{R}} \;=\; \max_{z}\, \overline{\rho}(z) \;=\; \phi^{\star}(\overline{\boldsymbol{u}}) - \phi_k^{\star}(\overline{\boldsymbol{u}}). \tag{19}$$

(The second equality follows from the definition of the conjugate function).

Since $(\overline{\boldsymbol{u}}, \overline{\boldsymbol{y}})$ is a feasible solution of the dual problem (4), it follows that (19) is an upper bound on the gap between the respective optima of the dual model problem (9) and the dual problem (4). The observation follows from convex duality. □

Remark 8 Prescribing a loose optimality tolerance on $\overline{\mathcal{B}}$ results in an early termination of the column generation process. Common experience with LP problems is that computational effort is substantially reduced by loosening the stopping tolerance.

Remark 9 Looking at the column-generation approach from a dual viewpoint we can see a cutting-plane method. This relationship between the primal and dual approaches is well known, see, e.g., Frangioni (2002, 2018). Details for the present case were worked out in the research report Fábián and Szántai (2017), a former version of the present paper.

The dual viewpoint admits a visual justification of the convergence of the sequence of the optimal dual vectors $\overline{\boldsymbol{u}}$. (Moreover, the cutting-plane method can be regularized, but we do not consider regularization in this paper.)

3 Working with gradient estimates

First we extend Theorem 5. Let $f : \mathbb{R}^n \to \mathbb{R}$ be such that Assumptions 1 and 3 hold. We wish to minimize $f(z)$ over $\mathbb{R}^n$ using a stochastic descent method. Let $z^\circ \in \mathbb{R}^n$ denote an iterate, and $\boldsymbol{g}^\circ = \nabla f(z^\circ)$ the corresponding gradient.

Let $\sigma^2 > 0$ be given. According to Assumption 3, realizations of a random vector $\boldsymbol{G}^\circ$ can be constructed, satisfying

$$\mathrm{E}\left(\boldsymbol{G}^\circ\right) = \boldsymbol{g}^\circ \quad \text{and} \quad \mathrm{E}\left(\left\|\boldsymbol{G}^\circ - \boldsymbol{g}^\circ\right\|^2\right) \;\le\; \sigma^2 \left\|\boldsymbol{g}^\circ\right\|^2. \tag{20}$$

From (20) follows

$$\mathrm{E}\left(\left\|\boldsymbol{G}^\circ\right\|^2\right) \;=\; \mathrm{E}\left(\left\|\boldsymbol{G}^\circ - \boldsymbol{g}^\circ\right\|^2\right) + \left\|\boldsymbol{g}^\circ\right\|^2 \;\le\; (\sigma^2+1)\,\left\|\boldsymbol{g}^\circ\right\|^2. \tag{21}$$

We consider the following randomized form of Theorem 5:

Theorem 10 *Under the above assumptions, we perform a steepest descent method using gradient estimates: at the current iterate z°, a gradient estimate $\boldsymbol{G}^\circ$ is generated and a line search is performed in that direction. We assume that gradient estimates at the respective iterates are generated independently, and (20)–(21) hold for each of them.*

Having started from the point z^0, and having performed j line searches, let $z^1, \ldots, z^j$ denote the respective iterates. Then we have

$$E\left[f\left(z^j\right)\right] - \mathcal{F} \;\leq\; \left(1 - \frac{\alpha}{\omega(\sigma^2+1)}\right)^j \left(f\left(z^0\right) \;- \mathcal{F}\right), \tag{22}$$

where $\mathcal{F} = \min_z f(z)$.

Proof Let $\boldsymbol{G}^0, \ldots, \boldsymbol{G}^{j-1}$ denote the respective gradient estimates for the iterates $z^0, \ldots, z^{j-1}$.

To begin with, we focus on the first line search whose starting point is $z^\circ = z^0$. Here z° is a given (not random) vector. We adapt the proof of Theorem 5, presented in Chapter 8.6 of Luenberger and Ye (2008), to employ the gradient estimate $\boldsymbol{G}^\circ$ instead of the gradient $\boldsymbol{g}^\circ$. From $\nabla^2 f(z) \preceq \omega I$, it follows that

$$f\left(z^\circ - t\boldsymbol{G}^\circ\right) \leq f\left(z^\circ\right) - t\, \boldsymbol{g}^{\circ\,T} \boldsymbol{G}^\circ + \frac{\omega}{2} t^2\, \boldsymbol{G}^{\circ\,T} \boldsymbol{G}^\circ$$

holds for any $t \in \mathbb{R}$ (a consequence of Taylor's theorem). Considering expectations on both sides, we get

$$\begin{aligned} \mathrm{E}\left[f\left(z^\circ - t\boldsymbol{G}^\circ\right)\right] &\leq\; f\left(z^\circ\right) - t\, \left\|\boldsymbol{g}^\circ\right\|^2 + \tfrac{\omega}{2} t^2\, \mathrm{E}\left(\left\|\boldsymbol{G}^\circ\right\|^2\right) \\ &\leq f\left(z^\circ\right) - t\, \left\|\boldsymbol{g}^\circ\right\|^2 + \tfrac{\omega}{2} t^2\, \left(\sigma^2+1\right)\, \left\|\boldsymbol{g}^\circ\right\|^2 \end{aligned}$$

according to (21). We consider the respective minima in t separately of the two sides. The right-hand side is a quadratic expression, yielding minimum at $t = \frac{1}{\omega(\sigma^2+1)}$. Inequality is inherited to minima, hence

$$\min_t \mathrm{E}\left[f\left(z^\circ - t\boldsymbol{G}^\circ\right)\right] \;\leq\; f\left(z^\circ\right) \;-\; \frac{1}{2\omega(\sigma^2+1)}\; \left\|\boldsymbol{g}^\circ\right\|^2. \tag{23}$$

For the left-hand side, we obviously have

$$\mathrm{E}\left[\,\min_t\, f\left(z^\circ - t\boldsymbol{G}^\circ\right)\right] \;\leq\; \min_t \mathrm{E}\left[f\left(z^\circ - t\boldsymbol{G}^\circ\right)\right]. \tag{24}$$

(This is analogous to the basic inequality comparing the wait-and-see and the here-and-now approaches for classic two-stage stochastic programing problems, see, e.g., Chapter 4.3 of Birge and Louveaux 1997).

Let z' denote the minimizer of the line search on the left-hand side of (24), i.e., $f\left(z'\right) = \min_t\, f\left(z^\circ - t\boldsymbol{G}^\circ\right)$. (Of course z' is a random vector since it depends on $\boldsymbol{G}^\circ$.) Substituting this in (24) and comparing with (23), we get

$$\mathrm{E}\left[f\left(z'\right)\right] \;\leq\; f\left(z^\circ\right) \;-\; \frac{1}{2\omega(\sigma^2+1)}\; \left\|\boldsymbol{g}^\circ\right\|^2.$$

Subtracting $\mathcal{F}$ from both sides results in

$$\mathrm{E}\left[f\left(z'\right)\right] - \mathcal{F} \;\leq\; f\left(z^\circ\right) - \mathcal{F} \;-\; \frac{1}{2\omega(\sigma^2+1)}\; \left\|\boldsymbol{g}^\circ\right\|^2. \tag{25}$$

Coming to the lower bound, a well-known consequence of $\alpha I \preceq \nabla^2 f(z)$ is

$$\left\|\boldsymbol{g}^\circ\right\|^2 \;\geq\; 2\alpha\,\left(f\left(z^\circ\right)\;-\mathcal{F}\right)$$

(see Chapter 8.6 of Luenberger and Ye 2008). Combining this with (25), we get

$$\begin{aligned}\mathrm{E}\left[f\left(z'\right)\right]-\mathcal{F} &\leq f\left(z^\circ\right)-\mathcal{F}-\frac{\alpha}{\omega(\sigma^2+1)}\,\left(f\left(z^\circ\right)\;-\mathcal{F}\right)\\ &=\left(1-\frac{\alpha}{\omega(\sigma^2+1)}\right)\left(f\left(z^\circ\right)\;-\mathcal{F}\right). \qquad (26)\end{aligned}$$

As we have assumed that z° is a given (not random) vector, the right-hand side of (26) is deterministic, and the expectation on the left-hand side is considered according to the distribution of $\boldsymbol{G}^\circ$.

Now, let us examine the $(l+1)$th line search (for $1 \leq l \leq j-1$) where the starting point is $z^\circ = z^l$ and the minimizer is $z' = z^{l+1}$. Of course (26) holds with these objects also, but now both sides are random variables, depending on the vectors $\boldsymbol{G}^0, \ldots, \boldsymbol{G}^{l-1}$. (The expectation on the left-hand side is a conditional expectation.) We consider the respective expectations of the two sides, according to the joint distribution of $\boldsymbol{G}^0, \ldots, \boldsymbol{G}^{l-1}$. As the random gradient vectors were generated independently, we get

$$\mathrm{E}\left[f\left(z^{l+1}\right)\right]-\mathcal{F} \;\leq\; \left(1-\frac{\alpha}{\omega(\sigma^2+1)}\right)\left(\mathrm{E}\left[f\left(z^l\right)\right]\;-\mathcal{F}\right), \qquad (27)$$

where the left-hand expectation is now taken according to the joint distribution of $\boldsymbol{G}^0, \ldots, \boldsymbol{G}^l$. This technique of proof is well known in the context of stochastic gradient schemes, see, e.g., Nesterov and Vial (2008).

Finally, (22) follows from the iterative application of (27). □

Coming back to problem 1, let Assumptions 1 and 3 hold for the objective function $\phi(z)$. We show that the column generation scheme of Sect. 2.3 can be implemented as a randomized method using gradient estimates. Specifically, we need to approximately solve the column generation subproblem (15).

Corollary 11 *Let a tolerance* β $(0 < \beta \ll 1)$ *and a probability* p $(0 < p \ll 1)$ *be given. In* $O(-\log(\beta\, p))$ *steps with the stochastic descent method, we find a vector* $\widehat{z}$ *such that*

$$P\Big(\;\overline{\rho}\left(\widehat{z}\,\right)\;\geq\;(1-\beta)\overline{\mathcal{R}}\Big)\;\geq\;1-p.$$

Proof We apply Theorem 10 to $f(z) = -\overline{\rho}(z)$. This function inherits Assumptions 1 and 3 from $\phi(z)$. Let $\varrho = 1 - \frac{\alpha}{\omega(\sigma^2+1)}$ with some $\sigma > 0$. We assume that gradient estimates at the respective iterates are generated independently, and (20)–(21) hold for each of them.

Substituting $z^0 = \overline{z}$ in (22) and taking into account (14), we get

$$\mathrm{E}\left[\overline{\rho}(z^j)\right]\;\geq\;\left(1-\varrho^j\right)\,\overline{\mathcal{R}}.$$

The gap $\overline{\mathcal{R}}$ is obviously non-negative. In case $\overline{\mathcal{R}} = 0$, the starting iterate $z^0 = \overline{z}$ of the steepest descent method was already optimal, due to (14). In what follows we assume $\overline{\mathcal{R}} > 0$. A trivial transformation results in

$$\mathrm{E}\left[1-\frac{\overline{\rho}(z^j)}{\overline{\mathcal{R}}}\right]\;\leq\;\varrho^j.$$

By Markov's inequality, we get

$$\mathrm{P}\left(1-\frac{\overline{\rho}(z^j)}{\overline{\mathcal{R}}}\geq\beta\right)\;\leq\;\frac{\varrho^j}{\beta},$$

and a trivial transformation yields

$$\mathrm{P}\Big(\,\overline{\rho}(z^j\;)\;\leq\;(1-\beta)\overline{\mathcal{R}}\Big)\;\leq\;\frac{1}{\beta}\,\varrho^j.$$

Hence

$$\mathrm{P}\Big(\,\overline{\rho}(z^j\;)\;>\;(1-\beta)\overline{\mathcal{R}}\Big)\;\geq\;1-\frac{1}{\beta}\,\varrho^j.$$

Performing j steps with j such that $\varrho^j\leq\beta p$ yields an appropriate $\widehat{z}=z^j$. □

Remark 12 Gradients of the function $-\overline{\rho}(z)$ have the form $\nabla\phi(z)-\overline{\boldsymbol{u}}$. The further the column generation procedure progresses, the smaller the norm $\|\nabla\phi(\overline{z})-\overline{\boldsymbol{u}}\|$ gets (see Observation 4 *(c)*).

