Volumetric Center Method for Stochastic Convex Programs using Sampling

Sanjay Mehrotra [1]

IE/MS Technical Report, December 1999 (minor revision January 2000)
Department of Industrial Engineering and Management Sciences,
Robert R. McCormick School of Engineering,
Northwestern University, Evanston, Illinois 60208
mehrotra@iems.nwu.edu

Abstract

We develop an algorithm for solving the stochastic convex program (SCP) by combining Vaidya's volumetric center interior point method (VCM) for solving non-smooth convex programming problems with the Monte-Carlo sampling technique to compute a subgradient. A near-central cut variant of VCM is used, and for this method an approach to perform bulk cut translation, and adding multiple cuts is given. We show that by using near-central VCM the SCP can be solved to a desirable accuracy with any given probability. For the two-stage SCP the solution time is independent of the number of scenarios.

Key Words: Stochastic Programming, Volumetric Center, Analytic Center, Interior Point Methods, Convex Programming, Volumetric Center

[1] The author thanks the Supreme Lord for all revelations that lead to this paper. All its fruits are surrendered to Him.

1 Introduction

In this paper we develop an algorithm for solving the general stochastic convex problem [73, 15] (SCP):

$$\begin{array}{ll} \min & c^0(x) \equiv E[r^0(x,\tilde{\xi})] \\ \text{s.t.} & c^i(x) \equiv E[r^i(x,\tilde{\xi})] \leq 0, i = 1, \ldots m, \\ & x \in X \subseteq \Re^n, \end{array}$$

where $\tilde{\xi}$ is a random vector defined on the probability space $(\Xi, \mathcal{F}, P)$. $\mathcal{F}$ is a $\sigma-$algebra of subsets of Ξ, and P is a probability measure on $\mathcal{F}$. The set X is a compact convex set, and we assume that it is given explicitly by a set of deterministic convex inequality constraints whose subgradient can be calculated. Additional assumptions are made at appropriate places. A particular realization of $\tilde{\xi}$ is represented by ξ. The functions $r^i : \Re^n \times \Xi \rightarrow \Re$, $i = 0, \ldots m$ are proper normal convex integrands, i.e., $r^i(\cdot, \xi)$ is proper and the epigraph of $r^i(\cdot, \xi)$: $\{(x, \alpha) | r^i(x, \xi) \leq \alpha, x \in \Re^n, \alpha \in \Re\}$ is closed, measurable in ξ and convex. Moreover, we assume that for any $\xi \in \Xi$ $-\infty < r^i(x, \xi) < \infty$, i.e., $r^i(x, \xi)$ are finite valued for $i = 0, \ldots m$. The expected value function is given by

$$E[r^i(x,\tilde{\xi})] \equiv \int_{\Xi} r^i(x,\xi) P(d\xi)$$

for $i = 0, \ldots m$, and it is also finite. We are interested in problems for which a subgradient of $E[\cdot]$ can be computed either exactly or stochastically. In an important class of two-stage SCP we will show how this can be accomplished. We will study this problem in more details. Next we briefly summarize the developments for solving SCP and background for the work presented in this paper to give it a context.

1.1 Background Review

The SCP has been studied extensively since it's linear case was first introduced by Dantzig [23] and Beale [10] in 1955. Studies in the 60's focused primarily on the linear stochastic program [43, 44, 75, 69]. These years also saw simultaneous development of the theory of subdifferention and integration of convex functions [51, 39, 49, 52]. Subsequent to this development, in the seventies, Rockafellar and Wets [54, 55, 53, 56] developed extended duality theory for SCP and gave conditions under which SCP is well defined. Hiriart-Urruty [34] and Rockafellar and Wets [56] studied the properties of the mean value function $E[r(\cdot)]$ and its subdifferential set, $\partial E[r(\cdot)]$. These works and several additional theoretical properties of SCP and general stochastic programming problem together with several applications are well surveyed in Wets [73]. A comprehensive reference list of books and collections of papers on Stochastic Programming appear in [74].

It is easy to see that in general $c^i(x)$ are (non-smooth) convex functions [73, Proposition 2.1]. Hence methods for non-smooth convex optimization are immediately applicable for solving SCP provided that an exact subgradient of $c^i(\cdot)$ can be computed. This is well recognized in the case of two-stage stochastic linear programming problems with finite number of scenarios (SLPF). For this problem several algorithms are designed that directly or indirectly use the subgradient information [69, 7, 17, 60]. Lemma 1 gives a way to compute a subgradient of $c^i(x)$ for the two stage SCP. As a result all known methods for solving non-smooth convex programs become available to solve this problem. The use of non-smooth techniques have received greater attention for handling coupling constraints in the multi-stage SCP (for example see Chun and Robinson [20]) or stabilizing traditional cutting plane methods (for example see Ruszczyński [60]). The interested reader can find methods for non-smooth convex optimization in the literature. Here we mention main methods and some recent references.

Methods for non-smooth convex optimization can be broadly classified in the following six categories: (i) subgradient methods, (ii) ellipsoid method, (iii) classical cutting plane methods, (iv) bundle methods, (v) proximal point methods and (vi) volumetric and analytic center (interior-point) methods. Subgradient and ellipsoid method are described in Shor [67]. An excellent survey of ellipsoid method is by Bland, Goldfarb and Todd [18]. Zangwill [77] gives a unified treatment of classical cutting plane methods. The book by Hiriart-Urruty and Lemaréchal [35] is a comprehensive source for bundle methods. For more recent developments on proximal point and bundle methods see Mifflin [45], Birge, Qi and Wei [17], and Güler [28]. Convergence results of analytic and volumetric center cutting plane methods are more recent. For development of methods based on analytic center recent references are Andersen, Mitchell, Roos and Terlaky [2], Goffin, Luo and Ye [29] and Nesterov and Vial [47]. A good source for developments on volumetric center method of Vaidya [68] is Anstreicher [5].

For SLPF the non-smooth convex optimization approach has lead to the development of cutting plane algorithms using decomposition. These include the widely used L-shaped method of Van Slyke and Wets [69], which can also be seen as an application of Bender's decomposition method. Regularization of the L-shaped method using ideas of the bundle method for non-smooth convex optimization have been suggested and implemented (see for example, Ruszczyński [59, 60]). Ariyawansa and Jiang [7] have given algorithms for SLPF based on the ellipsoid method, Vaidya's [68] volumetric center method, and the analytic center method [29]. In particular, they have shown that the complexity of volumetric center method grows only linearly with the number of scenarios, K.

In addition to methods based on non-smooth convex programming, several additional approaches have been proposed to solve SLPF. The SLPF problem can be formulated as a large linear program [71], and methods for large scale linear programming are considered (Wets [72]). Although the initial attempts were to specialize simplex method to exploit the structure of the problem[72], more recently primal-dual interior point methods have been applied (Carpenter, Lustig and Mulvey [19]) to the deterministic equivalent. The interior point methods have been found to be more efficient than the simplex based methods [19]. An interior point method was also analyzed by Birge and Qi [16], showing that the block-angular structure of SLPF can be exploited to get an interior point algorithm for the deterministic equivalent whose complexity grows as $O(K^{1.5})$. In practice the computational time for primal-dual interior point method to solve the deterministic equivalent grows only linearly in the number of scenarios (Czyzyk, Fourer and Mehrotra [21]). Parallel implementations show a near linear reduction in computing time with the number of processors (Czyzyk, Fourer and Mehrotra [22] and Yang and Zenios [76]).

An important property of non-smooth methods using subgradient calculation is that computation of subgradient for SCP decomposes in scenarios. A subgradient can be computed by solving a linear (convex) program for each of the scenario. This is important because it allows for subgradient computation in a distributed computing environment, where individual processing nodes may be unreliable. We note that decomposition is also possible in interior point methods that solve the deterministic equivalent, however, here the decomposition is in matrix factorization.

Scenarios in stochastic programs are generated as an approximation to some underlying distribution. The number of scenarios quickly get very large even when the distribution of each random data element is determined by just a small number of discrete points. For example, with 80 random data elements with each taking 3 possible values we get approximately 10^{38} scenarios (Infanger [38, Section 2.3]). Problems of such size can not be handled by deterministic decomposition algorithms. This has lead to the development of stochastic subgradient and decomposition algorithms. For a discussion on stochastic subgradient methods see Ermoliev [27], Ruszczyński and Syski [61], and Au, Higle and Sen [9]. Stochastic decomposition algorithms embed sampling into the cutting plane methods. There are two such approaches. First approach is based on using large samples to compute 'accurate' subgradients which are used to generate cuts. Dantzig and Glenn [24], Dantzig and Infanger [25] and Infanger [38] give such an algorithm based on the L-shaped method of Van Slyke and Wets [69]. The other approach is based on using samples whose size grow as the algorithm progress. Algorithms based on this approach are developed by Higle and Sen [30, 31, 33]. In Higel and Sen decomposition algorithms information from a new scenario is added at each iteration and previously added cuts are updated using this information progressively.

An alternative approach for problems with very large number of scenarios is to directly approximate the stochastic programs using Monte-Carlo samples. In particular, $c^i(x) \approx \frac{1}{N}\sum_{j=1}^{N} r^i(x,\xi^j)$ for $i = 0,\ldots m$, and N is the sample size. This approach is called the sample average method. For the sample average method the rate of convergence of the distance of an optimal solution of the approximate problem, $\hat{x}_N$, to an optimal solution of the true problem are given in King and Rockafellar [41] and Shapiro [62]. Under certain regularity, twice differentiability assumption on the Lagrangian function associated with SCP, and second order sufficiency condition (implies uniqueness of the optimal solution) at the optimal solution, King and Rockafellar [41] show that the rate of convergence is $O_p(N^{1/2})$, where $O_p(.)$ notation means that the bound is in probability. A similar bound is achieved by Shapiro [62] under different assumptions. For this approach using the large deviation principle Kaniovski, King and Wets [40] have shown that the probability of an event where $\|\hat{x}_N, x^*\| \geq \rho > 0, x^* \in \mathcal{C}^*$ tends to zero exponentially fast as $N \to \infty$. Here $\mathcal{C}^*$ is the set of optimal solutions. More recently, Shapiro and Homem-de-Mello [65] have shown that in the sample average method the approximate stochastic program gives an optimal solution of SCP for sufficiently large N. Using the large deviation principle Shapiro and Homem-de-Mello have shown that the probability of not finding this solution goes to zero exponentially fast with N. Kleywegt and Shapiro [**?**] and Ahmed and Shapiro [1] have developed this approach for stochastic discrete optimization. Linderoth, Shapiro and Wright [42] and Verweij, Ahmed, Kleywegt, Nemhauser and Shapiro, [70] have given empirical evidence showing that the sample average method finds an excellent approximation of the stochastic continuous and discrete optimization problems while using only a very small number of scenarios.

1.2 Contributions of this paper

The algorithm of this paper combines the 'accurate subgradient' approach with a variant of VCM. We call this variant a near-central cut VCM. This algorithm is analyzed using the large deviation principle. In the context of convex feasibility problems we show that near-central cut VCM allows for addition of multiple cuts and bulk cut translation with relative ease. The development of near-central cut VCM is motivated primarily because of its suitability for solving SCP. Using this variant we develop an algorithm for SCP that generates cuts using sampling. An important aspect of the proposed algorithm is that it gives performance bounds for finding an optimal solution of SCP with any desirable probability. This type of performance guarantee is not currently known for other cutting plane methods. The developed algorithm enjoys all the properties of a decomposition algorithm. In particular, this algorithm is naturally suitable for distributed computing environment and gives a linear speed up in subgradient computation for two-stage SCP. As a result one can find a solution of two-stage SCP with any desirable probability in 'polynomial time', possibly using exponential number of processors.

1.3 Organization of this paper

In order to motivate our subsequent development, in the next section we introduce the two stage stochastic convex program, describe its various properties and show how a subgradient can be computed for this problem. In Section 3 we develop near-central cut VCM. The VCM is designed for solving a convex feasibility problem. After an introduction and review of the VCM this section is divided into seven subsections. We summarize various properties of the volumetric barrier function and the volumetric center and some technical results in Section 3.1. We use these properties to analyze the progress in near-central cut VCM. In Section 3.2 we give a result on the progress towards computing volumetric center after a damped Newton-like step is computed. In Section 3.3 we analyze the change in the value of volumetric barrier after a cut is added. In Section 3.4 we analyze the change in the value of volumetric barrier after a constraint is dropped. This is sufficient to complete the analysis of a basic version of near-central cut VCM. We analyze this method in Section 3.5. In Sections 3.6 and 3.7 we give conditions that allow bulk constraint translation and multiple cut addition while maintaining the overall computational complexity of the algorithm.

We return to the two-stage SCP in Section 4. In this section we adapt the near-central cut VCM for solving two-stage SCP. In Section 4.1 we give an algorithm for solving two-stage SCP with finite number of scenarios using exact subgradient computation. In Section 4.2 we give an analysis for two-stage SCP, where subgradient computations are performed using sampling. Here we also discuss some practical ways of estimating the number of samples. In Section 5 we state the extension of the method for two-stage SCP to the general SCP under the assumption that subgradients can be computed in SCP. The two subsequent short sections contain concluding remarks and acknowledgements. The notation and abbreviations scattered through out the paper are summarized below.

1.4 Notation

Abbreviations: Stochastic Convex Program (SCP), Volumetric Center Method (VCM), two-stage Stochastic Linear Program with Finite number of scenarios (SLPF), Second Stage Problem (SSP), two-stage SCP (TSSCP). $k/0, k > 0$ is taken to be ∞ for any $k \geq 0$. All vectors are column vectors, and ^{T} denotes the transpose of a vector. The convex objective function is given by $c(\cdot)$ and $c^0(\cdot)$ which are used interchangeably. $\partial c(x)$ denotes the subdifferential set of $c(\cdot)$ at x. $\|x\|$ represents two-norm of a vector x and $\|x\|_Q$ represent the norm with respect to a positive definite matrix Q, i.e., $\|x\|_Q \equiv \sqrt{x^T Q x}$. $S(\hat{x}, \alpha) \equiv \{x \,|\, \|x - \hat{x}\| \leq \alpha\}$. Exp denotes the exponential function. Prob$(\cdot)$ denotes the probability of an event. $E[\cdot]$ represents the expected value of a random variable. $\det(\cdot)$ represents the determinant of a matrix. $diag(x)$ denotes a matrix whose diagonal elements are x_i. The notation $Q \preceq V$ mean that $V - Q$ is a positive definite matrix. $\mathcal{C}$ denotes a general convex set, $\mathcal{C}^1$ denotes the first stage feasible set in TSSCP, $\mathcal{C}^2(x, \xi)$ denotes the second stage feasible set for a given x and ξ. $\mathcal{C}^*$ denotes the set of optimal solutions and $\mathcal{C}_\rho$ denotes the set of $\rho-$optimal solutions. $g(\cdot)$ is used to denote the gradient of volumetric barrier in Section 3 and it represents a subgradient of an appropriate function in other sections. Additional notation is defined at appropriate places.