To satisfy the requirement (20) on variance, better and better estimates are needed. We control accuracy according to Assumption 3.

3.1 Bounding the optimality gap and reliability considerations for the randomized column generation scheme

By analogy with the deterministic scheme, let

$$\overline{\mathcal{B}}:=\frac{1}{1-\beta}\overline{\rho}\left(\widehat{z}\right), \tag{28}$$

with the β and $\widehat{z}$ of Corollary 11. Concerning the gap between the respective optima of the model problem (10) and the original convex problem (3), the reliability

$$\mathrm{P}\left(\overline{\mathcal{B}}\;\geq\;\text{`gap'}\right) \tag{29}$$

is at least $1-p$ with the p of Corollary 11.

Assume that our initial model included the columns $z_0,\ldots,z_\iota$. In the course of the column generation scheme, we select further columns according to Corollary 11, with gradient estimates generated independently. Let the parameters σ and β be fixed for the whole scheme, e.g., set $\beta=0.5$. On the other hand, we keep increasing the reliability of the individual steps during the process, i.e., let $p=p_\kappa\quad(\kappa=\iota+1,\iota+2,\ldots)$ decrease with κ.

Example 13 Given the number ι of the initial columns, let $p_\kappa\;=\;(\kappa-\iota+9)^{-2}\quad(\kappa\;=\;\iota+1,\iota+2,\ldots)$. Then we have $\prod_{\kappa=\iota+1}^{\infty}(1-p_\kappa)\;=0.9$. (This is easily proven. We learned it from Szász 1951, Volume II., Chapter X., Section 642).

To achieve reliability $1-p_\kappa$ set in Example 13, we need to make $O(\log\kappa)$ steps with the stochastic descent method when selecting the column z_κ.

We terminate the column generation process when $\overline{\mathcal{B}}$ of (28) gets below the prescribed accuracy. With the setting of Example 13, the terminal bound is correct with a probability at least 0.9, regardless of the number of new columns generated over the course of the procedure.

3.2 On stochastic gradient methods

The aim of this section is to place Theorem 10 and the column generation scheme into the broader context of stochastic gradient methods. The idea of stochastic approximation goes back to Robbins and Monro (1951). Important contributions include Ermoliev (1969), Gaivoronski (1978), Nemirovski and Yudin (1978, 1983), Nesterov (1983, 2009), Ermoliev (1983), Ruszczyński and Syski (1986), Uryasev (1988), Pflug (1988, 1996), Polyak (1990), Polyak and Juditsky (1992), Benveniste et al. (1993), Nemirovski et al. (2009), Lan (2012). The approach is attractive from a theoretical point of view, but early forms might perform poorly in practice. Recent forms combine theoretical depth with practical effectiveness.

We consider the problem

$$\min\ f(\boldsymbol{x}) \quad \text{subject to} \quad \boldsymbol{x} \in X, \tag{30}$$

where $X \subset \mathrm{I\!R}^n$ is a convex compact set, and $f : \mathrm{I\!R}^n \to \mathrm{I\!R}$ is a convex differentiable function. Our original problem (1) is easily transformed to this form.

Stochastic gradient methods are iterative, and a starting point $\boldsymbol{x}_1 \in X$ is needed. Let $\boldsymbol{x}_k \in X$ denote the kth iterate, and let G_k be a random estimate of the corresponding gradient $\boldsymbol{g}_k = \nabla f(z_k)$. Gradient estimates for different iterates are assumed to be based on independent, identically distributed samples. The next iterate is computed as

$$\boldsymbol{x}_{k+1} = \Pi_X\left(\boldsymbol{x}_k - h_k G_k\right), \tag{31}$$

where $h_k > 0$ is an appropriate step length, and Π_X denotes projection onto the feasible domain, i.e., $\Pi_X(\boldsymbol{x}) = \arg\min_{\boldsymbol{x}' \in X} \|\boldsymbol{x} - \boldsymbol{x}'\|$.

Methods differ in the construction of gradient estimates and in the determination of step lengths. Establishing an appropriate stopping rule is also a critical issue. Many of the methods apply averaging (like the example method sketched below), and some employ the dual space also.

As a recent example of the stochastic gradient approach, we sketch the robust stochastic approximation method of Nemirovski and Yudin. It is assumed that, given $\boldsymbol{x} \in X$, realizations of a random vector $\boldsymbol{G}$ can be constructed such that $\mathrm{E}(\boldsymbol{G}) = \nabla f(\boldsymbol{x})$, and $\mathrm{E}(\|\boldsymbol{G}\|^2) \leq M^2$ holds with a constant M independent of $\boldsymbol{x}$.

Nemirovski and Yudin prove different convergence results; from our present point of view, the most relevant one is the following. Suppose that we wish to perform N steps with the above procedure, and set step length to be constant:

$$h_k = \frac{\mathrm{diag}(X)}{M\sqrt{N}}, \tag{32}$$

where $\mathrm{diag}(X)$ is the longest (Euclidean) distance occurring in X. Then we have

$$\mathrm{E}\Big(f(\overline{\boldsymbol{x}}_N)\Big) - \mathcal{F} \leq \frac{M \cdot \mathrm{diag}(X)}{\sqrt{N}}, \tag{33}$$

where $\mathcal{F}$ denotes the minimum of (30), and

$$\overline{\boldsymbol{x}}_N = \sum_{j=1}^{N} \lambda_j^N \boldsymbol{x}_j \quad \text{with} \quad \lambda_j^N = \frac{h_j}{\sum_{j=1}^{N} h_j}. \tag{34}$$

Our present Assumption 1 is much stronger than mere differentiability, hence the convergence estimate of Theorem 10 is naturally stronger than (33).

We proved Theorem 10 for unconstrained minimization (over $\mathbb{R}^n$). In our approach, the constraint $A\boldsymbol{x} \leq \boldsymbol{b}$ in the convex problem (1) was taken into account through a column generation scheme. Comparing the column generation scheme with the above stochastic gradient approach, a solution of the linear programming model problem (10) is analogous to the iterate averaging (34) and the projection in (31). The analogy is not complete. Having solved the linear programming model problem, we perform an approximate line search instead of a simple translation by $-h_k G_k$. Larger effort of an individual step in the column generation scheme pays off when gradient estimation is taxing as compared to function value computation.

Having pondered a reviewer comment concerning further combinations of column generation and stochastic gradient schemes, we see a high potential in this approach. Different combinations of the column generation and stochastic gradient schemes may be efficient for functions with different characteristics.

4 Handling a difficult constraint

We work out an approximation scheme for the solution of the convex constrained problem (2). This scheme consists of the solution of a sequence of problems of the form (1), with a tightening stopping tolerance.

We consider the linear constraint set $A\boldsymbol{x} \leq \boldsymbol{b}$ of problem (1). The last constraint of this set is $\boldsymbol{a}^r\boldsymbol{x} \leq b_r$, where $\boldsymbol{a}^r$ denotes the rth row of A, and b_r denotes the rth component of $\boldsymbol{b}$. Assume that this last constraint is a cost constraint, and let $\boldsymbol{c}^T = \boldsymbol{a}^r$ denote the cost vector. We consider a parametric form of the cost constraint, namely, $\boldsymbol{c}^T\boldsymbol{x} \leq d$, where $d \in \mathbb{R}$ is a parameter.

Let $\breve{A}$ denote the matrix obtained by omitting the rth row in A, and let $\breve{\boldsymbol{b}}$ denote the vector obtained by omitting the rth component in $\boldsymbol{b}$. Using these objects, we consider the problem

$$\min\ \phi(T\boldsymbol{x}) \quad \text{subject to} \quad \breve{A}\boldsymbol{x} \leq \breve{\boldsymbol{b}}, \quad \boldsymbol{c}^T\boldsymbol{x} \leq d, \tag{35}$$

with the parameter $d \in \mathbb{R}$. This parametric form of the unconstrained problem will be denoted by (1: $b_r = d$).

Let $\chi(d)$ denote the optimal objective value of problem (35), as a function of the parameter d. This is obviously a monotone decreasing convex function. Let $\mathcal{I} \subset \mathbb{R}$ denote the domain over which the function is finite. We have either $\mathcal{I} = \mathbb{R}$ or $\mathcal{I} = [\underline{d}, +\infty)$ with some $\underline{d} \in \mathbb{R}$. Using the notation of the unconstrained problem, we say that $\chi(d)$ is the optimum of (1: $b_r = d$) for $d \in \mathcal{I}$.

Coming to the constrained problem (2), we may assume $\pi \in \chi(\mathcal{I})$. Let $d^\star \in \mathcal{I}$ be a solution of the equation $\chi(d) = \pi$, and let $l^\star(d)$ denote a linear support function to $\chi(d)$ at $d^\star$. In this section we work under.

Assumption 14 The support function $l^\star(d)$ has a significant negative slope, i.e., $l^{\star\prime} \ll 0$.

From $l^{\star\prime} < 0$, it follows that the optimal objective value of (2) is $d^\star$. (This slope will be used in estimating the number of Newton steps required to reach a prescribed accuracy; see Corollary 19, below. That is why we need it to be significantly negative.)

Remark 15 Assumption 14 is reasonable if the right-hand-side value π has been set by an expert, on the basis of preliminary experimental information. (A near-zero slope $l^{\star\prime}$ means that a slight relaxation of the probabilistic constraint allows a significant cost reduction.)

We find a near-optimal $\widehat{d} \in \mathcal{I}$ using an approximate version of Newton's method. The idea of regulating tolerances in such a procedure occurs in the discussion of the Constrained

Newton Method in Lemaréchal et al. (1995). Based on the convergence proof of the Constrained Newton Method, a simple convergence proof of Newton's method was reconstructed in Fábián et al. (2015). We adapt the latter to the present case.

First, we describe a deterministic approximation scheme. A randomized version is worked out in Sect. 4.2.

4.1 A deterministic approximation scheme

Let Assumptions 1 and 2 hold. A sequence of unconstrained problems (1: $b_r = d_\ell$) ($\ell = 1, 2, \ldots$) is solved with increasing accuracy. Over the course of this procedure, we build a single model $\phi_k(z)$ of the nonlinear objective $\phi(z)$, i.e., k is ever increasing. Columns added during the solution of (1: $b_r = d_\ell$) are retained in the model and reused in the course of the solution of (1: $b_r = d_{\ell+1}$).

Given the ℓth iterate $d_\ell \in \mathcal{I}$, we need to estimate $\chi(d_\ell)$ with a prescribed accuracy. This is done by performing a column generation scheme with the master problem (10: $b_r = d_\ell$). Let $\overline{\mathcal{B}}_\ell$ denote an upper bound on the gap between the respective optima of the model problem (10: $b_r = d_\ell$) and the convex problem (1: $b_r = d_\ell$). Such a bound is constructed according to the expression (18).

Let moreover $\overline{\chi}_\ell$ denote the optimum of the model problem. With these objects we have

$$\overline{\chi}_\ell \;\geq\; \chi(d_\ell) \;\geq\; \overline{\chi}_\ell - \overline{\mathcal{B}}_\ell. \tag{36}$$

The column generation process with the master problem (10: $b_r = d_\ell$) is terminated if $\overline{\chi}_\ell$ and $\overline{\mathcal{B}}_\ell$ satisfy a stopping condition, to be discussed below.

Let $d_0, d_1 \in \mathcal{I},\; d_0 < d_1 < d^\star$ be the starting iterates. The sequence of the iterates will be strictly monotone increasing, and converging to $d^\star$ from below.

4.1.1 Near-optimality condition for the constrained problem

Given a tolerance ϵ $(\pi \gg \epsilon > 0)$, let $\widehat{d} \in \mathcal{I}$ be such that

$$\widehat{d} \leq d^\star \quad \text{and} \quad \chi(\widehat{d}\,) \leq \pi + \epsilon. \tag{37}$$

Let $\widehat{\boldsymbol{x}}$ be an optimal solution of (35: $d = \widehat{d}$). Then $\widehat{\boldsymbol{x}}$ is an ϵ-feasible solution of (2) with objective value $\widehat{d}$. Exact feasible solutions of (2) have objective values not less than $d^\star \geq \widehat{d}$.

4.1.2 Stopping condition for the unconstrained subproblem

Let δ $(0 < \delta \ll \frac{1}{2})$ denote a fixed tolerance. (We can set e.g., $\delta = 0.25$ for the whole process).

Given iterate $d_\ell \in \mathcal{I},\; d_\ell \leq d^\star$, we perform a column generation scheme with the master problem (10: $b_r = d_\ell$). The process is terminated if either

$$\begin{aligned} &(i)\;\; \overline{\chi}_\ell - \pi \;\;\leq \epsilon, \quad \text{or} \\ &(ii)\; \overline{\mathcal{B}}_\ell \;\leq\; \delta\,(\overline{\chi}_\ell - \pi) \end{aligned} \tag{38}$$

holds. Taking into account (36), we conclude:

If *(i)* occurs then $\widehat{d} := d_\ell$ satisfies the near-optimality condition (37), and the Newton-like procedure stops.

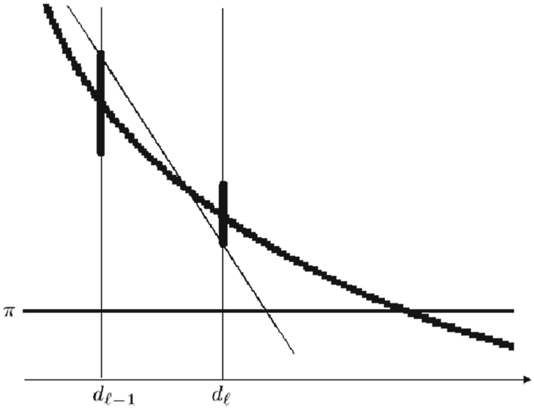

Fig. 1 The graph of the function $\chi(d)$, and the construction of the next iterate

If *(ii)* occurs then $\overline{\chi}_\ell$ satisfies

$$\overline{\chi}_\ell \ \geq \ \chi(d_\ell) \ \geq \ \overline{\chi}_\ell - \delta\,(\overline{\chi}_\ell - \pi). \tag{39}$$

A new iterate will be constructed in the latter case.

4.1.3 Finding successive iterates

Given $\ell \geq 1$, assume that we have bounded $\chi(d_{\ell-1})$ and $\chi(d_\ell)$, as in (39). The graph of the function $\chi(d)$ is shown in Fig. 1. Thick segments of the vertical lines $d = d_{\ell-1}$ and $d = d_\ell$ indicate confidence intervals for the function values $\chi(d_{\ell-1})$ and $\chi(d_\ell)$, respectively. Let $l_\ell : \mathbb{R} \to \mathbb{R}$ be the linear function determined by the upper endpoint of the former interval, and the lower endpoint of the latter one. Formally,

$$l_\ell(d_{\ell-1}) := \overline{\chi}_{\ell-1} \ \geq \chi(d_{\ell-1}) \quad \text{and} \quad l_\ell(d_\ell) := \overline{\chi}_\ell - \delta\,(\overline{\chi}_\ell - \pi) \ \leq \chi(d_\ell), \tag{40}$$

where the inequalities follow from (39).

Due to the convexity of $\chi(d)$ and to Assumption 14, the linear function $l_\ell(d)$ obviously has a negative slope $l'_\ell \leq l^{\star\prime} \ll 0$. Moreover $l_\ell(d) \leq \chi(d)$ holds for $d_\ell \leq d$.

The next iterate $d_{\ell+1}$ will be the point satisfying $l_\ell(d_{\ell+1}) = \pi$. Of course $d_\ell < d_{\ell+1} \leq d^\star$ follows from the observations above.

Remark 16 In a Newton-like scheme, the selection of the starting iterates strongly affects efficiency. Let us first consider the selection of d_1. An expert familiar with the model may easily set a budget slightly overtight. In the absence of such expert, we can resort to heuristics, evaluating $\chi(d)$ in a set of test points.

Once d_1 has been set, we can consider d_0. A good choice for d_0 is one that results in a large d_2. We have to take into account the accuracy of our evaluation of $\chi(d_1)$ on the one hand, and the slope of $\chi(d)$ on the other hand. A possible way of organizing the selection process is the following. First, we evaluate $\chi(d_1)$ by solving the problem (35: $d = d_1$). In the course of the solution, we build a model of the objective function. This model can then be used to estimate the slope of $\chi(d)$.

4.1.4 Convergence

Let the iterates $d_0, d_1, \ldots, d_s$ and the linear functions $l_1(d), \ldots, l_s(d)$ be as defined above. We assume that $s > 1$, and the procedure did not stop before step $(s+1)$. Then we have

$$\overline{\chi}_\ell - \pi > \epsilon \quad (j = 0, 1, \ldots, s). \tag{41}$$

To simplify the notation, we introduce the linear functions $L_\ell(d) := l_\ell(d) - \pi \quad (j = 1, \ldots, s)$. With these, (40) transforms into

$$L_\ell(d_{\ell-1}) = \overline{\chi}_{\ell-1} - \pi \quad \text{and} \quad L_\ell(d_\ell) = (1-\delta)(\overline{\chi}_\ell - \pi) \quad (j = 1, \ldots, s). \tag{42}$$

Positivity of the above function values follows from (41). Moreover, the derivatives satisfy

$$L'_\ell = l'_\ell \leq l^{\star\prime} \ll 0 \quad (j = 1, \ldots, s) \tag{43}$$

due to the observations in the previous section.

Theorem 17 *We have*

$$\gamma^{s-1} \cdot \frac{|L'_1|}{|l^{\star\prime}|} \cdot L_1(d_1) \geq \; L_s(d_s) \quad \textit{with} \quad \gamma := \left(\frac{1}{2(1-\delta)}\right)^2. \tag{44}$$

Proof The following statements hold for $j = 1, \ldots, s-1$. From (42), we get

$$\frac{L_{\ell+1}(d_\ell)}{L_\ell(d_\ell)} \;=\; \frac{\overline{\chi}_\ell - \pi}{(1-\delta)(\overline{\chi}_\ell - \pi)} \;=\; \frac{1}{1-\delta}. \tag{45}$$

By definition, we have

$$L_\ell(d_\ell) + (d_{\ell+1} - d_\ell)\, L'_\ell \;=\; L_\ell(d_{\ell+1}) \;= 0.$$

It follows that $d_{\ell+1} - d_\ell = \frac{L_\ell(d_\ell)}{|L'_\ell|}$. Using this, we get

$$L_{\ell+1}(d_\ell) \;=\; L_{\ell+1}(d_{\ell+1}) + (d_\ell - d_{\ell+1})\, L'_{\ell+1} \;=\; L_{\ell+1}(d_{\ell+1}) + \frac{L_\ell(d_\ell)}{|L'_\ell|}\, |L'_{\ell+1}|.$$

Hence

$$\frac{L_{\ell+1}(d_\ell)}{L_\ell(d_\ell)} \;=\; \frac{L_{\ell+1}(d_{\ell+1})}{L_\ell(d_\ell)} + \frac{|L'_{\ell+1}|}{|L'_\ell|}. \tag{46}$$

From (45), we have

$$\frac{1}{1-\delta} \;=\; \frac{L_{\ell+1}(d_{\ell+1})}{L_\ell(d_\ell)} + \frac{|L'_{\ell+1}|}{|L'_\ell|} \;\geq\; 2\sqrt{\frac{L_{\ell+1}(d_{\ell+1})\,|L'_{\ell+1}|}{L_\ell(d_\ell)\,|L'_\ell|}}.$$

(This is the well-known inequality between means.) It follows that

$$\left(\frac{1}{2(1-\delta)}\right)^2 \; L_\ell(d_\ell)\,|L'_\ell| \;\geq\; L_{\ell+1}(d_{\ell+1})\,|L'_{\ell+1}|. \tag{47}$$

By induction, we get

$$\left(\frac{1}{2(1-\delta)}\right)^{2(s-1)} \; L_1(d_1)\,|L'_1| \;\geq\; L_s(d_s)\,|L'_s|. \tag{48}$$

Applying $|L'_s| \geq |l^{\star\prime}|$ we obtain (44). □

Example 18 Let $\delta = 0.25$, then $\gamma = (\frac{1}{2(1-\delta)})^2 \; < 0.5$.

Corollary 19 *With the setting of Example 18, the number of Newton-like steps needed to reach the stopping tolerance ϵ does not exceed*

$$N(\epsilon) = \; \log\left(\frac{|L_1'|}{|l^{\star\prime}|} \cdot \frac{L_1(d_1)}{\epsilon}\right). \tag{49}$$

Note that $|l^{\star\prime}| \gg 0$ due to Assumption 14.

Given a problem, let us consider the efforts of its approximate solution as a function of the prescribed accuracy ϵ. According to (49), that is on the order of $\log \frac{1}{\epsilon}$.

4.2 A randomized version of the approximation scheme

Let Assumptions 1 and 3 hold.

Concerning the function $\chi(d)$, let Assumption 14 hold. Our aim, in principle, is the same as it has been in the deterministic case: find $\widehat{d} \in \mathcal{I}$ such that $\pi + \epsilon \geq \chi(\widehat{d}) \geq \pi$ holds with a pre-set tolerance ϵ. In the present uncertain environment, however, we may have to content ourselves with $\hat{d}$ such that $\pi + \epsilon \geq \chi(\hat{d}) > \pi - \epsilon$ holds. This problem statement is justifiable if the function $\chi(d)$ is not constant for $d > d^\star$. Let Assumption 20, below, hold.

Assumption 20 For our stopping tolerance ϵ, there exists (an unknown) $d_\epsilon^\star \in \mathcal{I}$ such that $\chi(d_\epsilon^\star) = \pi - \epsilon$.

Let q $(0.5 \ll q < 1)$ denote a pre-set reliability. Using the randomized column generation scheme, a sequence of unconstrained problems $(1\colon\; b_r = d_\ell)$ $\;(\ell = 1, 2, \ldots)$ is solved, each with reliability q, and with an accuracy determined by the Newton-like approximation scheme. As in the deterministic case, we build a single model $\phi_k(z)$ of the nonlinear objective $\phi(z)$, i.e., k is ever increasing. Let $k_{\ell-1}$ denote the number of columns at the outset of the solution of problem $(1\colon\; b_r = d_\ell)$.

Given the ℓth iterate $d_\ell \in \mathcal{I}$, we estimate $\chi(d_\ell)$ by performing a column generation scheme with the master problem $(10\colon\; b_r = d_\ell)$. Applying the procedure of Sect. 3.1, we obtain an estimate $\overline{\mathcal{B}}_\ell$ for the gap between the respective optima of the model problem $(10\colon\; b_r = d_\ell)$ and the convex problem $(1\colon\; b_r = d_\ell)$. Keeping to the setting of Example 13, we set the reliability parameter to $q = 0.9$, obtaining $\mathrm{P}(\overline{\mathcal{B}}_\ell \; \geq$ ‘gap’ $) \; \geq 0.9$. (Note that the columns with indices up to $k_{\ell-1}$ belong to the initial model, hence in terms of Sect. 3.1, we have $\iota = k_{\ell-1}$.)