2 Two Stage Stochastic Convex Program

2.1 Problem Definition

The two-stage SCP (TSSCP) with recourse is described as:

$$\begin{array}{ll} \min & c^0(x) \equiv \tilde{c}(x) + E[r(x, \tilde{\xi})] \\ \text{s.t.} & x \in \mathcal{C}^1 \equiv \{x | c^i(x) \leq 0\}, \quad i = 1, \dots m, \end{array}$$

where $x \in \Re^n$, $c^i(x) : \Re^n \to \Re$, $i = 0, \dots m$ are finite valued convex functions. The variables x are called first stage decision variables. The random vector $\tilde{\xi}$ is defined on the probability space $(\Xi, \mathcal{F}, P)$ defined as in the introduction. A particular realization of $\tilde{\xi}$ is represented by ξ. The objective is to minimize the sum of first stage costs and the expected recourse costs of taking a decision. For a given $\bar{x}$ and ξ a recourse action is found by solving a second stage problem SSP($\bar{x}$,ξ), which is given as:

$$\begin{array}{lll} r(x, \xi) \equiv & \min & f^0(x, y, \xi) \\ & \text{s.t.} & y \in \mathcal{C}^2(x, \xi) \equiv \{y | f^i(x, y, \xi) \leq 0, \quad i = 1, \dots m_2, \bar{x} - x = 0\}, \end{array}$$

where $y \in \Re^{n_2}$ and for any ξ, $f^i(x, y, \xi) : \Re^{n+n_2} \to \Re, i = 0, \dots m_2$, are finite valued normal convex integrands. Variables y are second stage variables which give a recourse action taken after a value of the random parameters is realized. We associate Lagrange multipliers $\pi(\bar{x}, \xi) \in \Re_+^{m_2}$ with inequality constraints in SSP($\bar{x}$, ξ), and $a(\bar{x}, \xi) \in \Re^n$ with the equality constraints $\bar{x} - x = 0$. The reason for including '$\bar{x} - x = 0$' constraints, instead of removing x variables from SSP($\bar{x}$,ξ), becomes clear in Lemma 1 below. We discuss the possibility of explicit substitution of $x = \bar{x}$ after this lemma. An optimal solution of SSP($\bar{x}$,ξ) is denoted by $y^*(\bar{x}, \xi)$, and the corresponding optimal Lagrange multipliers are denoted by $\pi^*(\bar{x}, \xi)$ and $a^*(\bar{x}, \xi)$, respectively. Note that for convenience we have taken constraints in both first and second stage problems in equality form. The linear equality constraints can be incorporated in the first and second stage problems. However, the presence of linear equality constraints requires a modification to the description of volumetric barrier method.

2.2 Technical Assumptions

We make following additional assumptions on the problem:

A0. The set Ξ is compact.

A1. The set $\mathcal{C}^1$ is compact, and it has a non-empty interior. Furthermore, $\mathcal{C}^1 \subset \mathcal{B} \equiv \{x | x_i^l \leq x_i \leq x_i^u, i = 1, \ldots n\}$, where x_i^l and x_i^u are known. More explicitly we assume that $x_i^u = -x_i^l = 2^{\hat{L}}$, and $\mathcal{C}^1$ contains a sphere of radius 2^{-L}.

A2. The set $\mathcal{C}^2(x, \xi)$ is non-empty and bounded for all x in an open set, $\mathcal{C}_o^1$, containing $\mathcal{C}^1$.

A3. Lagrange multipliers $\pi^*(\bar{x}, \xi), a^*(\bar{x}, \xi)$ satisfying KKT conditions are available together with $y^*(\bar{x}, \xi)$ while solving SSP$(\bar{x}, \xi)$. Furthermore, $|a_i^*(\bar{x}, \xi)| \leq \nu$ for all $\bar{x}, \xi$, $i = 1, \ldots m_2$. Also, the same optimum solution and Lagrange multipliers become available if a scenario is repeatedly generated.

Assumption A0 is needed to ensure that scenarios of the second stage problem remain well defined. Assumption A1 is needed in the volumetric center method. In practice, the feasibility assumption can be ensured by introducing an artificial variable with large cost in the first and second stage problems. Also, boundedness can be ensured by introducing a large bound on the first and second stage variables. Assuming that $\mathcal{C}^1$ is bounded, the bounds for the set $\mathcal{B}$ can be obtained by solving $2n$ first stage convex optimization problems $\min_{x \in \mathcal{C}^1} x_i$ and $\max_{x \in \mathcal{C}^1} x_i$ for $i = 1, \ldots n$. The assumption that $x_i^u = -x_i^l$ can be ensured by a simple shift of origin, and $x_i^u = -x_i^l = 2^{\hat{L}}, i = 1, \ldots n$, can be ensured by a simple scaling after the origin is shifted. Assumption A2 requires that for all possible first stage decisions a recourse action is always possible. This type of assumption is common in the stochastic programming literature even for the linear and quadratic case (for example, Rockafellar and Wets [57] and Higle and Sen [31, 33]).

Assumption A3 is needed for subgradient calculations in the proposed method. An optimum solution and lagrange multipliers are available when the second stage problems are linear and quadratic programs. For more general problems we can only expect to obtain good approximations of these quantities. We will discuss in Section 4 required modifications to our algorithm when exact multipliers are not available. The boundedness of Lagrange multipliers can be ensured by putting an explict bound on their value, or assuming that $\mathcal{C}^2(x, \xi)$ has a non-empty interior. In general, this bound depends on the size of the second stage problem scenarios, and it may be exponentially large in the worst case.

2.3 Properties of the Recourse Function

From Proposition 4 in [54] we know that $r(x,\xi)$ is also a normal convex integrand. Hence, for any $x \in \mathcal{C}_o^1$, we write the expected recourse cost of taking a decision as:

$$R(x) \equiv E[r(x,\tilde{\xi})] \equiv \int_{\Xi} r(x,\xi)P(d\xi).$$

Note $R(x)$ is a convex function [73]. We call g to be a subgradient of convex function $f(x,\xi)$ at $\bar{x} \in \mathcal{C}^1$ if

$$f(\bar{x},\xi) \leq f(\hat{x},\xi) - g^T(\hat{x}-\bar{x}) \tag{1}$$

for all $\hat{x} \in \mathcal{C}^1$. Furthermore, from Rockafellar and Wets [56] and Hiriart-Urruty [34, II.4.3] ([73, Proposition 2.10]), under Assumption A2 for all $x \in \mathcal{C}^1$, $\partial R(x) = E[\partial r(x,\xi)]$. The following lemma gives a way to compute an element of (a subgradient) $\partial R(x)$.

LEMMA 1 *Let $\pi^*(\bar{x},\xi) \in \Re_+^{m_2}$ and $a^*(\bar{x},\xi) \in \Re^n$ be optimal Lagrange multipliers associated with inequality and equality constraints ($\bar{x}-x=0$) in SSP($\bar{x}$,ξ). Let $u=(x,y)$, and by $g^i(u,\xi)$ denote a subgradient vector of $f^i(\cdot,\xi)$ at u, for $i=0,\dots,m_2$.*

Then, a subgradient of $r(x,\xi)$ and $R(x)$ at $\bar{x} \in \mathcal{C}^1$ is given by $a^*(\bar{x},\xi)$ and $\int_\Xi a^*(\bar{x},\xi)P(d\xi)$, respectively. Also, if the number of scenarios is finite (say K) and ϱ^i is the probability for scenario ξ^i, $i=1,\dots K$, then a subgradient of $R(x)$ at $\bar{x}$ can be computed as $\sum_{i=1}^K \varrho^i a^*(\bar{x},\xi^i)$.

Proof. For $\bar{x},\hat{x} \in \mathcal{C}^1$, let $\bar{u}=(\bar{x},y^*(\bar{x},\xi))$ and $\hat{u}=(\hat{x},y^*(\hat{x},\xi))$. For SSP$(\bar{x},\xi)$ the optimal multipliers $\pi^*(\bar{x},\xi) \geq 0$ and $a^*(\bar{x},\xi)$ satisfy

$$g^0(\bar{u},\xi) + \sum_{i=1}^{m_2} \pi_i^*(\bar{x},\xi) g^i(\bar{u},\xi) + \begin{pmatrix} -a^*(\bar{x},\xi) \\ 0 \end{pmatrix} = 0 \tag{2}$$

for some $g^i(\bar{u},\xi) \in \partial f^i(\bar{u},\xi)$, and

$$\pi_i^*(\bar{x},\xi) f^i(\bar{u},\xi) = 0, \;\; i=1,\dots m_2. \tag{3}$$

Now,

$$
\begin{aligned}
r(\bar{x},\xi) &= f^0(\bar{u},\xi) \\
&= f^0(\bar{u},\xi) + \sum_{i=1}^{m_2} \pi_i^*(\bar{x},\xi) f^i(\bar{u},\xi) \qquad \text{(using (3))} \\
&\leq f^0(\hat{u},\xi) - g^0(\bar{u},\xi)^T(\hat{u}-\bar{u}) + \sum_{i=1}^{m_2} \pi_i^*(\bar{x},\xi)\left(f^i(\hat{u},\xi) - g^i(\bar{u},\xi)^T(\hat{u}-\bar{u})\right) \text{(using (1))} \\
&= f^0(\hat{u},\xi) + \sum_{i=1}^{m_2} \pi_i^*(\bar{x},\xi) f^i(\hat{u},\xi) - a^*(\bar{x},\xi)^T(\hat{x}-\bar{x}) \qquad \text{(using (2))} \\
&\leq f^0(\hat{u},\xi) - a^*(\bar{x},\xi)^T(\hat{x}-\bar{x}) \quad (\text{since } f^i(\hat{u},\xi) \leq 0) \\
&= r(\hat{x},\xi) - a^*(\bar{x},\xi)^T(\hat{x}-\bar{x}).
\end{aligned}
$$

This shows that $a^*(\bar{x},\xi) \in \partial r(\bar{x},\xi)$. Now to see $\int_\Xi a^*(\bar{x},\xi)P(d\xi) \in \partial R(x)$ note that

$$
\begin{aligned}
R(\bar{x}) &= \int_\Xi r(\bar{x},\xi)P(d\xi) \\
&\leq \int_\Xi (r(\hat{x},\xi) - a^*(\bar{x},\xi))^T(\hat{x}-\bar{x})P(d\xi) \\
&= \int_\Xi r(\hat{x},\xi)P(d\xi) - \int_\Xi a^*(\bar{x},\xi)^T(\hat{x}-\bar{x})P(d\xi) \\
&= \int_\Xi r(\hat{x},\xi)P(d\xi) - \left(\int_\Xi a^*(\bar{x},\xi)P(d\xi)\right)^T(\hat{x}-\bar{x}).
\end{aligned}
$$

Here the last equality follows because $a^*(\bar{x},\xi)$, a subgradient of $r(\bar{x},\xi)$, is a measurable function of ξ (Rockafellar [50, Corollary 4.6]), and for measurable vector functions the equality above holds ([36, Section 3.42]). The case where the number of scenarios is finite is just a special case. □

We note that if $f^i(x,y,\cdot)$ are differentiable convex functions, then $x = \bar{x}$ can be explicitly substituted in the second stage problem definition. The vector $a^*(\bar{x},\xi)$ is recovered from (2), by computing the gradients of $f^i(x,y,\cdot)$ at $(\bar{x}, y^*(\bar{x}), \xi)$ and using non-negative multipliers from the reduced problem. This can also be done in the non-differentiable case if there is a way to extend a subgradient of $f^i(\bar{x},y,\cdot)$ at $y^*(\bar{x},\xi)$ to a subgradient of $f^i(x,y,\cdot)$ at $(\bar{x}, y^*(\bar{x},\xi))$ while keeping the components corresponding to y variable unchanged.

Assumption A3 implies that the subgradient of $R(x)$ computed in Lemma 1 is bounded by ν component-wise.

In the case where Ξ has finitely many elements, Lemma 1 gives a way to compute an exact subgradient of $R(x)$. This means that any method for finding a solution of non-smooth convex program can be used to solve TSSCP. However, for many practical situations either the number of elements in Ξ is very large, or Ξ is continuous. In such situations we resort to Monte-Carlo sampling to estimate $R(x)$ and $\partial R(x)$ (see discussion in [38, Section 2.3]). We will study the use of Monte-Carlo sampling in Section 4.2. In this approach the calculated subgradient will be approximate. For this reason in the next section we develop a near-central cut variant of volumetric center method.

3 Volumetric Center Cutting Plane Method

The volumetric center cutting plane method of Vaidya [68] is designed for the convex feasibility problem. Assuming that a convex set $\mathcal{C}$ is contained in a hypercube $\|x\|_\infty \leq 2^{\hat{L}}$, the convex feasibility problem is to find a point in $\mathcal{C}$ or conclude that the volume of $\mathcal{C}$ is less than that of a n-dimensional sphere of radius 2^{-L} for some given $L > 0$. Unless a point in $\mathcal{C}$ is found, VCM maintains a polyhedral set containing $\mathcal{C}$.

Let $\mathcal{P} = \{x \in \Re^n | Ax \geq b\}$, where $A \in \Re^{m \times n}$, and $b \in \Re^m$. Let $s(x) = Ax - b$, and $S(x) = diag(s(x))$ be a diagonal matrix whose diagonal elements are $s_i(x)$. Let a_i^T represent the ith row of A. The volumetric barrier for $\mathcal{P}$ is the function

$$V(x) \equiv \frac{1}{2} ldet(H(x)), \quad \text{where } H(x) \equiv A^T S^{-2}(x) A = \sum_{i=1}^{m} \frac{1}{s_i^2(x)} a_i a_i^T,$$

and $ldet(\cdot) \equiv \ln(det(\cdot))$. The matrix $H(x)$ is the Hessian of the log-barrier function: $\sum_{i=1}^{m} \ln(a_i^T x - b_i)$. The volumetric barrier function is strictly convex and its unique minimizer is called the volumetric center of $\mathcal{P}$.

Vaidya's VCM method has three ingredients: (i) Newton-like steps used to reduce the value of the volumetric barrier and find an approximate volumetric center, (ii) Addition of a cut at approximate volumetric center to reduce the region of uncertainty, and (iii) Deletion of a constraint if it is no longer desirable, and it satisfies certain criterion. The volumetric center method stops with an iterate when the value of the volumetric barrier is sufficiently large. Vaidya's main result is that the complexity of his volumetric cutting plane method is $O(n(L+\hat{L})T + n^4(L+\hat{L}))$ compared to $O(n^2(L+\hat{L})T + n^4(L+\hat{L}))$ arithmetic operations for the ellipsoid method [18]. Here T is the cost of computing a cut (an oracle). Vaidya showed that his method will terminate in $O(n(L+\hat{L}))$ iterations, while each iteration requiring $O(n^3)$ floating point computation. In theory, the work at each iteration of Vaidya's method can be reduced to $O(n^{2.38})$ using fast matrix multiplications, which can not be applied to ellipsoid method. The total number of outer iterations of Vaidya's algorithm are inversely proportional to a quantity ΔV, which is the difference of the minimum increase in the value of the volumetric barrier when a cut is added and the maximum decrease in the value when a constraint is removed. Inner iterations in Vaidya's algorithm are performed using Newton-like steps. At a point near the volumetric center, the method generates a cut and backs off this cut by a significant amount. Such cuts are called shallow cuts. In addition, his analysis results in very large constants hidden in $O(\cdot)$ notation (see Anstreicher [4, Section 7] and [5, Section 1] for detailed discussion). Ramaswamy and Mitchell [48] analyze a central-cut variant of volumetric center algorithm, where the new cut is passed through the point at which it is generated, while an affine step is used to generate a new iterate to start recomputation of the volumetric center of the new polytope. In Ramaswamy and Mitchell's central cut variant analysis the order of outer iterations remain the same, while it requires $O(\sqrt{n})$ Newton-type iterations to recompute the volumetric center. The central cut variant is preferable because, instead of $O(\sqrt{n})$, in practice one expects it to take very few Newton-type iterations for recomputing the volumetric center, while the number of outer iterations is reduced by a larger factor.

Another aspect of Vaidya's algorithm is the maximum number of constraints it carries. Very careful analysis by Anstreicher [4, 5] has reduced the number from $10^7 n$ in Vaidya's analysis for the shallow cut version to $25n$ for the central cut version of the algorithm. Moreover, Anstreicher's analysis [5] has shown that ΔV in Vaidya's algorithm can be increased from 1.3×10^{-7} to 1.4×10^{-3}, a gain of more than 10^4.

The volumetric method proposed and analysed in this section is a method in which the amount by which we back off a cut is much smaller. We call this a near-central cut version. Our main reason for proposing this variant is our context of stochastic programming problems. As seen in the previous section, the subgradient used to generate the cut at a given point can be computed only approximately when the number of scenarios is large (infinite). By backing off we can absorb the error in subgradient computation. In particular, we increase the probability of not cutting away the optimum solution. Although the cuts in the near-central cut variant do not go through the current iterate, they are still deep. For example, the slack at the current iterate in the added cut is about the same as the slack at the iterate obtained by moving along the affine direction in the central cut version of the algorithm analyzed by Anstreicher [5]. An important aspect of the near-central version of the algorithm is that it naturally allows for addition of multiple cuts, a feature that seems difficult to get for the central cut version (see Anstreicher [5, Conclusions]).

We now study these aspects of near-central cut variant of VCM. The next section collects several known results obtained by Vaidya[68] and Anstreicher[3, 4, 5]. The first time reader may jump ahead to Section 3.5 and then return to next four subsections.

3.1 Properties of the Volumetric Barrier

The volumetric barrier function $V(x)$ is a strictly convex function, and we denote its unique minimizer by w. Let x be such that $s(x) > 0$,

$$P(x) \equiv S^{-1}(x)A(A^TS^{-2}(x)A)^{-1}A^TS^{-1}(x)$$

and

$$\sigma_i(x) \equiv \frac{a_i^T(A^TS^{-2}(x)A)^{-1}a_i}{s_i^2(x)} = \frac{a_i^TH(x)^{-1}a_i}{s_i^2(x)}, \quad i = 1, \dots m,$$

where ${a_i}^T$ is the ith row of A. Let $D(x) = diag(\sigma(x))$. The gradient and Hessian of $V(\cdot)$ at x are given by (see Anstreicher [3, Lemma A.2,A.3] or Vaidya [68, Lemma 1,2])

$$g(x) \equiv \nabla V(x) = -A^TS^{-1}(x)\sigma(x) = -\sum_{i=1}^{m} \frac{\sigma_i(x)}{s_i(x)}a_i,$$

$$\nabla^2 V(x) \equiv A^TS^{-1}(x)(3D(x) - 2P^{(2)}(x))S^{-1}(x)A,$$

where $P^{(2)}$ denotes the Hadamard product of P with itself, i.e., $P_{ij}^{(2)} = (P_{ij})^2$. Let

$$Q(x) \equiv A^TS^{-2}(x)D(x)A = \sum_{i=1}^{m} \frac{\sigma_i(x)}{s_i^2(x)}a_i{a_i}^T.$$

The matrix $Q(x)$ is positive definite, which gives a good approximation of $\nabla^2 V(x)$. In particular,

$$Q(x) \preceq \nabla^2 V(x) \preceq 3Q(x). \tag{4}$$

The notation $Q \preceq V$ means that $V - Q$ is a positive semi-definite matrix. The above bound is due to Anstreicher [3, Lemma A.4]. A weaker bound was proved in Vaidya [68, Lemma 3].

In order to measure progress in VCM we need to know (i) the amount of reduction in $V(\cdot)$ after a Newton-like step is taken, (ii) the change (increase) in $V(\cdot)$ after a cut is added, (iii) the change (decrease) in $V(\cdot)$ after an undesirable constraint is deleted. The difference in the $V(\cdot)$ while adding and dropping cuts measures the convergence of the algorithm. The Newton-like step analysis ensures that worst case complexity for recentering after cuts are added or dropped. For this purpose, in Anstreicher's analysis [5], the following expansion of $V(\cdot)$ plays a fundamental role. Let $x, x+p \in \mathcal{P}$, for some $p \in \Re^n$, then

$$V(\bar{x}) = V(x) + g^T(x)p + \int_0^1 \int_0^\alpha p^T\nabla^2 V(x+\beta p)p \; d\beta d\alpha. \tag{5}$$

Anstreicher showed [5] that if $\|S^{-1}(x)Ap\|_\infty \le \delta \le 1$, then

$$\frac{p^TQ(x)p}{2(1+\delta)^2} \le \int_0^1 \int_0^\alpha p^T\nabla^2 V(x+\beta p)p \; d\beta d\alpha \le \frac{3+\delta^2}{2(1-\delta)^2}p^TQ(x)p. \tag{6}$$

The following proposition gives a condition which ensures boundedness of a polyhedral set.

Proposition 2 *[5, Theorem 2.4 and Corollary 2.5] Let $x \in \mathcal{P}$, $s(x) > 0$, suppose that column of A are linearly independent and p is given by $Q(x)p = g(x)$. Then, $\|S^{-1}(x)Ap\|_\infty < 1$ implies that $\mathcal{P}$ is bounded. Furthermore, if $\mu(x)\|g(x)\|_{Q^{-1}(x)} < 1$, then $\mathcal{P}$ is bounded.* $\square$

Proposition 3 shows that $\|p\|_Q$ can be used to bound $\|S^{-1}Ap\|_\infty$ and $\|S^{-1}Ap\|_2$. The first bound in Proposition 3 is due to Anstreicher [4, Lemma 2.3] and [3, Theorem 3.3]. The proof of the second bound is straight forward and it appears during the analysis in [3, 4, 5, 68].

Proposition 3 *Let $x \in \mathcal{P}$, and $s(x) > 0$, $\sigma_{\min}(x) \equiv \min_i\{\sigma_i(x)\}$,*

$$\mu(x) \equiv (2\sqrt{\sigma_{\min}(x)} - \sigma_{\min}(x))^{-1/2}, \text{ and } \hat{\mu}(x) \equiv \min\{[(1+\sqrt{m})/2]^{1/2}, \mu(x)\}.$$

Then for any $p \in \Re^n$,

$$\|S^{-1}(x)Ap\|_\infty \leq \hat{\mu}(x)\|p\|_{Q(x)}. \tag{7}$$

Also,

$$\|p\|_{H(x)} = \|S^{-1}(x)Ap\|_2 \leq \frac{1}{\sqrt{\sigma_{\min}(x)}}\|p\|_{Q(x)} \;\square \tag{8}$$

Proposition 4 bounds the change in various quantities as we move from x in some direction p. In particular, (9) is proved by Vaidya[68, Claim 3] and in Anstreicher[3, Lemma A.1]. The bounds in (10–11) are proved in Vaidya [68, Lemma 5] and in Anstreicher[3, Lemma 2.2]. Inequalities (12) and (13) follow from noting that $\hat{\mu}(\cdot)$ is a decreasing function of $\sigma_{\min}$.

Proposition 4 *Let $x \in \mathcal{P}$, and $s(x) > 0$. Then,*

$$0 \leq \sigma_i(x) \leq 1, i = 1, \ldots m, \text{ and } \sum_{i=1}^{m} \sigma_i(x) = n. \tag{9}$$

Let $\bar{x} = x + p$, with $\|S^{-1}(x)Ap\|_\infty \leq \delta \leq 1$. Then,

$$1 - \delta \leq \frac{s_i(\bar{x})}{s_i(x)} \leq 1 + \delta, \qquad \frac{(1-\delta)^2}{(1+\delta)^2} \leq \frac{\sigma_i(\bar{x})}{\sigma_i(x)} \leq \frac{(1+\delta)^2}{(1-\delta)^2}, \quad i = 1, \ldots, m, \tag{10}$$

and

$$\frac{(1-\delta)^2}{(1+\delta)^4}Q(x) \preceq Q(\bar{x}) \preceq \frac{(1+\delta)^2}{(1-\delta)^4}Q(x). \tag{11}$$

Furthermore, if $x = w$ *and* $\bar{x} = w + p$, $\|S^{-1}(w)Ap\|_\infty \le \delta \le 1$, *then*

$$\left(2\sqrt{\sigma_{\min}(\bar{x})\left(\frac{1+\delta}{1-\delta}\right)^2} - \sigma_{\min}(\bar{x})\left(\frac{1+\delta}{1-\delta}\right)^2\right)^{-1/2} \le \mu(w), \tag{12}$$

$$\mu(w) \le \left(2\sqrt{\sigma_{\min}(\bar{x})\left(\frac{1-\delta}{1+\delta}\right)^2} - \sigma_{\min}(\bar{x})\left(\frac{1-\delta}{1+\delta}\right)^2\right)^{-1/2} \quad \square \tag{13}$$

The following lemmas show that the ellipsoidal norm $\|.\|_Q$ used to measure distance from the volumetric center is related to the difference in the value of the volumetric barrier to its optimal value. In addition, the next lemma shows that if the gradient at the current point is small, then we are sufficiently close to the volumetric center. The first statement in Lemma 5 is a bit stronger when compared with the restatement in Anstreicher [5, Theorem 2.6], however, its proof is essentially the same.

LEMMA 5 *Let* $x \in \mathcal{P}$, $s(x) > 0$, *and* $\hat{mu}(x)\|g(x)\|_{Q^{-1}(x)} \le \gamma \le 1/6$. *Then,*

$$\|w - x\|_{Q(x)} \le 6\|g(x)\|_{Q^{-1}(x)}, \tag{14}$$

and

$$V(w) - V(x) \ge \min_{0\le\alpha\le 1} \frac{1}{(\hat{mu}(x))^2}\left(-\gamma\alpha + \frac{\alpha^2}{2(1+\alpha)^2}\right). \quad \square$$

The following lemma shows that if the value of the volumetric barrier at a point is close to its optimal value, then this point should be close to the volumetric center in Q-norm.

LEMMA 6 *Let* $x \in \mathcal{P}$, $s(x) > 0$, *and* w *be the volumetric center of* $\mathcal{P}$. *Let* $0 < V(x) - V(w) \le \frac{\delta^2}{2(1+\delta)^2(\hat{mu}(w))^2}$, $0 \le \delta \le 1$, *then* $\mu(w)\|w - x\|_{Q(w)} \le \delta$. $\square$

Proof. Assume that $\hat{\mu}(w)\|w - x\|_{Q(w)} > \delta$. Then, we have $\bar{x} = x + \lambda(w - x) = \lambda w + (1-\lambda)x$, $0 < \lambda < 1$, for which $\hat{\mu}(w)\|w - \bar{x}\|_{Q(w)} = \delta$. Due to the convexity of $V(\cdot)$, $V(\bar{x}) \le \lambda V(w) + (1-\lambda)V(x)$, hence $V(\bar{x}) - V(w) \le (1-\lambda)(V(x) - V(w)) < V(x) - V(w) \le \frac{\delta^2}{2(1+\delta)^2(\hat{\mu}(w))^2}$. Let $p = \bar{x} - w$. From Proposition 3 we have $\|S^{-1}(w)Ap\|_\infty \le \hat{\mu}(w)\|p\|_{Q(w)} = \delta$. In (5) using $g(w) = 0$, we get

$$\begin{aligned} V(\bar{x}) - V(w) &= \int_0^1\int_0^\alpha p^T\nabla^2 V(w+\beta p)p \, d\beta d\alpha \\ &\ge \frac{p^TQ(w)p}{2(1+\delta)^2} \quad \text{(using (6))} \\ &= \frac{\delta^2}{2(1+\delta)^2(\hat{\mu}(w))^2} \quad \text{(using } \|p\|_{Q(w)} = \delta/\hat{\mu}(w)). \end{aligned}$$

This is a contradiction, hence the claim follows. $\square$

Before we conclude this section we give a result on the property of the volumetric center which is used in Section 4 while analyzing our method for two-stage SCP. This lemma gives an ellipsoid that contains the set $\mathcal{P}$.