Let moreover $\overline{\chi}_\ell$ denote the optimum of the model problem. With these objects we have

$$\overline{\chi}_\ell \;\geq\; \chi(d_\ell) \quad \text{and} \quad \mathrm{P}\left(\chi(d_\ell) \geq \overline{\chi}_\ell - \overline{\mathcal{B}}_\ell\right) \geq 0.9. \tag{50}$$

We proceed in accordance with the deterministic scheme. The present stochastic scheme actually coincides with the deterministic one, provided the gap is estimated correctly in the unconstrained problem. In the stochastic scheme, however, we may underestimate the gap, meaning that $\overline{\mathcal{B}}_\ell$ is not an upper bound. Consequently the inequality $\chi(d_\ell) \;\geq\; \overline{\chi}_\ell - \overline{\mathcal{B}}_\ell$ may not hold in (50). In such a case, $d_{\ell+1} \;>\; d^\star$ and hence $\overline{\chi}_{\ell+1} \;<\; \pi$ may occur. If the latter is observed, then we step back to the previous iterate, i.e., set $d_{\ell+2} = d_\ell$. We then carry on with the Newton-like procedure; first resolving the model problem $(10\colon\; b_r = d_{\ell+2})$ with reliability $q = 0.9$.

4.2.1 Stopping condition for the unconstrained subproblem

In accordance with the above discussion, we now formulate the stopping condition of the column generation process at the Newton-like step ℓ. Solution with the master problem (10: $b_r = d_\ell$) is terminated if $\overline{\chi}_\ell$ and $\overline{\mathcal{B}}_\ell$ satisfy one of the following conditions:

$$\begin{aligned}&(\alpha)\ \overline{\chi}_\ell < \pi,\\ &(\beta)\ \pi \le \overline{\chi}_\ell < \pi + \epsilon \quad \text{and} \quad \overline{\mathcal{B}}_\ell \le \epsilon,\\ &(\gamma)\ \pi + \epsilon \le \overline{\chi}_\ell \quad \text{and} \quad \overline{\mathcal{B}}_\ell \ \le\ \delta\,(\overline{\chi}_\ell - \pi).\end{aligned} \tag{51}$$

If condition (α) occurs, then we step back to the previous iterate $d_{\ell-1}$.

If condition (β) occurs, then we stop the Newton-like process.

If condition (γ) occurs, then we carry on to a new iterate $d_{\ell+1} > d_\ell$, like we did in the deterministic scheme.

4.2.2 Convergence and reliability

Let the unconstrained subproblems each be solved with a reliability of $q = 0.9$, and let δ, γ be set according to Example 18. Moreover, let us assume that the randomized Newton-like scheme did not stop in L steps. The aim of this section is to show that, provided L is large enough, an ϵ-optimal solution of the constrained problem has been reached with a high probability.

According to our assumption, case (β) did not occur in the stopping condition of the previous section. Let us define 'correct' and 'incorrect' steps, depending on the starting point d_ℓ:

- In case $d_\ell \le d^\star$: We call step ℓ correct if $d_{\ell+1} \le d^\star$ and $0.5 \cdot L_\ell(d_\ell)\,|L'_\ell| \ge L_{\ell+1}(d_{\ell+1})\,|L'_{\ell+1}|$ also holds, otherwise we call step ℓ incorrect.
- In case $d_\ell > d^\star$:
  We call step ℓ correct if a backstep occurs (i.e., if $d_{\ell+1} = d_{\ell-1}$), otherwise we call it incorrect.

A step is correct with a probability at least $q = 0.9$; this follows from the proof of Theorem 17, namely the expression (47).

If the difference between the number of the correct steps and the number of the incorrect steps exceeds $N(\epsilon)$, then an ϵ-optimal solution of the constrained problem has been reached, according to Corollary 19.

Let Z_ℓ be the random variable

$$Z_\ell = \begin{cases} 0 \text{ if step } \ell \text{ is correct,} \\ \\ 1 \text{ if step } \ell \text{ is incorrect} \end{cases} \qquad (\ell = 1, \ldots, L).$$

As a step is correct with a probability at least $q = 0.9$, we have $\mathrm{E}(Z_\ell) \le 0.1$, and hence $\mathrm{E}(\sum_{\ell=1}^L Z_\ell) \le 0.1L$.

The difference between the number of the correct steps and the number of the incorrect steps is $L - 2\sum_{\ell=1}^L Z_\ell$. In order to show that the difference likely exceeds $N(\epsilon)$, we need an upper bound on the probability that $\sum_{\ell=1}^L Z_\ell$ is significantly larger than $\mathrm{E}(\sum_{\ell=1}^L Z_\ell)$.

Though all the gradient estimates were generated independently, there may be some interdependence among the random variables $Z_1, \ldots, Z_L$, because of the time structure of the process. But this interdependence is weak in the following sense. Suppose that we are at the beginning of the process. Given $0 < k \leq L$, we know that step k will be correct with a probability at least 0.9, no matter what happens in steps $1, \dots, k-1$. In particular,

$$\mathrm{P}\Big[\ Z_k = 1 \ \Big|\ Z_\ell = 1\ (\ell \in \mathcal{I}_k)\ \Big] \leq 0.1 \quad \text{holds for every} \ \ \mathcal{I}_k \subseteq \{1, \dots, k-1\}. \tag{52}$$

The condition in the above probability represents the event that $Z_\ell = 1$ occurs for every $\ell \in \mathcal{I}_k$. In case $k = 1$, the condition is empty, and (52) reduces to $\mathrm{P}(Z_1 = 1) \leq 0.1$.

Generalized Chernoff–Hoeffding bounds were proposed by Panconesi and Srinivasan (1997). Intuitive proofs of such bounds, based on a simple combinatorial argument, were given by Impagliazzo and Kabanets (2010). (In this latter paper, concentration bounds are also explained in terms of successes of random experiments, just our present situation.) We are going to use a Chernoff-type bound, Theorem 1.1 in Impagliazzo and Kabanets (2010):

Theorem 21 *Let $Z_1, \dots, Z_n$ be Boolean random variables such that, for some $p \in [0, 1]$,*

$$P\,[\,Z_\ell = 1\ (\ell \in A)\,] \ \leq \ p^{|A|} \quad \textit{holds for every} \ \ A \subseteq \{1, \dots, n\}, \tag{53}$$

where $|A|$ denotes the cardinality of A.

Then, for any $\kappa \in [p, 1]$, we have

$$P\left[\sum_{\ell=1}^{n} Z_\ell \ \geq \kappa n\right] \ \leq \ e^{-nD(\kappa||p)}, \tag{54}$$

where $D(\cdot||\cdot)$ is the relative entropy function, satisfying $D(\kappa||p) \geq 2(\kappa - p)^2$.

It is easy to see that our objects satisfy the precondition (53) with $p = 0.1$. Indeed, it follows from the repeated application of (52). A formal proof may apply induction on n. For $n = 1$, we have $\mathrm{P}(Z_1 = 1) \leq 0.1$. Now let us assume that (53) holds for $1 \leq n < k$. The statement for $n = k$ follows from (52), by setting $\mathcal{I}_k = A \cap \{1, \dots, k-1\}$.

As the precondition of Theorem 21 holds, we have (54) with $n = L,\ p = 0.1$ and $\kappa = 1/3$. Simple computation shows that, for $L \geq 22$,

$$\mathrm{P}\left[\sum_{\ell=1}^{L} Z_\ell \ < \frac{1}{3} L\right] \ \geq 0.9. \tag{55}$$

As we have seen, the difference between the number of the correct steps and the number of the incorrect steps is $L - 2\sum_{\ell=1}^{L} Z_\ell$ which exceeds $L/3$ if $\sum_{\ell=1}^{L} Z_\ell \ < 1/3L$ in (55) holds. We sum up the discussion in

Proposition 22 *Let the unconstrained problems each be solved with a reliability of $q = 0.9$; let δ, γ be set according to Example 18; and let $L = \max\{\,22,\ 3N(\epsilon)\,\}$ with $N(\epsilon)$ defined in Corollary 19.*

Assume that the randomized Newton-like scheme did not stop in L steps. Then an ϵ-optimal solution of the constrained problem has been reached with a probability at least 0.9.

Remark 23 If case (β) occurred in the stopping condition of the previous section, then further checks are needed to ensure reliability.

Remark 24 The stopping tolerance prescribed for the unconstrained subproblems is ever tightened in accordance with the progress of the Newton-like approximation scheme. However, the prescribed tolerance is never tighter than $\delta \cdot \epsilon = 0.25\epsilon$.

5 Adapting the approach to probabilistic problems

In this section we consider $\phi(z) = -\log F(z)$ with a nondegenerate n-dimensional standard normal distribution function $F(z)$. Assumption 1 does not hold with such a function. However, as illustrated in Fábián et al. (2018), Assumption 1 holds over any bounded ball around the origin. (The ratio $\frac{\alpha}{\omega}$ decreases as the radius of the ball increases.) Moreover, a construction was sketched in Fábián et al. (2018) that limits the column generation process to a ball of sufficiently large radius. That construction does not yield usable estimates for the values α and ω.

When applied to probabilistic problems, we look on Corollaries 6 and 11 merely as a means of justification of the efficiency of the procedure. The gap between the respective optima of the model problem and the original probabilistic problem is measured by different means, to be described presently. In this setting we may perform just a single line search in each column generation problem.

In order to apply the procedures described in the previous sections, we need Assumption 3 to hold, with the relaxation that function values $\phi(z)$ are computed with a high accuracy (instead of exactly). In the present case of a probabilistic $\phi(z)$, high-precision computation of $\log F(z)$ is impractical in points z with a low $F(z)$. Hence we need a technical assumption that helps keeping the process in a region where high-precision computation of $\phi(z)$ is possible.

Assumption 25 A significantly high probability can be achieved. Specifically, a feasible point $\check{z}$ is known such that $F(\check{z}) \geq 0.5$.

By including $\check{z}$ of Assumption 25 among the initial columns of the master problem, we always have $F(\overline{z}) \geq 0.5$ with the current solution $\overline{z}$ defined in (12). Hence $\phi(\overline{z})$ can be computed with a high accuracy.

We perform a single line search in each column generation subproblem, starting always from the current $\overline{z}$. It means that a high-quality estimate can be generated for the gradient, which designates the direction of the line search. Once the direction of the search is determined, we only work with function values (there is no need for any further gradient information in the current column generation subproblem). The line search is performed with a high accuracy over the region $\mathcal{L}(F, 0.5) = \{ z \mid F(z) \geq 0.5 \}$ which includes the optimal solution of the probability maximization problem (3).

We can carry on with the line search even if we have left the safe region $\mathcal{L}(F, 0.5)$. Given a point $\hat{z}$ along the search ray, let $\hat{p} > 0$ be such that $\hat{p} \leq F(\hat{z})$ holds almost surely. (Simulation procedures generally provide a confidence interval together with an estimate.) If the vector $\hat{z}$ is to be included in the master problem (10) as a new column, then we set the corresponding cost coefficient as $\phi = -\log \hat{p}$. Under such an arrangement, our model remains consistent, i.e., the model function $\phi_k(z)$ is almost surely an inner approximation of the probabilistic function $\phi(z)$.

5.1 A bounded formulation

Exploiting monotonicity of the function $\phi(z) = -\log F(z)$, the unconstrained problem with variable splitting is formulated with inequality between z and $T\boldsymbol{x}$:

$$\min \ \phi(z) \quad \text{subject to} \quad A\boldsymbol{x} - \boldsymbol{b} \leq \boldsymbol{0}, \quad z - T\boldsymbol{x} \leq \boldsymbol{0}. \tag{56}$$

A further speciality of the normal distribution function is the existence of a bounded box $\mathcal{Z}$ outside which the probability weight can be ignored. Including the constraint $z \in \mathcal{Z}$ in

(56) results in a closely approximating problem:

$$\min\ \phi(z) \quad \text{subject to} \quad A\boldsymbol{x}-\boldsymbol{b} \leq \boldsymbol{0}, \quad z - T\boldsymbol{x} \leq \boldsymbol{0}, \quad z \in \mathcal{Z}. \tag{57}$$

Observation 26 *The difference between the respective optima of problems* (56) *and* (57) *is insignificant.*

Proof Let z be a part of a feasible solution of (56), and let us consider the box $(z+\mathcal{N}) \cap \mathcal{Z}$, where $\mathcal{N}$ denotes the negative orthant. In case this box is empty, we have $F(z) \approx 0$ due to the specification of $\mathcal{Z}$. Taking into account Assumption 25, such z cannot be a part of an optimal solution of (56).

In case the box $(z+\mathcal{N}) \cap \mathcal{Z}$ is not empty, let Π_z denote its 'most positive' vertex. We have $\Pi_z \in \mathcal{Z},\ \Pi_z \leq z$, and $F(\Pi_z) \approx F(z)$. If $F(z) < 0.5$, then, due to Assumption 25 again, z cannot be a partial optimal solution of (56).

In the remaining case of $F(\Pi_z) \approx F(z) \geq 0.5$, we have $\phi(\Pi_z) \approx \phi(z)$. Moreover Π_z is a partial feasible solution of (57), due to $\Pi_z \in \mathcal{Z},\ \Pi_z \leq z$. □

Remark 27 We could build duals and models of the above forms in the manner of Sect. 2. Observation 26 allows us to restrict z to $\mathcal{Z}$ in the dual formulation (4). Formally, this would mean working with the restricted functions

$$\phi_{\mathcal{Z}}(z) = \begin{cases} \phi(z) \text{ if } & z \in \mathcal{Z}, \\ \\ +\infty & \text{otherwise} \end{cases} \qquad \text{and} \quad \phi^{\star}_{\mathcal{Z}}(\boldsymbol{u}) = \max_{z\in\mathcal{Z}}\{\boldsymbol{u}^T z - \phi(z)\} \tag{58}$$

instead of $\phi(z)$ and $\phi^{\star}(\boldsymbol{u})$, respectively.

In a pure form of this bounded scheme, new columns would always be selected from $\mathcal{Z}$. An obvious drawback of such a scheme is that Theorem 5 does not apply to the resulting bounded optimization problem. In Sect. 5.2 we develop a hybrid scheme, including a restriction to $\mathcal{Z}$ in the master problem, but selecting new columns by unconstrained maximization.

5.2 A hybrid form of the column generation scheme

Introducing new variables $z' \in \mathbb{R}^n$, we transform (57) to

$$\min\ \phi(z) \quad \text{subject to} \quad A\boldsymbol{x}-\boldsymbol{b} \leq \boldsymbol{0}, \quad z' - T\boldsymbol{x} \leq \boldsymbol{0}, \quad z' \in \mathcal{Z}, \quad z = z'. \tag{59}$$

The above problem has the general pattern of (3), hence the dual problem can be formulated in the manner of Sect. 2, relaxing the equality constraint $z = z'$. Model problems are then formulated according to Sects. 2.1 and 2.2. Hence the columns z_i $(i = 0, \ldots, k)$ may, in theory, fall outside $\mathcal{Z}$, but their convex combination is restricted to $\mathcal{Z}$.

We implemented this procedure. In our experiments reported in Sect. 7, the restriction $z' \in \mathcal{Z}$ was never active in any optimal solution of the master problem.

Let $\overline{z} \in \mathcal{Z}$ denote the current primal iterate, obtained in the form (12) using an optimal solution of the model problem. Let $\overline{\boldsymbol{g}} = \nabla\phi(\overline{z})$ be the corresponding gradient. Let moreover $(\overline{\vartheta}, \overline{\boldsymbol{u}})$ be part of an optimal dual solution of the current model problem. Finally, let $\overline{\mathcal{R}}$ denote the gap between the respective optima of the model problem and the original probabilistic problem.

Observation 28 *With the above objects, we have:*

$$\overline{\mathcal{R}} \le \Big(\phi_k(\overline{z}) - \phi(\overline{z})\Big) + \max_{z \in \mathcal{Z}} (\overline{\boldsymbol{u}} - \overline{\boldsymbol{g}})^T (z - \overline{z}). \tag{60}$$

Proof An adaptation of Observation 7 to the present bounded setting is

$$\overline{\mathcal{R}} = \max_{z \in \mathcal{Z}} \left\{ \overline{\vartheta} + \overline{\boldsymbol{u}}^T z - \phi(z) \right\}.$$

Applying $\phi(z) \ge \phi(\overline{z}) - \overline{\boldsymbol{g}}^T (z - \overline{z})$ we get

$$\begin{aligned}\overline{\mathcal{R}} \le \overline{\vartheta} - \phi(\overline{z}) + \overline{\boldsymbol{g}}^T \overline{z} \ + \max_{z \in \mathcal{Z}} (\overline{\boldsymbol{u}} - \overline{\boldsymbol{g}})^T z = \phi_k(\overline{z}) - \Big(\overline{\vartheta} + \overline{\boldsymbol{u}}^T \overline{z}\Big) \ + \overline{\vartheta} - \phi(\overline{z}) + \overline{\boldsymbol{g}}^T \overline{z} \\ + \max_{z \in \mathcal{Z}} (\overline{\boldsymbol{u}} - \overline{\boldsymbol{g}})^T z.\end{aligned}$$

The above equality is a consequence of Observation 4 *(a)*. □

The exact gradient $\overline{\boldsymbol{g}}$ is of course not known, but we can construct a gradient estimate together with a confidence interval, as will be described in Sect. 6.5. Given an error tolerance $\Delta > 0$, a probability p $(0 < p \ll 1)$ and the iterate $\overline{z} \in \mathcal{Z}$, we obtain a gradient estimate $\overline{\boldsymbol{G}}$ together with a confidence interval $\overline{\mathcal{I}}$. (The former is a random vector, the latter is an n-dimensional interval having random dimensions. The interval has the vector as a center.) The random objects satisfy the following rules:

$$\mathrm{E}(\overline{\boldsymbol{G}}) = \overline{\boldsymbol{g}}, \quad \mathrm{P}\left(\overline{\boldsymbol{g}} \in \overline{\mathcal{I}}\right) \ge 1 - p \quad \text{and} \quad \mathrm{diag}\left(\overline{\mathcal{I}}\right) \le \Delta, \tag{61}$$

where diag denotes the largest distance in the interval.

The following observation shows that we can use $\overline{\boldsymbol{G}}$ to estimate the maximum on the right-hand side of (60).

Observation 29 *The objects of* (61) *admit the following estimate:*

$$\begin{aligned}\max_{z \in \mathcal{Z}} (\overline{\boldsymbol{u}} - \overline{\boldsymbol{g}})^T (z - \overline{z}) \quad &\le \quad \max_{z \in \mathcal{Z}} (\overline{\boldsymbol{u}} - \overline{\boldsymbol{G}})^T (z - \overline{z}) \ + \ \Delta \cdot \mathrm{diag}(\mathcal{Z}) \\ \textit{holds with a probability at least} \ \ 1 - p.\end{aligned} \tag{62}$$

Proof Based on the confidence interval, a pessimist estimate of the left-hand side of (62) could be obtained by solving the (nonconvex) quadratic programming problem

$$\max \ (\overline{\boldsymbol{u}} - \boldsymbol{g})^T (z - \overline{z}) \quad \text{such that} \quad z \in \mathcal{Z}, \quad \boldsymbol{g} \in \overline{\mathcal{I}}. \tag{63}$$

Instead of the quadratic programming problem, we just solve the linear programming problem

$$\max \ (\overline{\boldsymbol{u}} - \overline{\boldsymbol{G}})^T (z - \overline{z}) \quad \text{such that} \quad z \in \mathcal{Z}. \tag{64}$$

Let $\left(\grave{z}, \grave{\boldsymbol{g}}\right)$ denote an optimal solution of (63), and let $\widehat{z}$ denote an optimal solution of (64). The difference between the respective optima is

$$\begin{aligned}(\overline{\boldsymbol{u}} - \grave{\boldsymbol{g}})^T (\grave{z} - \overline{z}) - (\overline{\boldsymbol{u}} - \overline{\boldsymbol{G}})^T (\widehat{z} - \overline{z}) &\le (\overline{\boldsymbol{u}} - \grave{\boldsymbol{g}})^T (\grave{z} - \overline{z}) - (\overline{\boldsymbol{u}} - \overline{\boldsymbol{G}})^T (\grave{z} - \overline{z}) \\ &= (\overline{\boldsymbol{G}} - \grave{\boldsymbol{g}})^T (\grave{z} - \overline{z}),\end{aligned} \tag{65}$$

the inequality being a consequence of the selection of $\widehat{z}$. The estimate (62) follows by the Cauchy–Bunyakovsky–Schwarz inequality. □

To sum up the above discussion: based on a gradient estimate $\overline{\boldsymbol{G}}$ satisfying (61), it is easy to compute

$$\overline{\mathcal{B}} := \Big(\phi_k(\overline{z}) - \phi(\overline{z})\Big) + \max_{z\in\mathcal{Z}} (\overline{\boldsymbol{u}} - \overline{\boldsymbol{G}})^T (z - \overline{z}) \; + \; \Delta \cdot \operatorname{diag}(\mathcal{Z}). \tag{66}$$

According to Observations 28 and 29, we have $\mathrm{P}(\, \overline{\mathcal{B}} \geq \text{‘gap’}\,) \geq 1 - p$.

5.3 Regulating accuracy and reliability when solving an unconstrained probabilistic problem

Given iterate $\overline{z}$, we wish to construct an estimate $\overline{\boldsymbol{G}}$ for the corresponding gradient. We have two objectives.

- We need Corollary 11 to ensure efficiency of a descent step in the course of column selection. Hence (20) should hold with an appropriate σ between the vectors $\boldsymbol{g}^\circ = \overline{\boldsymbol{g}} - \overline{\boldsymbol{u}}$ and $\boldsymbol{G}^\circ = \overline{\boldsymbol{G}} - \overline{\boldsymbol{u}}$. Specifically,

$$\mathrm{E}\left(\left\|\overline{\boldsymbol{G}} - \overline{\boldsymbol{g}}\right\|^2\right) \; \leq \; \sigma^2 \left\|\overline{\boldsymbol{g}} - \overline{\boldsymbol{u}}\right\|^2 \tag{67}$$

  should hold.
- We need (61) to hold with appropriate parameters Δ and p to ensure that the bound $\overline{\mathcal{B}}$ is tight and reliable. We slightly re-formulate the definition of $\overline{\mathcal{B}}$ in (66) as follows:

$$\Big(\phi_k(\overline{z}) - \phi(\overline{z})\Big) + \max_{z\in\mathcal{Z}} \Big((\overline{\boldsymbol{u}} - \overline{\boldsymbol{g}}) + (\overline{\boldsymbol{g}} - \overline{\boldsymbol{G}})\Big)^T (z - \overline{z}) \; + \; \Delta \cdot \operatorname{diag}(\mathcal{Z}). \tag{68}$$

Concerning p, we increase reliability with each master iteration, as we did in the general case of Sect. 3.1. Having added κ columns, we prescribe the reliability $1 - p_\kappa$, with p_κ set according to Example 13.