LEMMA 7 *Let w be the volumetric center of $\mathcal{P}$. Then for any $x \in \mathcal{P}$*

(i)

$$\|w - x\|_{Q(w)} \leq \frac{n}{\sqrt{\sigma_{\min}(w)}}. \tag{15}$$

Furthermore, for $\bar{x} \in \mathcal{P}$ such that $\hat{\mu}(w)\|w - \bar{x}\|_{Q(w)} \leq \delta \leq 1$, we have

(ii)

$$\|\bar{x} - x\|_{Q(\bar{x})} \leq \frac{(1+\delta)}{(1-\delta)^2}\left(\frac{n}{\sqrt{\sigma_{\min}(w)}} + \delta\right). \tag{16}$$

In particular, for $\delta \leq .01$, $\sigma_{\min}(\bar{x}) \geq .04$, we have $\sigma_{\min}(w) \geq .0384$, and

(iii)

$$\|\bar{x} - x\|_{Q(\bar{x})} \leq 5.3n \text{ and } \|\bar{x} - x\|_{H(\bar{x})} \leq 26.5n. \tag{17}$$

Proof. In order to show (15) first note that $g(w) = 0$, i.e., $\sum_{i=1}^m \frac{\sigma_i(w)}{s_i(w)} a_i = 0$ which for all $x \in \mathcal{P}$ implies that

$$\sum_{i=1}^m \frac{\sigma_i(w)}{s_i(w)} a_i^T(x - w) = 0 \text{ and } \sum_{i=1}^m \frac{\sigma_i(w)s_i(x)}{s_i(w)} = \sum_{i=1}^m \sigma_i(w) = n, \tag{18}$$

where the last equality follows from (9). For any $x \in \mathcal{P}$,

$$\begin{aligned}
\sum_{i=1}^m \frac{\sigma_i(w)(a_i^T(x-w))^2}{s_i^2(w)} &= \sum_{i=1}^m \sigma_i(w)\left(\frac{s_i(x)}{s_i(w)} - 1\right)^2 \\
&= \sum_{i=1}^m \sigma_i(w) - 2\sum_{i=1}^m \frac{\sigma_i(w)s_i(x)}{s_i(w)} + \sum_{i=1}^m \frac{\sigma_i(w)s_i^2(x)}{s_i^2(w)} \\
&= n - 2n + \sum_{i=1}^m \frac{\sigma_i(w)s_i^2(x)}{s_i^2(w)} \quad \text{(using 18)} \\
&\leq -n + \frac{1}{\sigma_{\min}(w)} \sum_{i=1}^m \frac{\sigma_i^2(w)s_i^2(x)}{s_i^2(w)} \quad \text{(using } \sigma_i(w) \geq 0\text{)} \\
&\leq -n + \frac{1}{\sigma_{\min}(w)}\left(\sum_{i=1}^m \frac{\sigma_i(w)s_i(x)}{s_i(w)}\right)^2 = -n + \frac{n^2}{\sigma_{\min}(w)}.
\end{aligned}$$

This proves (15). Now observe that

$$\|\bar{x} - x\|_{Q(\bar{x})} = \|\bar{x} - w + w - x\|_{Q(\bar{x})} \leq \|\bar{x} - w\|_{Q(\bar{x})} + \|w - x\|_{Q(\bar{x})}. \tag{19}$$

Under the hypothesis and using Proposition 3 we have $\|S^{-1}(w)Ap\|_\infty \leq \delta$, hence from (10) in Proposition 4 we have

$$1 - \delta \leq \frac{s_i(\bar{x})}{s_i(w)} \leq 1 + \delta, \text{ and } \frac{(1-\delta)^2}{(1+\delta)^2} \leq \frac{\sigma_i(\bar{x})}{\sigma_i(w)} \leq \frac{(1+\delta)^2}{(1-\delta)^2},$$

hence

$$\|w - x\|^2_{Q(\bar{x})} = \sum_{i=1}^m \frac{\sigma_i(\bar{x})(a_i^T(w-x))^2}{s_i^2(\bar{x})} \leq \frac{(1+\delta)^2}{(1-\delta)^4} \sum_{i=1}^m \frac{\sigma_i(w)(a_i^T(w-x))^2}{s_i^2(w)},$$

which gives $\|w - x\|_{Q(\bar{x})} \leq \frac{(1+\delta)}{(1-\delta)^2}\|w - x\|_{Q(w)}$. Similarly,

$$\|w - \bar{x}\|^2_{Q(\bar{x})} = \sum_{i=1}^m \frac{\sigma_i(\bar{x})(a_i^T(w-\bar{x}))^2}{s_i^2(\bar{x})} \leq \frac{(1+\delta)^2}{(1-\delta)^4} \sum_{i=1}^m \frac{\sigma_i(w)(a_i^T(w-\bar{x}))^2}{s_i^2(w)} \leq \frac{(1+\delta)^2}{(1-\delta)^4} \frac{\delta^2}{\hat{\mu}^2(w)},$$

which implies $\|w - \bar{x}\|_{Q(\bar{x})} \leq \frac{(1+\delta)\delta}{(1-\delta)^2 \hat{\mu}(w)} \leq \frac{(1+\delta)\delta}{(1-\delta)^2}$, since $\hat{\mu}(w) \geq 1$. By using the last two inequalities in (19), and using (15) we obtain

$$\|\bar{x} - x\|_{Q(\bar{x})} \leq \frac{(1+\delta)}{(1-\delta)^2} \left(\frac{n}{\sqrt{\sigma_{\min}(w)}} + \delta \right),$$

which gives the desired result in (16). Also from (8) in Proposition 3 we have $\|\bar{x} - x\|_{H(\bar{x})} \leq \|\bar{x} - x\|_{Q(\bar{x})}/\sqrt{\sigma_{\min}(\bar{x})}$. This together with the choice of constants in (16) gives the desired bounds in (17). □

3.2 Newton-like Steps

At a given point $x \in \mathcal{P}$, $s(x) > 0$, the search direction d is computed by solving

$$Q(x)d = -\frac{1}{\hat{\mu}(x)\|g(x)\|_{Q(x)^{-1}}} g(x), \tag{20}$$

and a new iterate is generated as

$$x(\alpha) = x + \alpha d, \tag{21}$$

here α is a step length parameter. We would like to know the improvement in $V(\cdot)$ at the new iterate $x(\alpha)$ for a specific choice of α. The following theorem accomplishes this. The bound in (22) in this theorem was proved in Anstreicher[5, Lemma 2.8]. Bounds based on specific parameter choices are straight forward and we omit them here.

Theorem 8 *If d is computed from (20) and $x(\alpha)$ is given by (21), then,*

$$V(x(\alpha)) - V(x) \le \frac{1}{\mu^2(x)} \left(-\alpha\mu(x)\|g(x)\|_{Q(x)^{-1}} + \frac{(3+\alpha^2)\alpha^2}{2(1-\alpha)^2} \right). \tag{22}$$

Furthermore, if $\hat{\mu}(x)\|g(x)\|_{Q(x)^{-1}} \ge 1$, then for $\alpha = .2$ we have $V(x(\alpha)) - V(x) \le -.1/\sqrt{m}$. Otherwise, $\hat{\mu}(x)\|g(x)\|_{Q(x)^{-1}} \le 1$, and for $\alpha = .2\hat{\mu}(x)\|g(x)\|_{Q(x)^{-1}}$ we have $V(x(\alpha)) - V(x) \le -.1\|g(x)\|^2_{Q(x)^{-1}}$. □

Corollary 9 *Let d be computed as in (20) and $x(\alpha)$ be given by (21). Also let α be chosen as in Theorem 8. Then, starting from a $x \in \mathcal{P}, s(x) > 0$, satisfying $V(x) - V(w) = O(1)$, we can obtain a $\bar{x}$ satisfying $\hat{\mu}(\bar{x})\|g(\bar{x})\|_{Q^{-1}(\bar{x})} \le .01/6$ in $O(\sqrt{m})$ iterations.*

Proof. Let $\bar{g} = g(\bar{x})$, $\bar{Q} = Q(\bar{x})$, and $\bar{\hat{\mu}} = \hat{\mu}(\bar{x})$, where $\bar{x}$ is an iterate after $O(\sqrt{m})$ Newton-like iterations. If $\|\bar{g}\|_{\bar{Q}^{-1}} \ge \frac{10^{-2}}{6m^{1/4}}$, then from Theorem 8 we know that at all Newton-like iterations $V(\cdot)$ is decreased by at least $\frac{10^{-5}}{36m^{1/2}}$, and we can not have more than $O(\sqrt{m})$ iterations like this. Otherwise, $\bar{\hat{\mu}}\|\bar{g}\|_{\bar{Q}^{-1}} \le .01/6$, since $\bar{\hat{\mu}} \le m^{1/4}$. Hence the corollary follows. □ We point out that in Theorem 8 we have used values of α that would give good choices in practice. This is important since evaluation of $V(\cdot)$ is expensive, which makes performing line searches expensive.

3.3 Adding A Cut

Let $\tilde{\mathcal{P}} \equiv \{x | x \in \mathcal{P}, a^T_{m+1}x \ge b_{m+1}\}$ be the new region obtained after adding an inequality to $\mathcal{P}$. Let $\tilde{A} \equiv \begin{pmatrix} A \\ a^T_{m+1} \end{pmatrix}$, $\tilde{b} \equiv \begin{pmatrix} b \\ b_{m+1} \end{pmatrix}$. Note that $x \in \tilde{\mathcal{P}} \Rightarrow x \in \mathcal{P}$. For $x \in \tilde{\mathcal{P}}$, let $\tilde{s}(x) \equiv \tilde{A}x - \tilde{b}$, and $\tilde{S}(x) \equiv diag(\tilde{s}(x))$. Clearly, $\tilde{s}(x) = (s(x), s_{m+1}(x))$, where $s_{m+1}(x) = a^T_{m+1}x - b_{m+1}$. Let

$$\tau(x) \equiv \frac{a^T_{m+1}(A^T S(x)^{-2} A)^{-1} a_{m+1}}{s^2_{m+1}(x)} = \frac{a^T_{m+1} H(x)^{-1} a_{m+1}}{s^2_{m+1}(x)}. \tag{23}$$

Let $\tilde{V}(.)$ be the volumetric barrier function for $\tilde{\mathcal{P}}$ and $\tilde{w}$ be its volumetric center. Let $\tilde{H}(x) \equiv \tilde{A}^T \tilde{S}(x)^{-2} \tilde{A}$. For $x \in \tilde{P}$,

$$\begin{aligned} \tilde{V}(x) &= \frac{1}{2} ldet(\tilde{H}(x)) \\ &= \frac{1}{2} ldet(H(x) + \frac{1}{s^2_{m+1}(x)} a_{m+1} a^T_{m+1}) \\ &= \frac{1}{2} ldet\left(H(x) \left[I + \frac{1}{s^2_{m+1}(x)} H(x)^{-1} a_{m+1} a^T_{m+1} \right] \right) \\ &= V(x) + \frac{1}{2} \ln\left(1 + \frac{a^T_{m+1} H(x)^{-1} a_{m+1}}{s^2_{m+1}(x)} \right), \end{aligned} \tag{24}$$

where the last equality uses the fact that $det(I + uv^T) = 1 + v^T u$.

The following theorem shows that the quantity $\tilde{V}(\tilde{w}) - V(w)$ has a constant lower bound, i.e., the value of volumetric barrier increases by sufficiently large amount after adding a cut. We use it in establishing the global convergence of the volumetric method.

THEOREM 10 *Let $x \in \mathcal{P}$, $s(x) > 0$ be such that $\hat{\mu}(x)\|g(x)\|_{Q^{-1}(x)} \le \hat{\delta}/6$, $0 \le \hat{\delta} < 1$, and $\tau(x)$ be given by (23). Then,*

$$\tilde{V}(\tilde{w}) - V(w) \ge \min_{0 \le \tilde{\delta} \le 1} \left\{ \frac{\tilde{\delta}^2}{2(1+\tilde{\delta})^2(\mu(w))^2} + \frac{1}{2} \ln \left(1 + \tau(x) \frac{(1-\tilde{\delta})^2(1-\hat{\delta})^2}{(1+\delta')^2} \right) \right\}, \quad (25)$$

where δ' is defined in (33) below.