In setting the parameters σ and Δ, we aim to find a balance between the error of the polyhedral model function on the one hand, and the error of the gradient estimation on the other hand. According to Observation 4 *(c)*, $\overline{\boldsymbol{u}} \in \partial\phi_k(\overline{z})$ holds. Taking into account $\overline{\boldsymbol{g}} = \nabla\phi(\overline{z})$, the vector $\overline{\boldsymbol{u}} - \overline{\boldsymbol{g}}$ in (67) and (68) represents the gradient error of the polyhedral model function $\phi_k(z)$. Similarly, $\phi_k(\overline{z}) - \phi(\overline{z})$ in (68) represents the error in function value. On the other hand, the vector $\overline{\boldsymbol{G}} - \overline{\boldsymbol{g}}$ in (67) and (68) represents the error of the gradient estimate $\overline{\boldsymbol{G}}$.

A balance between those two types of error is found by a two-stage procedure. We begin with estimating the order of the magnitude of $\|\overline{\boldsymbol{u}} - \overline{\boldsymbol{g}}\|$, and then refine the estimation as needed.

6 Estimation of the multivariate normal probability distribution function values and gradients

If a multivariate probability distribution function is differentiable everywhere then its partial derivatives have the general formula

$$\frac{\partial F(z_1, \ldots, z_n)}{\partial z_i} = F(z_1, \ldots, z_{i-1}, z_{i+1}, \ldots, z_n \,|\, z_i) f_i(z_i) \tag{69}$$

where $F(z_1, \ldots, z_n)$ is the probability distribution function of the random variables $\xi_1, \ldots, \xi_n$, and $f_i(z)$ is the probability density function of the random variable ξ_i. $F(z_1, \ldots,$

$z_{i-1}, z_{i+1}, \ldots, z_n | \, z_i)$ is the conditional probability distribution function of the random variables $\xi_1, \ldots, \xi_{i-1}, \; \xi_{i+1}, \ldots, \xi_n$, given that $\xi_i = z_i$. See Formula (6.6.22) on page 203 of the book by Prékopa (1995).

It is known that any conditional probability distribution of the multivariate normal probability distribution is also normal. Therefore from Formula (69) it follows that we can calculate the multivariate normal probability distribution function values and their partial derivatives by the same procedure. This is the reason why in this section we give a list of possible procedures for the estimation of multivariate probability distribution function values only.

6.1 Genz's method

This method was published in Genz (1992). In this paper Genz was dealing with the estimation of the multivariate normal probability content of a rectangle, which is a more general problem than the calculation of multivariate probability distribution function values.

The main idea is to transform the integration region to the unit cube $[0, 1]^n$ by a sequence of elementary transformations. This comes at the expense of a slightly more complicated integrand.

The sequence begins with the Cholesky transformation which transforms the components of the multivariate normally distributed random vector into independent random variables, however the integration limits become more complicated. Then the integration variables are transformed further by the inverse function of the one dimensional standard normal probability distribution function. The effect of this transformation is that all integrands will be equal one but the integration limits become even more complicated. Finally, by a simple linear transformation, the integration region changes to the unit cube $[0, 1]^n$ and the integrand functions will be the differences of the earlier complicated integration limits.

We remark that the i-th integrand function is always independent of the i-th integrand variable and can be pulled out of one integral which allows explicit integration of the innermost integral. This way the numerical integration may be carried out on the unit cube $[0, 1]^{n-1}$.

This sequence of transformations has also forced a priority ordering on the components of $\mathbf{x}$ which makes the problem amenable to the application of subregion adaptive algorithms. The method works best if the components are presorted so that the innermost integration has the most "weight".

Genz describes three different methods for solving this transformed integral. The first method is based on a polynomial approximation of the integrand. For better performance, the unit cube is split into subregions which are subsequently partitioned further whenever the approximation is not accurate enough. The second method uses quasi-random integration points. Finally, the third method uses pseudo-random integration points, which results in error estimates that are statistical in nature.

6.2 Deák's method

This method was first published in Deák (1980) and later in Deák (1986). Its main thrust is to decompose the normal random vector into two parts, a direction and a distance from the origin. This decomposition can be used both in the generation of sample points and in the calculation of the probability content of a rectangle. It is well known that the direction is uniformly distributed on the n-dimensional unit sphere, the distance from the origin has a chi-distribution with n degrees of freedom and they are independent of each other.

A simple Monte Carlo method is to generate N sample points uniformly distributed on the n-dimensional unit sphere, determine the probability content of the intersection of the rectangle in issue with the generated directions and finally average them. The determination of the probability content of the intersection can be done simply by applying a code to calculate the probability distribution function of the chi-distribution. The advantage of this method is that it counts the probability content of the rectangle not in a 'point to point' way, rather in a 'line section to line section' way.

In addition it is easy to apply some type of antithetic random variables technique to reduce the variance further. Deák devised an improvement over this scheme that is intended to distribute a large number of directions as uniformly as possible on the unit sphere.

A set of n directions is chosen first and converted into an orthonormal system, that is, n unit vectors which are mutually orthogonal. From each orthonormal system $\{s_1, \ldots, s_n\}$ one obtains $2^k \binom{n}{k}$ directions by computing the sum

$$d(v, l_1, \ldots, l_k) = \frac{1}{\sqrt{k}} \sum_{j=1}^{k} v_j s_{l_j},$$

where $v = (v_1, \ldots, v_k)$ is a sign vector (each component is either $+1$ or -1, and $1 \leq l_1 < \cdots < l_k \leq n$.

The estimator can then be calculated jointly for the set of $2^k \binom{n}{k}$ directions resulting in faster calculation and further variance reduction. The parameter k can in principle be chosen arbitrarily from the set $\{1, 2, \ldots, n\}$, but the computational complexity increases very fast. Best results are obtained for $k = 2$ or $k = 3$.

It is easy to see that the variance of even the simplest Deák estimator is less than the variance of the crude Monte Carlo method, for a given sample size N.

We remark here that the recent paper Teng et al. (2015) on spherical Monte Carlo simulations for multivariate normal probabilities provides various related simulation schemes.

6.3 Szántai's method

The procedure was first published in Hungarian, see Szántai (1976) and Szántai (1985). In English it was first published in Szántai (1988) and it is quoted in Sects. 6.5 and 6.6 of the book Prékopa (1995).

This procedure can be applied to any multivariate probability distribution function. The only condition is that we have to be able to calculate the one- and the two-dimensional marginal probability distribution function values. Accuracy can easily be controlled by changing the sample size. This way we can construct gradient estimates satisfying Assumption 3. As we have

$$F(z_1, \ldots, z_n) = \mathrm{P}\left(\xi_1 < z_1, \ldots, \xi_n < z_n\right) = 1 - \mathrm{P}(\overline{A}_1 \cup \cdots \cup \overline{A}_n),$$

where

$$\overline{A}_i = \{\xi_i \geq z_i\} \quad (i = 1, \ldots, n),$$

we can apply bounding and simulation results for the probability of union of events.

If μ denotes the number of those events which occur out of the events $\overline{A}_1, \overline{A}_2, \ldots, \overline{A}_n$, then the random variable

$$\nu_0 = \begin{cases} 0, & \text{if} \quad \mu = 0 \\ 1, & \text{if} \quad \mu \geq 1 \end{cases}$$

obviously has expected value $\overline{P} = \mathrm{P}(\overline{A}_1 \cup \overline{A}_2 \cup \cdots \cup \overline{A}_n)$.

Further two random variables having expected value $\overline{P}$ can be defined by taking the differences between the true probability value and its second order lower and upper Boole–Bonferroni bounds. The definitions of these bounds can be found in the book Prékopa (1995).

We can estimate the expected value of these three random variables in the same Monte Carlo simulation procedure and so we get three different estimates for the probability value $\overline{P}$. If we estimate the pairwise covariances of these estimates it will be easy to get a final, minimal variance estimate, too. This technique is well known as regression in the simulation literature.

Gassmann (1988) combined Szántai's general algorithm and Deák's algorithm into a hybrid algorithm. The efficiency of this algorithm was explored in Deák et al. (2002).

One can use higher than second order Boole–Bonferroni bounds, too. It will further reduce the variance of the final estimation. However, the necessary CPU time increases, which may reduce the overall efficiency of the resulting estimation. Many new bounds for the probability of the union of events have been developed in the last two decades. These bounds use not only the aggregated information of the first few binomial moments but they also use the individual product event probabilities which sum up the binomial moments. The most important results of this type can be found in the papers by Hunter (1976), Worsley (1982), Tomescu (1986), Prékopa et al. (1995), Bukszár and Prékopa (2000), Bukszár and Szántai (1999), Boros and Veneziani (2002) and Mádi-Nagy and Prékopa (2004). Szántai showed in his paper Szántai (2000), that the efficiency of his variance reduction technique can be improved significantly if one uses some of the above listed bounds.

6.4 The method of Ambartzumian, Der Kiureghian, Ohanian and Sukiasian

Ambartzumian et al. (1998) proposed to use the Sequential Conditioned Importance Sampling (**SCIS**) algorithm for the estimation of the cumulative distribution function values of a multivariate normal distribution. This is a variance reduction algorithm which is especially effective in the case of estimating extremely small probability values. This algorithm is based on the Sequential Conditioned Sampling (**SCS**) technique which is the following.

If the random vector $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_n)^T$ is normally distributed with mean vector $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_n)^T$ and positive definite covariance matrix $\mathbf{C}$ with elements $c_{ij}, i, j = 1, \ldots, n$, then its probability density function is given by

$$f(x_1, \ldots, x_n) = \frac{1}{(2\,\pi)^{\frac{n}{2}} \mid \mathbf{C} \mid^{\frac{1}{2}}} \exp\left[-\frac{1}{2}\,(\mathbf{x}-\boldsymbol{\mu})^T\,\mathbf{C}^{-1}\,(\mathbf{x}-\boldsymbol{\mu})\right]$$
$$= \frac{1}{(2\,\pi)^{\frac{n}{2}} \mid \mathbf{D} \mid^{-\frac{1}{2}}} \exp\left[-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}\,(x_i-\mu_i)\left(x_j-\mu_j\right)\right],$$

where d_{ij} are the elements of the inverse matrix $\mathbf{D} = \mathbf{C}^{-1}$. The **SCS** technique consists of generating first a random number according to the one dimensional marginal probability distribution of the random variable ξ_1 and then sequentially generating of random numbers according to the one dimensional probability density functions $\varphi_k\,(x_k \mid x_1, \ldots, x_{k-1})$ which are the one dimensional conditional probability density functions of ξ_k for given $\xi_1 = x_1, \ldots, \xi_{k-1} = x_{k-1}$. It is known that these are one dimensional normal distributions with mean

$$\mu_k(x_1, \ldots, x_{k-1}) = \mu_k - \sum_{i=1}^{k-1} d_{ki}\,\frac{x_i-\mu_i}{d_{kk}},$$

and variance

$$v_k = \frac{1}{d_{kk}},$$

for $k = 2, \dots, n$.

It is known that the crude Monte Carlo method for estimating very small multivariate normal probability distribution function values is less effective. However, Ambartzumian et al. (1998) proved that in such cases the **SCS** technique can be easily extended into **SCIS** by using an importance sampling density function (practically a truncated univariate normal density function) at each step.

6.5 Application of the numerical integration and the variance reduction Monte Carlo simulation algorithms in our procedures for probability maximization

In the course of the procedures proposed in this paper, we many times need to obtain a fixed size confidence interval for our probability distribution function value estimations. This is pronounced in Assumption 3; we need this for determining gradient estimates fulfilling the inequality given in (20) and we do this when constructing the fixed size multidimensional confidence interval described in (61). All this can be done by applying the results of Stein (1945). This is a two-stage sampling procedure. In the first stage we take a sample of size n_1 where n_1 is a positive integer not smaller than 2, otherwise it is arbitrary. Then in a second stage we take a sample of n_2 elements where n_2 is computed on the basis of the result of the first stage sampling. This way the total sample of size $n_1 + n_2$ results in a fixed size interval of the required confidence level. For a summary, see Section 7.10 of the book Prékopa (1995).