Proof. From (24) we have

$$\tilde{V}(\tilde{w}) = V(\tilde{w}) + \frac{1}{2} \ln \left(1 + \frac{a_{m+1}^T H(\tilde{w})^{-1} a_{m+1}}{s_{m+1}^2(\tilde{w})} \right),$$

which gives

$$\tilde{V}(\tilde{w}) - V(w) = V(\tilde{w}) - V(w) + \frac{1}{2} \ln \left(1 + \tau(x) \left[\frac{s_{m+1}^2(x)}{s_{m+1}^2(\tilde{w})} \right] \left[\frac{a_{m+1}^T H(\tilde{w})^{-1} a_{m+1}}{a_{m+1}^T H(x)^{-1} a_{m+1}} \right] \right). \quad (26)$$

If $V(\tilde{w}) - V(w) \ge \frac{1}{8(\mu(w))^2}$, then the result follows immediately from (26) by taking $\tilde{\delta} = 1$ in (25), so without loss of generality assume that $V(\tilde{w}) - V(w) = \frac{\tilde{\delta}^2}{2(1+\tilde{\delta})^2(\mu(w))^2}$, $0 \le \tilde{\delta} < 1$. We lower bound the term inside ln(.) in (26). From Lemma 6, we have $\mu(w)\|w - \tilde{w}\|_{Q(w)} \le \tilde{\delta}$. This from (7) in Proposition 3 gives $\|S^{-1}(w)A(\tilde{w} - w)\|_\infty \le \tilde{\delta}$, and thus from (10) in Proposition 4 we have

$$1 - \tilde{\delta} \le \frac{s_i(\tilde{w})}{s_i(w)} \le 1 + \tilde{\delta}, \qquad \frac{(1-\tilde{\delta})^2}{(1+\tilde{\delta})^2} \le \frac{\sigma_i(\tilde{w})}{\sigma_i(w)} \le \frac{(1+\tilde{\delta})^2}{(1-\tilde{\delta})^2}, \quad i = 1, \ldots, m. \quad (27)$$

For any $x \in \mathcal{P}$ satisfying $\mu(x)\|g(x)\|_{Q^{-1}(x)} \le \hat{\delta}/6$ from (14) in Lemma 5 we have $\mu(x)\|w - x\|_{Q(x)} \le 6\mu(x)\|g(x)\|_{Q^{-1}(x)} \le \hat{\delta}$. This from (7) in Proposition 3 gives $\|S^{-1}(x)A(w - x)\|_\infty \le \hat{\delta}$, and thus from (10) in Proposition 4 we have,

$$1 - \hat{\delta} \le \frac{s_i(w)}{s_i(x)} \le 1 + \hat{\delta}, \qquad \frac{(1-\hat{\delta})^2}{(1+\hat{\delta})^2} \le \frac{\sigma_i(w)}{\sigma_i(x)} \le \frac{(1+\hat{\delta})^2}{(1-\hat{\delta})^2}, \quad i = 1, \ldots, m. \quad (28)$$

Now using (27) and (28) for any $p \in \Re^n$,

$$p^T H(x) p = \sum_{i=1}^m \frac{(p^T a_i)^2}{s_i^2(x)} \ge (1-\tilde{\delta})^2(1-\hat{\delta})^2 \sum_{i=1}^m \frac{(p^T a_i)^2}{s_i^2(\tilde{w})} = (1-\tilde{\delta})^2(1-\hat{\delta})^2 p^T H(\tilde{w}) p.$$

Hence from Proposition **??**, for all $p \in \Re^n$ we have

$$p^T H(x)^{-1} p \leq \frac{1}{(1-\tilde{\delta})^2(1-\hat{\delta})^2} p^T H(\tilde{w})^{-1} p,$$

which for $p = a_{m+1}$ gives

$$\frac{a_{m+1}^T H(\tilde{w})^{-1} a_{m+1}}{a_{m+1}^T H(x)^{-1} a_{m+1}} \geq (1-\tilde{\delta})^2(1-\hat{\delta})^2. \tag{29}$$

Towards bounding $s_{m+1}(x)/s_{m+1}(\tilde{w})$ first note that

$$\left|\frac{s_{m+1}(\tilde{w})}{s_{m+1}(x)} - 1\right| = \left|\frac{a_{m+1}^T(\tilde{w}-x)}{s_{m+1}(x)}\right| \leq \frac{\|a_{m+1}\|_{H(w)^{-1}} \ \|\tilde{w}-x\|_{H(w)}}{s_{m+1}(x)}. \tag{30}$$

We now bound the two terms in (30). Using (28) we have

$$\|a_{m+1}\|^2_{H(w)} = \sum_{i=1}^{m} \frac{({a_i}^T a_{m+1})^2}{s_i^2(w)} \geq \frac{1}{(1+\hat{\delta})^2} \sum_{i=1}^{m} \frac{({a_i}^T a_{m+1})^2}{s_i^2(x)} = \frac{1}{(1+\hat{\delta})^2} \|a_{m+1}\|^2_{H(x)},$$

which from Proposition **??** gives

$$\|a_{m+1}\|^2_{H(w)^{-1}} \leq (1+\hat{\delta})^2 \|a_{m+1}\|^2_{H(x)^{-1}}. \tag{31}$$

The triangular inequality gives $\|\tilde{w}-x\|_{H(w)} \leq \|\tilde{w}-w\|_{H(w)} + \|w-x\|_{H(w)}$. From (8) in Proposition 3 we have $\|\tilde{w}-w\|_{H(w)} \leq \frac{\|\tilde{w}-w\|_{Q(w)}}{\sigma^{1/2}_{\min}(w)} \leq \frac{\tilde{\delta}}{\mu(w)\sigma^{1/2}_{\min}(w)}$, since $\mu(w)\|w-\tilde{w}\|_{Q(w)} \leq \tilde{\delta}$. Similarly, $\|w-x\|_{H(x)} \leq \frac{\|w-x\|_{Q(x)}}{\sigma^{1/2}_{\min}(x)} \leq \frac{\hat{\delta}}{\mu(x)\sigma^{1/2}_{\min}(x)}$, since $\mu(x)\|w-x\|_{Q(x)} \leq \hat{\delta}$. Hence, from using (28) we have $\|w-x\|_{H(w)} \leq \frac{\hat{\delta}}{(1-\hat{\delta})\mu(x)\sigma^{1/2}_{\min}(x)}$. Therefore, we have

$$\|\tilde{w}-x\|_{H(w)} \leq \frac{\tilde{\delta}}{\mu(w)\sigma^{1/2}_{\min}(w)} + \frac{\hat{\delta}}{(1-\hat{\delta})\mu(x)\sigma^{1/2}_{\min}(x)}. \tag{32}$$

Substituting (32) and (31) in equation (30) and using (23) gives,

$$\left|\frac{s_{m+1}(\tilde{w})}{s_{m+1}(x)} - 1\right| \leq \tau(x)^{1/2}(1+\hat{\delta}) \left(\frac{\tilde{\delta}}{\mu(w)\sigma^{1/2}_{\min}(w)} + \frac{\hat{\delta}}{(1-\hat{\delta})\mu(x)\sigma^{1/2}_{\min}(x)}\right) \equiv \delta'. \tag{33}$$

hence

$$1-\delta' \leq \frac{s_{m+1}(\tilde{w})}{s_{m+1}(x)} \leq 1+\delta'. \tag{34}$$

Now using inequalities (34) and (29) in (26) we get

$$\tilde{V}(\tilde{w}) - V(w) \geq \frac{\tilde{\delta}^2}{2(1+\tilde{\delta})^2(\mu(w))^2} + \frac{1}{2}\ln\left(1 + \tau(x)\frac{(1-\tilde{\delta})^2(1-\hat{\delta})^2}{(1+\delta')^2}\right).$$

The theorem follows. $\square$

Corollary 11 *Let $x \in \tilde{\mathcal{P}}$ with $\mu(x)\|g(x)\|_{Q^{-1}(x)} \leq \hat{\delta}/6$, $\hat{\delta} = .01$. Let $\tau(x) = 4$, and $\mu(x) \geq .04$. Then,*

$$\tilde{V}(\tilde{w}) - V(w) \geq .0358. \tag{35}$$

Also,

$$\tilde{V}(x) - \tilde{V}(\tilde{w}) \leq .77 \tag{36}$$

Proof. From Proposition 3, and (12, 13) we know that $.0384 \leq \sigma_{\min}(w) \leq .042$ and straight forward calculation gives $1.651 \leq \mu(w) \leq 1.682$. The bound in (35) is obtained from a numerical calculation, which shows that the minimum in (25) occurs at $\tilde{\delta} \approx 0.71$, with the value $\approx .0358$. Now to show (36) first observe that from (24) we have,

$$\tilde{V}(x) - \tilde{V}(\tilde{w}) = V(x) - V(w) + V(w) - \tilde{V}(\tilde{w}) + \frac{1}{2}\ln(1 + \tau(x)).$$

Now from Lemma 5 for $\gamma = .01/6$ numerical calculations show that for $\alpha = .0017$, $V(x) - V(w) \leq 5 \times 10^{-7}$. Using (35) and the value of $\tau(x) = 4$ we obtain the result. $\square$

3.4 Dropping a Constraint

Without loss of generality assume that $\sigma_{\min}(x) = \sigma_m(x)$, and assume that mth constraint is dropped. Let $\tilde{\mathcal{P}} \equiv \{x | a_i^T x \geq b_i, i = 1, \ldots m-1\}$, and $A = \begin{pmatrix} \tilde{A} \\ a_m^T \end{pmatrix}$, $b = \begin{pmatrix} \tilde{b} \\ b_m \end{pmatrix}$. For $x \in \tilde{\mathcal{P}}$, let $\tilde{s}(x) \equiv \tilde{A}x - \tilde{b}$. Note that $s(x) = (\tilde{s}(x), s_m(x))$, and $x \in \mathcal{P} \Rightarrow x \in \tilde{\mathcal{P}}$. Let $\tilde{H}(x) \equiv \tilde{A}^T \tilde{S}^{-2}(x)\tilde{A}$, and $\tilde{V}(x) \equiv \frac{1}{2} ldet(\tilde{H}(x))$ be the volumetric barrier for $\tilde{\mathcal{P}}$ and $\tilde{w}$ be its volumetric center. Let $\tilde{\sigma}_i(x) \equiv \frac{a_i^T \tilde{H}(x)^{-1} a_i}{\tilde{s}_i^2(x)}, i = 1, \ldots m-1$ and define $\tilde{Q}(x), \tilde{\sigma}_{\min}(x), \tilde{\mu}(x), \hat{\tilde{\mu}}(x)$ similarly. For any $x \in \mathcal{P}$

$$\begin{aligned}
\tilde{V}(x) &= \frac{1}{2} ldet(\tilde{H}(x)) \\
&= \frac{1}{2} ldet(H(x) - \frac{1}{s_m^2(x)} a_m a_m^T) \\
&= \frac{1}{2} ldet\left(H(x)\left[I - \frac{1}{s_m^2(x)} H(x)^{-1} a_m a_m^T\right]\right) \\
&= V(x) + \frac{1}{2}\ln\left(1 - \frac{a_m^T H(x)^{-1} a_m}{s_m^2(x)}\right) \\
&= V(x) + \frac{1}{2}\ln(1 - \sigma_m(x)))
\end{aligned} \tag{37}$$

where the second last equality uses the fact that $det(I - uv^T) = 1 - v^T u$. We need to bound $\tilde{V}(\tilde{w}) - V(w)$. This is accomplished in Corollary 13, which uses the following theorem due to Anstreicher[4, Lemma 5.2, Theorem 5.3].

THEOREM 12 *Suppose that $\tilde{\mathcal{P}}$ is obtained by deleting the mth constraint and $\sigma_{\min}(x) = \sigma_m(x)$. Then,*

$$\sigma_i(x) \leq \tilde{\sigma}_i(x) \leq \sigma_i(x)/(1 - \sigma_{\min}(x)), i = 1, \ldots m-1.$$

and

$$\|\tilde{g}(x)\|_{\tilde{Q}(x)^{-1}} \leq \frac{1}{\sqrt{1 - \sigma_{\min}(x)}} \left(\|g(x)\|_{Q(x)^{-1}} + \sigma_{\min}(x) \left(1 + \frac{1}{\sqrt{1 - \sigma_{\min}(x)}} \right) \right). \quad \square$$

COROLLARY 13 *Let $x \in \mathcal{P}$ with $\mu(x)\|g(x)\|_{Q^{-1}(x)} \leq \hat{\delta}/6, \hat{\delta} = .01$. Assume that we have chosen a constraint for deletion for which $\sigma_i(x) \leq .04$. Then,*

$$V(w) - \tilde{V}(\tilde{w}) \leq .0315, \tag{38}$$

Also,

$$\tilde{V}(x) - \tilde{V}(\tilde{w}) \leq .012. \tag{39}$$

Proof. The proof of this corollary follows the steps in the proof of Anstreicher [5, Theorem 5.2]. From Theorem 12 we have $\sigma_{\min}(w) \leq \sigma_i(w) \leq \tilde{\sigma}_i(w)$, for $i = 1, \ldots m-1$, hence we have $\sigma_{\min}(w) \leq \tilde{\sigma}_{\min}(w)$. Therefore, since $\hat{\mu}(.)$ is a decreasing function of $\sigma_{\min}$, we have $\hat{\tilde{\mu}}(w) \leq \hat{\mu}(w)$. Furthermore, $\tilde{\mu}(w) \leq \mu(w)$, since we have reduced the number of constraints by one. By taking $x = w$ in Theorem 12, noting that $g(w) = 0$, and multiplying both sides by $\tilde{\mu}(w)$ we have,

$$\tilde{\mu}(w)\|\tilde{g}(w)\|_{\tilde{Q}(w)^{-1}} \leq \frac{\tilde{\mu}(w)\sigma_{\min}(w)}{\sqrt{1 - \sigma_{\min}(w)}} \left(1 + \frac{1}{\sqrt{1 - \sigma_{\min}(w)}} \right).$$

From Proposition 3, and (10–13) we have $.0384 \leq \sigma_{\min}(w) \leq .042$ and $1.651 \leq \mu(w) \leq 1.682$. Using this in above it is easy to see that

$$\tilde{\mu}(w)\|\tilde{g}(w)\|_{\tilde{Q}(w)^{-1}} \leq .146 \frac{\tilde{\mu}(w)}{\mu(w)} \leq .146.$$

This from Proposition 2 shows that $\tilde{\mathcal{P}}$ is bounded. First consider the case where $\tilde{\mu}(w) \leq .81\mu(w)$. In this case, $\tilde{\mu}(w)\|\tilde{g}(w)\|_{\tilde{Q}(w)^{-1}} \leq .119$ and in Lemma 5 taking $\gamma = .119$ and using numerical calculations it is seen that $\tilde{V}(\tilde{w}) - \tilde{V}(w) \geq -\frac{.01}{\tilde{\mu}^2(w)} \geq -.01$ (minimum at $\alpha \approx 0.21$).

Now consider the case where $\tilde{\mu}(w) \geq .81\mu(w)$. We still have $\tilde{\mu}(w)\|\tilde{g}(w)\|_{\tilde{Q}(w)^{-1}} \leq .146$ and $\tilde{\mu}(w) \geq 1.33$. Once again from Lemma 5 by using numerical calculations it is seen that $\tilde{V}(\tilde{w}) - \tilde{V}(w) \geq -.01$ (minimum at $\alpha \approx .4$). Hence, in all cases we have

$$\tilde{V}(\tilde{w}) - \tilde{V}(w) \geq -.01.$$

From (37) for $x = w$ we have

$$\begin{aligned}
\tilde{V}(\tilde{w}) - V(w) &= \tilde{V}(\tilde{w}) - \tilde{V}(w) + \frac{1}{2}\ln(1 - \sigma_{\min}(w)) \\
&\geq \tilde{V}(\tilde{w}) - \tilde{V}(w) + \frac{1}{2}\ln(1 - .42) \\
&\geq -.01 - .0215 = -.0315
\end{aligned}$$

Now from (37) we have

$$\tilde{V}(x) - \tilde{V}(\tilde{w}) = V(x) - V(w) + V(w) - \tilde{V}(\tilde{w}) + \frac{1}{2}\ln(1 - \sigma_{\min}(x)). \qquad (40)$$

From Lemma 5 (for $\gamma = .01/6$) numerical calculations show that $V(x) - V(w) \leq 5 \times 10^{-7}$. Hence, $\tilde{V}(x) - \tilde{V}(\tilde{w}) \leq 5 \times 10^{-7} + .0315 + \frac{1}{2}\ln(1 - .04) \leq .012$ $\square$

3.5 Near-Central Cut VCM and its Convergence.

We now describe the near-central cut version of the volumetric center method and provide a convergence analysis for this algorithm. At the start of each iteration $k \geq 0$, we have a bounded polyhedron $\mathcal{P}^k$ which contains the optimal solution. The hypercube containing $\mathcal{C}$ is taken as a starting polyhedron. It is straight forward to show that $x^0 = 0$ is the volumetric center of $\mathcal{P}^0$. The algorithm is described below.

Algorithm 1. (Near-Central Cut Volumetric Center Method)

Input. x^0, $\mathcal{P}^0$, m_0, L, $\sigma = .04, \hat{\delta} = .01, \tau = 4$

Step 1. (Termination Check) If $V^k(x^k) \geq .7nL + n\ln(m_k)$, then STOP. Else go to Step 2.

Step 2. (Decide if we should add or drop a constraint) If $\sigma_{\min}(x^k) \geq \sigma$, go to Step 3, else go to Step 4.

Step 3. (Add a Cut) Call the oracle to check if $x^k \in \mathcal{C}$. If yes, STOP. Otherwise the oracle returns a vector $a \in \Re^n$ such that $a^T x \geq a^T x^k$ for all $x \in \mathcal{C}$. Let $s^k = A^k x^k - b^k$, $S^k = diag(s^k)$. Add the constraint $a^T x \geq \beta$ to the existing constraint system. Here $\beta = a^T x^k - (a^T ({A^k}^T (S^k)^{-2} A^k)^{-1} a/\tau)^{1/2}$. Represent the new constraint system by (A^{k+1}, b^{k+1}), $m_k = m_k + 1$. Go to Step 5.

Step 4. (Delete a Constraint) Suppose that $\sigma_{\min}(x^k) = \sigma_j(x^k) < \sigma$. Let (A^{k+1}, b^{k+1}) be the constraint system obtained by removing the jth row of (A^k, b^k), $m_k = m_k - 1$. Go to Step 5.

Step 5. (Centering Steps) Let $\bar{x}^0 = x^k$. Starting from $\bar{x}^0$ take a sequence of damped Newton-like steps of the form $\bar{x}^{j+1} = \bar{x}^j - \alpha Q^{-1}(\bar{x}^j)g(\bar{x}^j)$, $j \geq 0$, until $\mu(\bar{x}^J)\|g(\bar{x}^J)\|_{Q^{-1}(\bar{x}^J)} \leq \hat{\delta}/6$. Let $x^{k+1} = \bar{x}^J$, $k = k+1$, and go to Step 1

The following Lemma from Anstreicher [4, Lemma 3.1] shows that if the algorithm terminates in Step 1, then the volume of $\mathcal{C}$ is sufficiently small.

LEMMA 14 *Consider the volumetric cutting plane algorithm with* $\hat{\delta}/6 \leq .03$ *and assume that* $L \geq 1$, *and let* $V^k_{\max} = .7nL + n\ln(m_k)$. *Then, termination in Step 1 proves that the volume of* $\mathcal{C}$ *is less than that of an* $n-$*dimensional sphere of radius* 2^{-L} $\square$

The next result is on the number of iterations after which we meet the termination criterion in Step 1.

THEOREM 15 *For* $\sigma = .04$, $\tau = 4$, $\hat{\delta} = .01$ *in the volumetric cutting plane algorithm the termination criterion in Step 1 is satisfied after* $O(nL)$ *major iterations, while performing* $O(\sqrt{n})$ *Newton-like steps at each major iteration. The total number of calls to the oracle are* $O(nL)$.

Proof. At a major iteration we either add a cut or drop a constraint and recenter. Since $\sum_{i=1}^{m_k} \sigma_i(x^k) = n$, and a constraint is added only when $\sigma_i(x^k) \geq \sigma = .04$, the total number of constraints can not exceed $n/\sigma + 1$, i.e., $m_k \leq 25n + 1$. Also since $\mathcal{P}^k$ is bounded $m_k \geq n+1$. Therefore, the difference of the number of added cuts and deleted constraints is bounded by $24n$. If we add a cut at iteration k from Corollary 11, $V^{k+1}(w^{k+1}) - V^k(w^k) \geq .0358$. If we delete a constraint at an iteration from Corollary 13 we have $V^{k+1}(w^{k+1}) - V^k(w^k) \geq -.0315$. Hence, $V^k(w^k) - V^0(w^0) \geq \frac{.043(k-24n)}{2} - 24 \times .0315n$. Note that $V^0(x^0) \leq -n\hat{L}$. Hence, after $O(n(L+\hat{L}))$ iterations $V^k(w^k) \geq .7nL + n\ln(m_k) + 5 \times 10^{-7}$. Since at each iteration for $\hat{\delta} = .01$, $V^k(x^k) - V^k(w^k) \leq 5 \times 10^{-7}$ the termination check in Step 1 is satisfied after at most $O(n(L+\hat{L}))$ iterations. Corollary 11 together with Theorem 8 shows that the number of Newton-like iterations required to recenter after a cut is added is $O(\sqrt{n})$ because $m_k = O(n)$. Similarly, Corollary 13 together with Theorem 8 shows that the number of Newton-like iterations required to recenter after a constraint is dropped is also $O(\sqrt{n})$. The calls to the oracle are $O(nL)$ because we only call the oracle at a major iteration in the case of adding a cut. $\square$.

3.6 Translating Cuts

In the context of optimization problem it is often possible to translate (strengthen) a previously generated objective cut. This can help speed up the algorithm. It is therefore important to analyze the effect of cut translations. The analysis for such a modification was done in Ariyawansa and Jiang [7] for Vaidya's VCM in the context of using this method for solving SLPF. Our analysis here is for the near-central cut variant and it is considerably simpler than the analysis of Ariyawansa and Jiang [7]. Furthermore, we allow for translation of more than one previously generated cuts simultaneously. Our main purpose is to derive conditions that can be easily checked and that ensure that we can recenter in $O(\sqrt{n})$ Newton-like iterations, as in the case of adding new cuts and dropping a constraint. In order to simplify notations we assume that all constraints are being translated, by changing the constraint right hand side from b_i to $\tilde{b}_i$, $\tilde{b}_i \geq b_i$.

Let $\tilde{\mathcal{P}} \equiv \{x | a_i^T x \geq \tilde{b}_i, i = 1, \ldots m\}$, be the new region obtained after translation. Let $\tilde{b} \equiv (\tilde{b}_i)$ and let $\tilde{s}(x) \equiv Ax - \tilde{b}$ and $\tilde{S}(x) \equiv diag(\tilde{s}(x))$. Note that $x \in \tilde{\mathcal{P}} \Rightarrow x \in \mathcal{P}$ and $s(x) - \tilde{s}(x) = \tilde{b} - b \geq 0$. Let $\tilde{V}(.)$ be the volumetric barrier function for $\tilde{\mathcal{P}}$ and $\tilde{w}$ be its volumetric center. Let $\tilde{H}(x) \equiv A^T \tilde{S}^{-2}(x) A$ and let $H(x) = L(x)L(x)^T$, where $L(x)$ is the Cholesky factor of $H(x)$. Let

$$M(x) \equiv I + \sum_{i=1}^{m} \left(\frac{1}{\tilde{s}_i^2(x)} - \frac{1}{s_i^2(x)} \right) L^{-1}(x) a_i a_i^T L^{-T}(x).$$

Now for an $x \in \tilde{P}$,

$$\begin{aligned}
\tilde{V}(x) &= \frac{1}{2} ldet(\tilde{H}(x)) \\
&= \frac{1}{2} ldet \left(H(x) + \sum_{i=1}^{m} \left(\frac{1}{\tilde{s}_i^2(x)} - \frac{1}{s_i^2(x)} \right) a_i a_i^T \right) \\
&= \frac{1}{2} ldet \left(L(x) \left[I + \sum_{i=1}^{m} \left(\frac{1}{\tilde{s}_i^2(x)} - \frac{1}{s_i^2(x)} \right) L^{-1}(x) a_i a_i^T L^{-T}(x) \right] L^T(x) \right) \\
&= \frac{1}{2} \ln \left(det(L(x)) det(L^T(x)) det \left(\left[I + \sum_{i=1}^{m} \left(\frac{1}{\tilde{s}_i^2(x)} - \frac{1}{s_i^2(x)} \right) L^{-1}(x) a_i a_i^T L^{-T}(x) \right] \right) \right) \\
&= \frac{1}{2} \ln(det(L(x)) det(L^T(x))) + \frac{1}{2} ldet(M(x)) \\
&= V(x) + \frac{1}{2} ldet(M(x)). \qquad (41)
\end{aligned}$$

Since all singular values of $M(x)$ are larger than 1, $ldet(M(x)) \geq 0$. Hence, from (41),

$\tilde{V}(x) \geq V(x)$ for all $x \in \tilde{\mathcal{P}}$. In particular, $\tilde{V}(\tilde{w}) \geq V(\tilde{w})$. Also from (41) we have

$$\begin{aligned}
\tilde{V}(x) - \tilde{V}(\tilde{w}) &= V(x) - V(w) + V(w) - V(\tilde{w}) + V(\tilde{w}) - \tilde{V}(\tilde{w}) + \frac{1}{2} ldet(M(x)) \\
&\leq V(x) - V(w) + V(w) - V(\tilde{w}) + \frac{1}{2} ldet(M(x)) \\
&\leq V(x) - V(w) + \frac{1}{2} ldet(M(x)), \qquad (42)
\end{aligned}$$

where the last inequality follows because w is the minimizer of $V(.)$. Since the Newton-like iterations terminate when $\mu(x)\|g(x)\|_{Q^{-1}(x)} \leq .01/6$, ensuring $V(x) - V(w) \leq 5 \times 10^{-7}$, in order to have an $O(1)$ bound on $\tilde{V}(x) - \tilde{V}(\tilde{w})$ it is sufficient to have an $O(1)$ bound on $ldet(M(x))$. We now give conditions that ensure this bound. We need the following proposition for this purpose.

PROPOSITION 16 *Let* $u_1, \ldots u_m \in \Re^n$, $\alpha_i \geq 0$, *and let*

$$M_k = \left(I + \sum_{i=1}^{k} \alpha_i u_i u_i^T \right), k = 1, \ldots m.$$

Then, $det(M_m) \leq \prod_{i=1}^{m} (1 + \alpha_i u_i^T u_i)$.

Proof. The bound is satisfied with equality for $m = 1$. Assume that it is true for $k < m$. Now,

$$\begin{aligned}
det(M_{k+1}) &= det(M_k + \alpha_{k+1} u_{k+1} u_{k+1}^T) \\
&= det\left(M_k \left(I + \alpha_{k+1} M_k^{-1} u_{k+1} u_{k+1}^T\right)\right) \\
&= det(M_k) det\left(I + \alpha_{k+1} M_k^{-1} u_{k+1} u_{k+1}^T\right) \\
&= det(M_k)(1 + \alpha_{k+1} u_{k+1}^T M_k^{-1} u_{k+1}) \\
&\leq det(M_k)(1 + \alpha_{k+1} u_{k+1}^T u_{k+1}),
\end{aligned}$$

here the last inequality follows from noting that $p^T M_k p \geq p^T p$ for all $p \in \Re^n$ and using Proposition **??**. $\square$

Since $\ln(.)$ is an increasing function, an immediate consequence of Proposition 16 is that

$$\begin{aligned}
ldet(M(x)) &\leq \sum_{i=1}^{m} \ln\left(1 + \left(\frac{1}{\tilde{s}_i^2(x)} - \frac{1}{s_i^2(x)}\right) a_i^T L^{-T}(x) L^{-1}(x) a_i\right) \\
&= \sum_{i=1}^{m} \ln\left(1 + \left(\frac{1}{\tilde{s}_i^2(x)} - \frac{1}{s_i^2(x)}\right) a_i^T H^{-1}(x) a_i\right) \\
&= \sum_{i=1}^{m} \ln\left(1 + \left(\frac{s_i^2(x)}{\tilde{s}_i^2(x)} - 1\right) \sigma_i(x)\right),
\end{aligned}$$

where the last equality used the definition of $\sigma_i(x)$. The following theorem is now immediate.

THEOREM 17 *Assume that the constraints are translated so that*

$$\sum_{i=1}^{m} \ln\left(1 + \left(\frac{s_i^2(x)}{\tilde{s}_i^2(x)} - 1\right)\sigma_i(x)\right) = O(1),$$

then $\tilde{V}(x) - \tilde{V}(\tilde{w}) = O(1)$. $\square$

3.7 Generating Multiple Cut

Recall that a cut at each iteration is generated from using subgradient information at the current iterate. Since the set of subgradients (subdifferential set) at the current iterate may have more than one elements, or multiple constraints may be violated at a given iterate, it may be possible to generate more than one cuts at the current iterate. It is therefore important for us to allow the possibility of adding multiple cuts at the cut addition step of a volumetric algorithm. Algorithm 1 needs a straight forward modification to allow for multiple cut addition in Step 3. It is however important that after adding multiple cuts, we can quickly recompute the approximate center in Step 5 of the algorithm. For this reason in this section we give a condition that guarantee that the number of iterates needed in Step 5 is of the same order ($O(\sqrt{m})$) as in the case of single cut addition. This condition is similar to the conditions in the cut translation situation of Section 3.6. The addition of multiple cuts appears to be more difficult in the central cut variant of VCM. This is because the central cut method requires generation of a "good new feasible solution" after adding cuts (see Anstreicher [5, Conclusions]). This is not needed in the near-central cut variant.