We believe that the above described two stage sampling technique can be realized on the variance reduction Monte Carlo simulation algorithms of Sects. 6.2, 6.3 and 6.4 more easily than on the numerical integration algorithm of Sect. 6.1. Careful numerical testing is necessary to choose the most appropriate procedure which may be different in the different phases of our optimization procedure.

7 A computational experiment

The aim of this experiment is to demonstrate the workability of the randomized column generation scheme of Sect. 3, in case of probabilistic problems. Namely, we have $\phi(z) = -\log F(z)$ with a nondegenerate n-dimensional standard normal distribution function $F(z)$.

7.1 Cash matching problem

Like in the previous paper (Fábián et al. 2018), we tested our implementation on a cash matching problem, with a fifteen dimensional normal distribution. In this problem we are interested in investing a certain amount of cash on behalf of a pension fund that needs to make certain payments over the coming 15 years of time. This problem originates from Dentcheva et al. (2004) and Henrion (2004). The cash matching test problem had originally been formulated as cost minimization under a probabilistic constraint. We transformed the problem to probability maximization under a cost constraint.

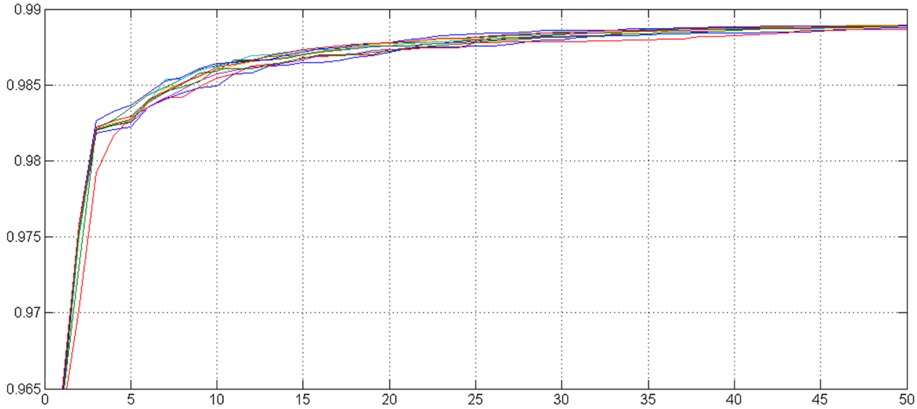

Fig. 2 Probability levels obtained, as a function of iteration counts. Different runs are represented by different threads

7.2 Implementation

We used MATLAB with the IBM ILOG CPLEX (Version 12.6.3) optimization toolbox and the numerical computation of multivariate normal distribution values was performed with the QSIMVNV Matlab function implemented by Genz (1992).

Our solver is based on the implementation used in our former paper Fábián et al. (2018). In the present version we used the randomized procedure of Sect. 3. We implemented the bounding method of Sect. 5, with the hybrid form of Sect. 5.2.

The initial solution was set by the procedure described in Fábián et al. (2018). The time needed for setting the initial solution was negligible as compared to the time needed for a single iteration with the column generation scheme.

In the course of the randomized column generation scheme, we perform just a single line search in each column generation subproblem. This line search starts from the current $\overline{z}$ vector. Gradients of the form $\nabla\phi(\overline{z}) - \overline{\boldsymbol{u}}$ need to be estimated, as mentioned in Remark 12. This goes back to the estimation of the gradient $\nabla F(\overline{z})$ of the distribution function. A component of $\nabla F(\overline{z})$ is, in turn, obtained according to (69).

Accuracy in Genz's subroutine is controlled by setting the sample size. In the present simple implementation of the iterative scheme, we control accuracy in such a way that the norm of the error of the current gradient $\nabla\phi(\overline{z}) - \overline{\boldsymbol{u}}$ be less than one tenth of the norm of the previous gradient $\|\nabla\phi(\overline{z}_{-}) - \overline{\boldsymbol{u}}_{-}\|$.

7.3 Results and observations

We performed 10 runs of the randomized procedure, each with 50 iterations. The sequences of the probability levels obtained, i.e., of the values $F(\overline{z})$, are shown in Fig. 2. At each iteration, the gradient $\nabla\phi(\overline{z}) - \overline{\boldsymbol{u}}$ is estimated by $\overline{\boldsymbol{G}} - \overline{\boldsymbol{u}}$. The norm of this estimate decreases as the procedure progresses. For a single typical run, this decrease is shown in Fig. 3.

We applied no stopping condition besides iteration count. After 50 iterations, optimal probability levels obtained in the different runs were already very near to each other (the difference between highest and lowest being less than 0.0003.) On the other hand, the value of the bound $\overline{\mathcal{B}}$ of (66) was between 0.025 and 0.03 at the end of our runs. We conclude that,

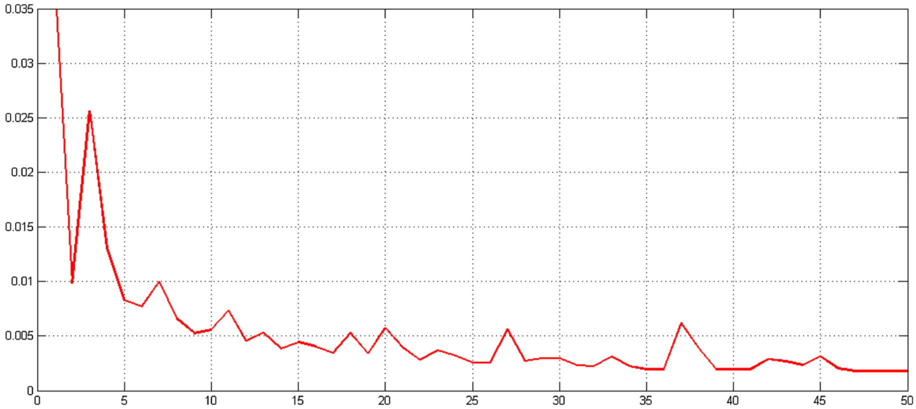

Fig. 3 Decrease of the gradient norm as a function of iteration counts, in a single run

though the bounding procedure is workable, it needs further technical improvements to keep pace with the stochastic approximation scheme.

In accordance with the hybrid bounding form of Sect. 5.2, we did not restrict new columns z_i to the box $\mathcal{Z}$. Still, the probability level was high in all iterates, $F(z_i) \geq 0.9$ holding with the columns added in the course of the column generation process. This allowed high-accuracy computation of all probabilistic function values. As mentioned in Sect. 5.2, the restriction $z' \in \mathcal{Z}$ of (59) was never active in any optimal solution $z' = \overline{z}$ of the master problem.

Density function values occurring in the computation of partial derivatives (69) have always been significant. From the 15 density function values occurring in a single gradient computation, two were always around the magnitude of 10^{-2}, another one around $5 * 10^{-3}$, and the rest around 10^{-3}. In other problems, near-zero density function values may occur in (69) for many partial derivatives. For such components, the corresponding conditional distribution function need not be computed.

Our present, very simple implementation took about 2 min to perform 50 iterations on the cash-matching problem. Though it may seem long, we expect that technical improvements will substantially shorten solution times. (According to our experience, technical improvements may result in a speedup of one or two magnitudes.)

8 Conclusion and discussion

In this paper, we proposed a stochastic approximation procedure to minimize a function whose gradient estimation is taxing. In course of the process, we build an inner approximating model of the objective function. To handle a difficult constraint function, we proposed a Newton-like scheme, employing a parametric form of the stochastic minimization procedure. The scheme enables the regulation of accuracy and reliability in a coordinated manner.

We adapted this approach to probabilistic problems. In comparison with the outer approximation approach widely used in probabilistic programming, we mention that the latter is difficult to implement due to noise in gradient computation. The outer approximation approach applies a direct cutting-plane method. Even a fairly accurate gradient may result in a cut cutting into the epigraph (especially in regions farther away from the current iterate). One either needs sophisticated tolerance handling to avoid cutting into the epigraph—see, e.g., Szántai (1988), Mayer (1998), Arnold et al. (2014),—or else one needs a sophisticated convex optimization method that can handle cuts cutting into the epigraph—see, e.g., de Oliveira et al. (2011), van Ackooij and Sagastizábal (2014). Yet another alternative is perpetual adjustment of existing cuts to information revealed in the course of the process; see Higle and Sen (1996).

Inner approximation of the level set $\mathcal{L}(F, p) = \{ z \mid F(z) \geq p \}$, an approach initiated by Prékopa (1990), results in a model that is easy to validate. The level set is approximated by means of p-efficient points. In the cone generation approach initiated by Dentcheva et al. (2000), new approximation points are found by minimization over $\mathcal{L}(F, p)$. As this entails a substantial computational effort, the master part of the decomposition framework should succeed with as few p-efficient points as possible. This calls for specialized solution methods like those of Dentcheva et al. (2004), Dentcheva and Martinez (2013), van Ackooij et al. (2017). An increasing level of complexity is noticeable.

In this paper we apply inner approximation of the epigraph of the probabilistic function $\phi(z) = -\log F(z)$. This approach endures noise in gradient computation without any special effort. Noisy gradient estimates may yield iterates that do not improve much on our current model. But we retain a true inner approximation of the function, provided function values are evaluated with appropriate accuracy. This inherent stability of the model enables the application of randomized methods of simple structure.

For probability maximization, we propose a stochastic approximation procedure with relatively easy generation of new test points. A probabilistic constraint function is handled in a Newton-like scheme, approximately solving a short sequence of probability maximization problems, with increasing accuracy. As this scheme is built from randomized components, we provide a statistical analysis of its validity.

The proposed stochastic approximation procedure can be implemented using standard components. The master problem is conveniently solved by an off-the-shelf solver. New approximation points are found through simple line search whose direction can be determined by standard implementations of classic Monte Carlo simulation procedures. The Newton-like scheme can be implemented through minor variations on a standard Newton method.

In case of a probabilistic function derived from a multivariate standard normal distribution, computing a single non-zero component of a gradient vector will involve an effort comparable to that of computing a function value. The variance reduction Monte Carlo simulation procedures described in Sect. 6 were successfully applied in outer approximation approaches to the solution of jointly probabilistic constrained stochastic programming problems, see Szántai (1988). We trust that they will perform as well in the inner approximation approach discussed in the present paper. An elaborate implementation and a systematic computational study will be needed to verify this. We mention that a means of alleviating the difficulty of gradient computation in case of multivariate normal distribution has recently been proposed by Hantoute et al. (2018).

Emerging applications of probabilistic programming afford room for different solution approaches; e.g., new models of electricity markets or traffic control, brought about by novel infocommunication technologies.

Acknowledgements Open access funding provided by John von Neumann University (NJE). We are grateful to the Editors and the anonymous Reviewers for insightful and constructive comments that led to substantial improvements of the paper.

References

Ambartzumian, R., Der Kiureghian, A., Ohanian, V., & Sukiasian, H. (1998). Multinormal probability by sequential conditioned importance sampling: Theory and applications. *Probabilistic Engineering Mechanics*, *13*, 299–308.

Arnold, T., Henrion, R., Möller, A., & Vigerske, S. (2014). A mixed-integer stochastic nonlinear optimization problem with joint probabilistic constraints. *Pacific Journal of Optimization*, *10*, 5–20.

Benveniste, A., Métivier, M., & Priouret, P. (1993). *Adaptive algorithms and stochastic approximations*. New York: Springer.