We assume that t new constraints are added and we let $\tilde{\mathcal{P}} \equiv \{x | x \in \mathcal{P}, a_{m+j}^T x \geq b_{m+j}, j = 1, \ldots t\}$ be the new region obtained after adding these t inequality to $\mathcal{P}$. Let

$$\tilde{A} = \begin{pmatrix} A \\ a_{m+1}^T \\ \vdots \\ a_{m+t}^T \end{pmatrix}, \quad \tilde{b} = \begin{pmatrix} b \\ b_{m+1} \\ \vdots \\ b_{m+t} \end{pmatrix}.$$

Note that $x \in \tilde{\mathcal{P}} \Rightarrow x \in \mathcal{P}$. For $x \in \tilde{\mathcal{P}}$, let $\tilde{s}(x) = \tilde{A}x - \tilde{b}$, and $\tilde{S}(x) = diag(\tilde{s}(x))$. Clearly, $\tilde{s}(x) = (s(x), \tilde{s}_{m+1}(x), \ldots, \tilde{s}_{m+t}(x))$, where $\tilde{s}_{m+j}(x) = a_{m+j}^T x - b_{m+j}$, $j = 1, \ldots, t$. Let

$$\tau_j(x) = \frac{a_{m+j}^T (A^T S^{-2}(x) A)^{-1} a_{m+j}}{\tilde{s}_{m+j}^2(x)} = \frac{a_{m+j}^T H(x)^{-1} a_{m+j}}{\tilde{s}_{m+j}^2(x)}, j = 1, \ldots t.$$

Let $\tilde{V}(.)$ be the volumetric barrier function for $\tilde{\mathcal{P}}$ and $\tilde{w}$ be its volumetric center. Let $\tilde{H}(x) = \tilde{A}^T\tilde{S}^{-2}(x)\tilde{A}$. The following theorem and its proof is similar to Theorem 17.

THEOREM 18 *Assume that t constraints are added as above, and let*

$$\sum_{i=1}^{t} \ln(1+\tau_i(x)) = O(1),$$

then $\tilde{V}(x) - \tilde{V}(\tilde{w}) = O(1)$.

Proof. Following the steps used to arrive at (41) we can see that $\tilde{V}(x) = V(x) + \frac{1}{2}ldet(M(x))$, where $M(x) \equiv I + \sum_{i=1}^{t}\frac{1}{\tilde{s}^2_{m+i}(x)}L^{-1}(x)a_{m+i}a_{m+i}^T L^{-T}(x)$, and $L(x)$ is a Cholesky factor of $H(x)$. Also using arguments similar to those used to arrive at (42) we can show that $\tilde{V}(x) - \tilde{V}(\tilde{w}) \le V(x) - V(w) + \frac{1}{2}ldet(M(x))$. Next, using Proposition 16, and $V(x) - V(w) \le 5 \times 10^{-7}$, we have

$$ldet(M(x)) \le \sum_{i=1}^{t} \ln\left(1 + \frac{1}{\tilde{s}^2_{m+i}(x)}a_{m+i}^T H^{-1}(x)a_{m+i}\right) = \sum_{i=1}^{t}\ln(1+\tau_i(x)) \quad \Box$$

4 An Algorithm for Two Stage Convex Stochastic Program

We now use the volumetric center algorithm of the previous section to solve two stage convex stochastic programs. The modifications are straight forward for the case where the number of scenarios are finite and the subgradient of $R(x)$ is calculated exactly. We cover this case in the next subsection. We then modify the algorithm using exact subgradients to an algorithm which calculates these subgradients approximately in Section 4.2.

4.1 Near-Central Cut Volumetric Algorithm for Two Stage Convex Stochastic Programs with Exact Subgradients

In this section we assume that an oracle can compute an exact subgradient of $c^i(.), i = 1, \dots m^1$ for all $x \in \mathcal{B}$. Another oracle can compute a subgradient of $R(.), \tilde{c}(.)$ for all $x \in \mathcal{C}^1$. It is easy to see [39] that a subgradient of $c(.)$ is available by adding the available subgradients of $\tilde{c}(.)$ and $R(.)$. For a given $\rho > 0$, let

$$\mathcal{C}_\rho \equiv \mathcal{C}^1 \cap \{x | c(x) \le c(x^*) + \rho\}.$$

The next well known proposition, which follows immediately from the definition of a subgradient, is used for generating cuts.

PROPOSITION 19 *Let $\bar{x} \in \mathcal{B}$, $\bar{x} \notin \mathcal{C}^1$, and assume that the ith inequality is violated. Let g^i be a subgradient of $c^i(x)$ at $\bar{x}$. Then,*

$$\mathcal{C}_\rho \subseteq \{x | {g^i}^T x \leq {g^i}^T \bar{x} + c^i(x) - c^i(\bar{x})\}.$$

Now let $\bar{x} \in \mathcal{C}^1$, $\bar{x} \notin \mathcal{C}_\rho$, and assume that g^0 is a subgradient of $c^0(x)$ at $\bar{x}$, then

$$\mathcal{C}_\rho \subseteq \{x | {g^0}^T x \leq {g^0}^T \bar{x} + c^0(x^*) + \rho - c^0(\bar{x})\},$$

where x^* is an optimal solution of TSSCP. $\square$

Since $c^0(x^*)$ and $c^i(x^*)$ are not known, the following weaker inequalities obtained from the above proposition are used:

$$ {g^i}^T x \leq {g^i}^T \bar{x} \quad \text{(feasibility cut)} \tag{43}$$
$$ {g^0}^T x \leq {g^0}^T \bar{x} \quad \text{(optimality cut).} \tag{44}$$

The inequality (43) is valid because $c^i(x^*) \leq 0$, and $c^i(\bar{x}) > 0$ since $\bar{x}$ is infeasible. Inequality (44) is valid because if $c^0(\bar{x}) \leq c^0(x^*) + \rho$, then we have a desired solution, otherwise, $c^0(x^*) + \rho - c^0(\bar{x}) < 0$. Since $c^i(\bar{x}) > 0$, the feasibility cut can be translated as the algorithm progresses. The optimality cut can be translated as better estimates of optimal objective value become available. The approach used for translating these cuts was discussed in Section 3.6.

We now modify the near-central cut VCM of Section 3.5 for TSSCP. We state the algorithm without cut translation and multiple cuts. These modifications can be easily incorporated in the algorithm.

Algorithm 2 (Near-Central Cut VCM for TSSCP).

Input. x^0, $\mathcal{P}^0$, m_0, L, $\sigma = .04, \hat{\delta} = .01, \tau = 4$

Step 1. (Termination check). If $V^k(x^k) \geq .7nL + n\ln(m_k)$, then STOP. Else go to Step 2.

Step 2. (Decide if we should add or drop a constraint) If $\sigma_{\min}(x^k) \geq \sigma$, go to Step 3, else go to Step 6.

Step 3. (Feasibility Test) Check if $x^k \in \mathcal{C}^1$. If no, go to Step 4a, otherwise go to Step 4b.

Step 4a. (Feasibility cut subgradient). Call the oracle which returns a vector $g^i \in \Re^n$ such that ${g^i}^T x \le {g^i}^T x^k$ for all $x \in \mathcal{C}^1$. Let $a = -g^i$ and go to Step 5.

Step 4b. (Optimality cut subgradient). Call the oracle which returns a vector $g^0 \in \Re^n$ such that ${g^0}^T x \le {g^0}^T x^k$ is satisfied by x^*. Let $a = -g^0$ and go to Step 5.

Step 5. (Add a cut). Let $s^k = A^k x^k - b^k$, $S^k = diag(s^k)$. Add the constraint $a^T x \ge \beta$ to the existing constraint system. Here $\beta = a^T x^k - (a^T({A^k}^T (S^k)^{-2} A^k)^{-1} a/\tau)^{1/2}$. Represent the new constraint system by (A^{k+1}, b^{k+1}), $m_k = m_k + 1$. Go to Step 7.

Step 6. (Delete a constraint). Suppose that $\sigma_{\min}(x^k) = \sigma_j(x^k) < \sigma$. Let (A^{k+1}, b^{k+1}) be the constraint system obtained by removing the jth row of (A^k, b^k), $m_k = m_k - 1$. Go to Step 7.

Step 7. (Centering steps). Let $\bar{x}^0 = x^k$. Starting from $\bar{x}^0$ take a sequence of damped Newton-like steps of the form $\bar{x}^{j+1} = \bar{x}^j - \alpha Q^{-1}(\bar{x}^j) g(\bar{x}^j)$, $j \ge 0$, until $\mu(\bar{x}^J)\|g(\bar{x}^J)\|_{Q^{-1}(\bar{x}^J)} \le \hat{\delta}/6$. Let $x^{k+1} = \bar{x}^J$, $k = k+1$, and go to Step 1

It is straight forward to see that an analogue of Lemma 14 and Theorem 15 is also true for Algorithm 2. The following theorem follows from these results.

THEOREM 20 *Let parameters for Algorithm 2 be chosen as in Theorem 15 ($\sigma = .04, \tau = 4, \hat{\delta} = .01$). Algorithm 2 either finds a point in $\mathcal{C}_\rho$ or it proves that the volume of $\mathcal{C}_\rho$ is smaller than that of a $n-$dimensional sphere of radius 2^{-L}. The overall complexity of the algorithm is $O(n(L+\hat{L})KC + n^{4.5}(L+\hat{L}))$, where C is the cost of solving an instance of second stage problem, and K is the number of scenarios.*

Proof. The feasibility cuts do not cut away a point in $\mathcal{C}_\rho$. The only way an optimality cut can cut away a point in $\mathcal{C}_\rho$ is if it is generated at a feasible point where the objective value is lower than $c(x^*) + \rho$, in which case we have found a desired point. Now assume that all cuts are generated at points that are not in $\mathcal{C}_\rho$, in which case they are valid for $\mathcal{C}_\rho$. From Lemma 14 we have that at termination the volume of $\mathcal{C}_\rho$ be smaller than that of a $n-$dimensional sphere of radius 2^{-L}. $\square$

Theorem 20 states that Algorithm 2 correctly solves the problem if $\mathcal{C}_\rho$ contains a $n-$ dimensional ball of radius 2^{-L}. For proper choices of ρ and L such an assumption is justified if the set $\mathcal{C}^1$ has a non-empty interior. As discussed in Section 2.2, this can be ensured by introducing an artificial variable with a large unknown cost. In practice we guess this large cost. The cost is increased it if the artificial variable is not sufficiently small at the solution available at termination.

4.2 Algorithm for Two Stage Convex Stochastic Programs Using Sampling

As discussed in Section 2.3 for many practical problems either the number of scenarios is too large or the probability space is continuous. In these situations computation of exact subgradient is not practical, and we need to resort to Monte-Carlo simulation. For developing an algorithm the natural idea is to replace the exact subgradient with the subgradient computed through simulation when computing optimality cuts in Algorithm 2. Since we can not compute a subgradient exactly, we relax the optimality requirements by requiring a solution of desired accuracy with any desirable probability, but not with probability one. The analysis of this section gives two ways to accomplish this: (i) a probability arbitrarily close to one is achieved in one single run of the algorithm, (ii) the algorithm is run repeatedly from randomly (independently) generated starting points, with each run having a positive probability of producing a solution with desirable accuracy (see Remark 3 below).

Recall that when an optimality cut is generated in Algorithm 2, instead of passing the cut through the point at which it is generated, it is made weaker. In particular, at a point $\bar{x}$ at which the cut is generated, instead of adding a constraint $a^T x \geq a^T \bar{x}$ in Step 5 we added $a^T x \geq \beta$, where $\beta = a^T \bar{x} - (a^T(A^T S(\bar{x})^{-2} A)^{-1} a/\tau)^{1/2}$. We use this property of the algorithm with exact subgradients to develop an algorithm with approximate subgradients computed by sampling. Now assume that a subgradient was computed approximately, so instead of g^0, we have an estimate $\bar{g}$, and $g^0 = \bar{g} + \epsilon$, for some $\epsilon \in \Re^n$. In our context the estimate $\bar{g}$ is obtained from Monte-Carlo simulation. Clearly, $(\bar{g} + \epsilon)^T x \leq (\bar{g} + \epsilon)^T \bar{x}$ is a valid cut, i.e.,

$$\bar{g}^T x \leq \bar{g}^T \bar{x} + \epsilon^T (\bar{x} - x)$$

is satisfied by all $\hat{x} \in \mathcal{C}_\rho$, unless a point in $\mathcal{C}_\rho$ is already found. This means that the cut

$$\bar{g}^T x \leq \bar{g}^T \bar{x} + (\bar{g}^T (A^T S(\bar{x})^{-2} A)^{-1} \bar{g}/\tau)^{1/2}$$

added in Step 5 of Algorithm 2 does not cut away any point in $\mathcal{C}_\rho$ as long as

$$\max_{\hat{x} \in \mathcal{C}_\rho} \epsilon^T (\bar{x} - \hat{x}) \leq (\bar{g}^T (A^T S(\bar{x})^{-2} A)^{-1} \bar{g}/\tau)^{1/2}. \tag{45}$$

We would like to know the probability with which (45) is satisfied as $\bar{g}$ is obtained with increased sample size. We obtain this probability using the large deviation principle.

Let $\zeta^i, i = 1, \ldots N$ be independent and identically distributed observations of a random variable $\tilde{\zeta}$. Assume that $|\tilde{\zeta}| \leq \nu$, and $E[\tilde{\zeta}] = 0$. Let $\zeta_N = \frac{1}{N} \sum_{i=1}^N \zeta^i$, and observe that $\lim_{N \to \infty} \zeta_N = 0$. The next lemma shows a bound on $Prob(\zeta_N \geq \theta)$ as an exponential function of N.

Lemma 21 *Let ζ_N be the sample mean of a random variable $\tilde{\zeta}$ using N samples, and $|\tilde{\zeta}| \leq \nu$, then*

$$Prob(\zeta_N \geq \theta) \leq e^{\frac{-N\theta^2}{2\nu^2}}.$$

Proof. Let $\mathbf{1}_{\zeta_N - \theta \geq 0}$ be an indicator function which is 1 if $\zeta_N - \theta \geq 0$ and zero otherwise. For any $\lambda \geq 0$ and $\theta > 0$, following the steps of the proof of Theorem 2.2.3 in Dembo and Zeitouni [26] and Shapiro and Homem-de-Mello [65] we have,

$$\begin{aligned} Prob(\zeta_N \geq \theta) &= E[\mathbf{1}_{\zeta_N - \theta \geq 0}] \\ &\leq E[e^{N\lambda(\zeta_N - \theta)}] \\ &= e^{-N\lambda\theta} \prod_{i=1}^{N} E[e^{\lambda\zeta^i}] \\ &= e^{-N[\lambda\theta - \Lambda(\lambda)]}, \end{aligned}$$

where $\Lambda(\lambda) \equiv \ln E[e^{\lambda\zeta}]$. The inequality above is a Chebycheff's inequality, and the second equality above uses independence of ζ^i. Note that $\Lambda(\lambda)$ is the log of moment generating function of ζ.