Birge, J., & Louveaux, F. (1997). *Introduction to stochastic programming*. New York: Springer.

Boros, E., & Veneziani, P. (2002). *Bounds of degree 3 for the probability of the union of events*. Technical report, Rutgers Center for Operations Research, RUTCOR Research Report 3-2002.

Bukszár, J., Prékopa, A. (2000). *Probability bounds with cherry-trees*. Technical report, Rutgers Center for Operations Research, RUTCOR Research Report 44-2000.

Bukszár, J., & Szántai, T. (1999). Probability bounds given by hyper-cherry-trees. *Alkalmazott Matematikai Lapok*, *2*, 69–85. **(in Hungarian)** .

de Oliveira, W., & Sagastizábal, C. (2014). Level bundle methods for oracles with on-demand accuracy. *Optimization Methods and Software*, *29*, 1180–1209.

de Oliveira, W., Sagastizábal, C., & Scheimberg, S. (2011). Inexact bundle methods for two-stage stochastic programming. *SIAM Journal on Optimization*, *21*, 517–544.

Deák, I. (1980). Three digit accurate multiple normal probabilities. *Numerische Mathematik*, *35*, 369–380.

Deák, I. (1986). Computing probabilities of rectangles in case of multinormal distributions. *Journal of Statistical Computation and Simulation*, *26*, 101–114.

Deák, I., Gassmann, H., & Szántai, T. (2002). Computing multivariate normal probabilities: A new look. *Journal of Statistical Computation and Simulation*, *11*, 920–949.

Dentcheva, D., Lai, B., & Ruszczyński, A. (2004). Dual methods for probabilistic optimization problems. *Mathematical Methods of Operations Research*, *60*, 331–346.

Dentcheva, D., & Martinez, G. (2013). Regularization methods for optimization problems with probabilistic constraints. *Mathematical Programming*, *138*, 223–251.

Dentcheva, D., Prékopa, A., & Ruszczyński, A. (2000). Concavity and efficient points of discrete distributions in probabilistic programming. *Mathematical Programming*, *89*, 55–77.

Ermoliev, Y. (1969). On the stochastic quasigradient method and stochastic quasi-Feyer sequences. *Cybernetics*, *5*, 208–220.

Ermoliev, Y. (1983). Stochastic quasigradient methods and their application to system optimization. *Stochastics*, *9*, 1–36.

Fábián, C., & Szántai, T. (2017). *A randomized method for smooth convexminimization, motivated by probability maximization*. Technical report, OptimizationOnline, March 2017.

Fábián, C., Csizmás, E., Drenyovszki, R., van Ackooij, W., Vajnai, T., Kovács, L., et al. (2018). Probability maximization by inner approximation. *Acta Polytechnica Hungarica*, *15*, 105–125.

Fábián, C., Eretnek, K., & Papp, O. (2015). A regularized simplex method. *Central European Journal of Operations Research*, *23*, 877–898.

Frangioni, A. (2018). *Standard bundle methods: Untrusted models and duality*. Technical reports, Department of Informatics, University of Pisa, Italy. http://eprints.adm.unipi.it/2378/1/StandardBundle.pdf. Accessed August 26, 2018

Frangioni, A. (2002). Generalized bundle methods. *SIAM Journal on Optimization*, *13*, 117–156.

Gaivoronski, A. (1978). Nonstationary stochastic programming problems. *Kybernetika*, *4*, 89–92.

Gassmann, H. (1988). Conditional probability and conditional expectation of a random vector. In Y. Ermoliev & R. B. Wets (Eds.), *Numerical techniques for stochastic optimization* (pp. 237–254). Berlin: Springer.

Genz, A. (1992). Numerical computation of multivariate normal probabilities. *Journal of Computational and Graphical Statistics*, *1*, 141–150.

Hantoute, A., Henrion, R., Pérez-Aros, P. (2018). Subdifferential characterization of probability functions under Gaussian distribution. *Mathematical Programming*. https://doi.org/10.1007/s10107-018-1237-9

Henrion, R. (2004). *Introduction to chance constraint programming*. Technical report, Weierstrass-Institut für Angewandte Analysis und Stochastik. www.wias-berlin.de/people/henrion/ccp.ps

Higle, J., Sen, S. (1996). Stochastic decomposition: A statistical method for large scale stochastic linear programming. In: *Nonconvex optimization and its applications* vol. 8. Springer.

Hunter, D. (1976). Bounds for the probability of a union. *Journal of Applied Probbility*, *13*, 597–603.

Impagliazzo, R., & Kabanets, V. (2010). Constructive proofs of concentration bounds. In: M. Serna, R. Shaltiel, K. Jansen, J. Rolim (Eds) *Approximation, randomization, and combinatorial optimization. Algorithms and techniques, RANDOM 2010, APPROX 2010. Lecture Notes in Computer Science* vol. 6302 (pp. 617–631). Berlin: Springer.

Lan, G. (2012). An optimal method for stochastic composite optimization. *Mathematical Programming*, *133*, 365–397.

Lemaréchal, C., Nemirovski, A., & Nesterov, Y. (1995). New variants of bundle methods. *Mathematical Programming*, *69*, 111–147.

Luedtke, J., Ahmed, S., & Nemhauser, G. (2010). An integer programming approach for linear programs with probabilistic constraints. *Mathematical Programming*, *122*, 247–272.

Luenberger, D., Ye, Y. (2008). Linear and nonlinear programming. In *International series in operations research and management science*. Springer.

Mádi-Nagy, G., & Prékopa, A. (2004). On multivariate discrete moment problems and their applications to bounding expectations and probabilities. *Mathematics of Operations Research*, *29*, 229–258.

Mayer, J. (1998). *Stochastic linear programming algorithms: A comparison based on a model management system*. Philadelphia: Gordon and Breach Science Publishers.

Nemirovski, A., Yudin, D. (1978). On Cezari's convergence of the steepest descent method for approximating saddle point of convex-concave functions. *Soviet Mathematics Doklady*, *19*.

Nemirovski, A., Juditsky, A., Lan, G., & Shapiro, A. (2009). Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, *19*, 1574–1609.

Nemirovski, A., & Yudin, D. (1983). *Problem complexity and method efficiency in optimization, Wiley-interscience series in discrete mathematics* (Vol. 15). New York: Wiley.

Nesterov, Y. (1983). A method for unconstrained convex minimization with the rate of convergence of $o(1/k^2)$. *Doklady AN SSSR*, *269*, 543–547.

Nesterov, Y. (2009). Primal-dual subgradient methods for convex problems. *Mathematical Programming*, *120*, 221–259.

Nesterov, Y., & Vial, J. P. (2008). Confidence level solutions for stochastic programming. *Automatica*, *44*, 1559–1568.

Panconesi, A., & Srinivasan, A. (1997). Randomized distributed edge coloring via an extension of the Chernoff–Hoeffding bounds. *SIAM Journal on Computing*, *26*, 350–368.

Pflug, G. (1988). Stepsize rules, stopping times and their implementation in stochastic quasigradient algorithms. In Y. Ermoliev & R. Wets (Eds.), *Numerical techniques for stochastic optimization* (pp. 353–372). Berlin: Springer.

Pflug, G. (1996). *Optimization of stochastic models. The interface between simulation and optimization*. Boston: Kluwer.

Polyak, B. (1990). New stochastic approximation type procedures. *Automat i Telemekh*, *7*, 98–107.

Polyak, B., & Juditsky, A. (1992). Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, *30*, 838–855.

Prékopa, A., Vizvári, B., Regős, G. (1995). *Lower and upper bounds on probabilities of Boolean functions of events*. Technical report, Rutgers Center for Operations Research, RUTCOR Research Report 36-95.

Prékopa, A. (1990). Dual method for a one-stage stochastic programming problem with random RHS obeying a discrete probability distribution. *ZOR: Methods and Models of Operations Research*, *34*, 441–461.

Prékopa, A. (1995). *Stochastic programming*. Dordrecht: Kluwer Academic Publishers.

Prékopa, A., Vizvári, B., & Badics, T. (1998). Programming under probabilistic constraint with discrete random variable. In F. Giannesi, T. Rapcsák, & S. Komlósi (Eds.), *New trends in mathematical programming* (pp. 235–255). Dordrecht: Kluwer.

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, *22*, 400–407.

Rockafellar, R. (1970). *Convex analysis*. Princeton: Princeton University Press.

Ruszczyński, A., Syski, W. (1986). A method of aggregate stochastic subgradients with on-line stepsize rules for convex stochastic programming problems. In: Prékopa A, Wets R (eds) *Stochastic programming 84 Part II, Mathematical Programming Studies* (vol. 28, pp. 113–131) Berlin: Springer.

Ruszczyński, A. (2006). *Nonlinear optmization*. Princeton: Princeton University Press.

Stein, C. (1945). A two-sample test for a linear hypothesis whose power is indpendent of the variance. *Annals of Mathematical Statistics*, *16*, 243–258.

Szántai, T. (1985). *Numerical evaluation of probabilities concerning multidimensional probability distributions*. Thesis, Hungarian Academy of Sciences, Budapest.

Szántai, T. (1976). A procedure for determination of the multivariate normal probability distribution function and its gradient values. *Alkalmazott Matematikai Lapok*, *2*, 27–39. **(in Hungarian)** .

Szántai, T. (1988). A computer code for solution of probabilistic-constrained stochastic programming problems. In Y. Ermoliev & R. B. Wets (Eds.), *Numerical techniques for stochastic optimization* (pp. 229–235). Berlin: Springer.

Szántai, T. (2000). Improved bounds and simulation procedures on the value of the multivariate normal probability distribution function. *Annals of Operations Research*, *100*, 85–101.

Szász, P. (1951). *Elements of differential and integral calculus*. Budapest: Közoktatásügyi Kiadóvállalat **(in Hungarian)**.

Teng, H. W., Kang, M. H., & Fuh, C. D. (2015). On spherical Monte Carlo simulations for multivariate normal probabilities. *Advances in Applied Probability*, *47*, 817–836.

Tomescu, I. (1986). Hypertrees and Bonferroni inequalities. *Journal of Combinatorial Theory, Series B*, *41*, 209–217.

Uryasev, S. (1988). Adaptive stochastic quasigradient methods. In Y. Ermoliev & R. Wets (Eds.), *Numerical techniques for stochastic optimization* (pp. 373–384). Berlin: Springer.

van Ackooij, W., Berge, V., de Oliveira, W., & Sagastizábal, C. (2017). Probabilistic optimization via approximate p-efficient points and bundle methods. *Computers & Operations Research*, *77*, 177–193.

van Ackooij, W., & Sagastizábal, C. (2014). Constrained bundle methods for upper inexact oracles with application to joint chance constrained energy problems. *SIAM Journal on Optimization*, *24*, 733–765.

Worsley, K. (1982). An improved Bonferroni inequality and applications. *Biometrika*, *69*, 297–302.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Affiliations

Csaba I. Fábián[1] · **Edit Csizmás**[1] · **Rajmund Drenyovszki**[1] · **Tibor Vajnai**[1] · **Lóránt Kovács**[1] · **Tamás Szántai**[2]

Edit Csizmás
csizmas.edit@gamf.uni-neumann.hu

Rajmund Drenyovszki
drenyovszki.rajmund@gamf.uni-neumann.hu

Tibor Vajnai
vajnai.tibor@gamf.uni-neumann.hu

Lóránt Kovács
kovacs.lorant@gamf.uni-neumann.hu

Tamás Szántai
szantai@math.bme.hu

[1] Department of Informatics, GAMF, Faculty of Engineering and Computer Science, John von Neumann University, Izsáki út 10, Kecskemét 6000, Hungary

[2] Department of Differential Equations, Institute of Mathematics, Budapest University of Technology and Economics, Műegyetem rakpart 3-9, Budapest 1111, Hungary