Clearly, $\Lambda(0) = 0$, and

$$\Lambda'(\lambda) = \frac{E[\zeta e^{\zeta\lambda}]}{E[e^{\zeta\lambda}]} \implies \Lambda'(0) = E[\zeta] = 0,$$

and

$$|\Lambda''(\lambda)| = \left| \frac{E[\zeta^2 e^{\lambda\zeta}]}{E[e^{\lambda\zeta}]} - \left(\frac{E[\zeta e^{\lambda\zeta}]}{E[e^{\lambda\zeta}]} \right)^2 \right| \leq \nu^2.$$

Since for any $\lambda > 0$,

$$\Lambda(\lambda) = \Lambda(0) + \lambda\Lambda'(0) + \frac{\lambda^2}{2}\Lambda''(\alpha\lambda), \text{ for some } \alpha, 0 \leq \alpha \leq 1,$$

hence we have $\Lambda(\lambda) \leq \frac{\lambda^2\nu^2}{2}$. Hence by taking $\lambda = \theta/\nu^2$, we have the desired result. □

We now work towards generating a bound for the right and left hand side in (45). The following proposition says that if the difference between the objective value at the current iterate, $\bar{x}$, and the optimal objective value is large, then a subgradient at $\bar{x}$ should be sufficiently large in magnitude.

Proposition 22 *Let $\bar{x} \in \mathcal{C}^1$, $\bar{x} \notin \mathcal{C}_\rho$, and $x^* \in \mathcal{C}^*$. Let g be a subgradient of $c^0(.)$ at $\bar{x}$, and let $g = \bar{g} + \epsilon$, where $\bar{g}$ is an estimate of g. Then,*

$$\|\bar{g}\|_{\bar{H}^{-1}} \geq \frac{\rho - \epsilon^T(\bar{x} - x^*)}{\|x^* - \bar{x}\|_{\bar{H}}}, \tag{46}$$

where $\bar{H} = A^T S(\bar{x})^{-2} A$.

Proof. Since $c(.)$ is a convex function, for any $\bar{x} \in \mathcal{C}^1$, and $\bar{x} \notin \mathcal{C}_\rho$ and $x^* \in \mathcal{C}^*$ we have

$$\rho \leq c(\bar{x}) - c(x^*) \leq -g^T(x^* - \bar{x}) = -(\bar{g} + \epsilon)^T(x^* - \bar{x}) \leq \|\bar{g}\|_{\bar{H}^{-1}} \|x^* - \bar{x}\|_{\bar{H}} - \epsilon^T(x^* - \bar{x}),$$

hence the inequality in (46) follows. $\square$

We are now in a position to prove the following result, which shows that the probability of not cutting away the set $\mathcal{C}_\rho$ can be made arbitrarily small when using a cut generated from sampling.

LEMMA 23 *Assume that the subgradient is estimated by taking the sample mean of N samples. The probability of $\mathcal{C}_\rho \in \mathcal{P}^{k+1}$ after adding a cut in Step 4b is given by*

$$Prob(\mathcal{C}_\rho \in \mathcal{P}^{k+1}) \geq (1 - e^{-N\psi + ln(2n)}) Prob(\mathcal{C}_\rho \in \mathcal{P}^k),$$

where

$$\psi \equiv \frac{\rho^2}{1.2 \times 10^6 n^7 2^{2L} \nu^2}. \tag{47}$$

Proof. Let $\hat{z} = \max_{\hat{x} \in \mathcal{C}_\rho} \epsilon^T(\bar{x} - \hat{x})$. For $\bar{x} \in \mathcal{C}^1$, and $\bar{x} \notin \mathcal{C}_\rho$, from Proposition 22 we have

$$\begin{aligned}
Prob\left(\hat{z} \leq \frac{\|\bar{g}\|_{\bar{H}^{-1}}}{\tau^{1/2}}\right) &\geq Prob\left(\hat{z} \leq \frac{\rho - \epsilon^T(\bar{x} - x^*)}{\tau^{1/2}\|\bar{x} - x^*\|_{\bar{H}}}\right) \\
&= Prob\left(\hat{z} + \frac{\epsilon^T(\bar{x} - x^*)}{\tau^{1/2}\|\bar{x} - x^*\|_{\bar{H}}} \leq \frac{\rho}{\tau^{1/2}\|\bar{x} - x^*\|_{\bar{H}}}\right) \\
&\geq Prob\left(\hat{z} \leq \frac{\rho}{1 + \tau^{1/2}\|\bar{x} - x^*\|_{\bar{H}}}\right),
\end{aligned}$$

where the last inequality uses that $\hat{z} \geq \epsilon^T(\bar{x} - x^*)$, since $x^* \in \mathcal{C}_\rho$. For $\bar{x}$ as in Lemma 7 (near-center point at termination in Step 7 of Algorithm 2), if $x^* \in \mathcal{P}^k$, then $\|\bar{x} - x^*\|_{\bar{H}} \leq 26.5n$, and $1 + \tau^{1/2}\|\bar{x} - x^*\|_{\bar{H}} \leq 54n$. Also if $\mathcal{C}_\rho \in \mathcal{P}^k$, then for any $\hat{x} \in \mathcal{C}_\rho$, $\epsilon^T(\bar{x} - \hat{x}) \leq \|\epsilon\|_{\bar{Q}^{-1}} \|\bar{x} - \hat{x}\|_{\bar{Q}} \leq 5.3n\|\epsilon\|_{\bar{Q}^{-1}}$, hence $\hat{z} = \max_{\hat{x} \in \mathcal{C}_\rho} \epsilon^T(\bar{x} - \hat{x}) \leq 5.3n\|\epsilon\|_{\bar{Q}^{-1}}$. Furthermore, from Corollary **??** we have $\hat{z} \leq (5.3n)^2 2^L \|\epsilon\|$. Hence we have,

$$\begin{aligned}
Prob(\mathcal{C}_\rho \subseteq \mathcal{P}^{k+1}) &= Prob(\mathcal{C}_\rho \subseteq \mathcal{P}^{k+1} | \mathcal{C}_\rho \subseteq \mathcal{P}^k) Prob(\mathcal{C}_\rho \subseteq \mathcal{P}^k) \\
&\geq Prob\left(\hat{z} \leq \frac{\|\bar{g}\|}{\tau^{1/2}} | \mathcal{C}_\rho \subseteq \mathcal{P}^k\right) Prob(\mathcal{C}_\rho \subseteq \mathcal{P}^k) \\
&\geq Prob(\hat{z} \leq \frac{\rho}{54n}) Prob(\mathcal{C}_\rho \subseteq \mathcal{P}^k) \\
&\geq Prob(\|\epsilon\| \leq \frac{\rho}{1525n^3 2^L}) Prob(\mathcal{C}_\rho \subseteq \mathcal{P}^k).
\end{aligned}$$

Let $\rho' = \frac{\rho}{1525n^3 2^L}$. Clearly the event $\|\epsilon\| \geq \rho'$ implies the event $|\epsilon_i| \geq \rho'/\sqrt{n}$ for some i. Hence, $Prob(\|\epsilon\| \geq \rho') \leq nProb(|\epsilon_i| \geq \rho'/\sqrt{n}) \leq 2ne^{\frac{-N\rho'^2}{2n\nu^2}}$, where the bound in the last inequality follows from Lemma 21 and observing that the random vector ϵ is bounded, $\epsilon_N = g^0 - \sum_{i=1}^N g^i \to 0$ as $N \to \infty$ and $E[\epsilon] = 0$. Here g^i is a sampled subgradient, and without loss of generality we have taken ν to be the bound on all $|\epsilon_i|$. $\square$

The following corollary follows immediately from Lemma 23.

Corollary 24 *After k iterations of Algorithm 2 using subgradients estimated from sampling, either an $x \in \mathcal{C}_\rho$ has been found, or*

$$Prob(\mathcal{C}_\rho \subseteq \mathcal{P}^k) \geq 1 - e^{-N\psi + ln(2nk)}$$

Proof. In k iterations of Algorithm 2 at most k cuts are added. From Lemma 23, we have

$$Prob(\mathcal{C}_\rho \subseteq \mathcal{P}^k) \geq (1 - e^{-N\psi + ln(2n)})^k \geq 1 - ke^{-N\psi + ln(2n)} = 1 - e^{-N\psi + ln(2nk)}. \ \square$$

We now have the following theorem regarding the convergence of Algorithm 2 using sampled subgradients.

Theorem 25 *Let parameters for Algorithm 2 be chosen as in Theorem 15 ($\sigma = .04, \tau = 4, \hat{\delta} = .01$). Assume that an estimate of a subgradient of $c^0(.)$ is obtained by using $N = \frac{\mu + \ln(2n^2(L+\hat{L})) - ln(1-\wp)}{\psi}$, ($\psi$ defined in (47) and μ is log of constant in $O(n(L+\hat{L}))$) samples at each iteration to generate a cut in Step 4b of Algorithm 2. Then, with probability greater than $\wp$ Algorithm 2 finds a point in $\mathcal{C}_\rho$ or it proves that the volume of $\mathcal{C}_\rho$ is smaller than that of a $n-$dimensional sphere of radius 2^{-L} in $O(n(L+\hat{L}))$ iterations. The overall complexity of the algorithm is $O(n(L+\hat{L})NC + n^{4.5}(L+\hat{L}))$, where C is the cost of solving an instance of second stage problem.*

Furthermore, if N processors are used and each second stage problem can be solved in polynomial time (in n, L, $\hat{L}$), then the two stage stochastic program can be solved in time which is polynomial in $n, L, \hat{L}, \ln(\nu)$ and $\ln(\wp)$.

Proof. The proof of the first part of Theorem 25 follows from Corollary 24. To see the second part it is sufficient to observe that each of the N processors can be used to generate an observation of the subgradient. Using these processors the sum $\frac{1}{N}\sum_{i=1}^N g^i$ can be computed in $O(\ln N)$ steps (see [12, Section 1.2]). $\square$

Remark 1 (Measure of Problem Difficulty). While finding the number of samples N to ensure that $\mathcal{C}_\rho \subseteq \mathcal{P}^k$ at termination with probability $\wp$, we have a factor $\ln(O(n(L+\hat{L}))) = \mu + \ln(n + (L+\hat{L}))$ from the total number of iterations after with the algorithm is stopped. We expect that the number of iterations after which the algorithm is stopped will be much smaller than $O(n(L+\hat{L}))$, most likely $O(n)$ or smaller. Also for most practical situations we expect that ρ and 2^L will be a constant, as 3 to 6 digits of accuracy in the solution will be sufficient. Similarly $\wp$ will also be a constant, i.e., for most practical problems we will be required to have a solution say with probability $\wp = .999$. This implies that the computational difficulty in solving TSSCP will largely depend on the value of ν.

Remark 2 (Sample Size in Practice). The parameter ν is difficult to estimate in advance, and the analysis above does not suggest a practical value of sample size. However, in practice we can estimate $Prob(\hat{z} \leq \frac{\|\bar{g}\|_{\bar{H}^{-1}}}{\tau^{1/2}})$, using Monte-Carlo simulation as follows. The constant $\|\bar{g}\|_{\bar{H}^{-1}}/\tau^{1/2}$ can be computed directly in the implementation, and we need not use a bound as used in the analysis. After a constant number of samples, ϵ will have a near multi-variate normal distribution whose co-variance matrix can be estimated. Using this distribution we can generate instances of ϵ, say ϵ^i, and solve $\max_{x\in\mathcal{P}^k} {\epsilon^i}^T x$. The desired probability is estimated by recording the number of instances that satisfy (45) and dividing it by the total number of instances generated.

Remark 3 (Computing Environment). It is possible to have a computing environment having clusters of processors, where the cost of communicating among processors in a cluster is small compared to cost of communicating across clusters. For example, we may have separate clusters of processors available at two geographically distant locations, where the cost of communicating over network between these two locations is large. In this situation the analysis suggests an alternative implementation strategy. Instead of making the probability of $\mathcal{C}_\rho \subseteq \mathcal{P}^k$ large by using large number of scenarios while generating cut at each iteration of near-central cut VCM, we may generate cuts ensuring $\mathcal{C}_\rho \in \mathcal{P}^k$ with smaller probability. Next we can independently solve our problem a fixed number of times, making large the probability that one of these runs give the desired solution. In particular, assume that a particular run of our algorithm ensures that $x \in \mathcal{C}_\rho$ is found with probability $\bar{\wp}$ and the desirable probability is $\wp, \bar{\wp} < \wp$. Then, after $\lceil \frac{\ln(1-\wp)}{\ln(1-\bar{\wp})} \rceil$ independent runs we will have the desired solution with probability $\wp$ in at least one of the runs.

5 Algorithm for General Convex Stochastic Program

We now apply the ideas of previous section to develop an algorithm for general convex stochastic programs. For $\bar{x} \notin X$ let $a_X^T x \geq \beta$ represent an inequality that is generated so that $X \subseteq \{x | a_X^T x \geq \rho\}$. Also, for a given $\bar{x} \in X$, we assume that a subgradient of $c^i(x)$ is estimated with increasing accuracy using sampling. A subgradient of $c^i(x)$ is represented by g^i and its estimate by $\bar{g}^i$. The following algorithm is a modification of Algorithm 2 where subgradient is estimated to generate feasibility and optimality cuts.

Algorithm 3 (Near-Central Cut VCM for SCP).

Input. x^0, $\mathcal{P}^0$, m_0, L, $\sigma = .04, \hat{\delta} = .01, \tau = 4$.

Step 1. (Termination check). If $V^k(x^k) \geq .7nL + n\ln(m_k)$, then STOP. Else go to Step 2.

Step 2. (Decide if we should add or drop a constraint). If $\sigma_{\min}(x^k) \geq \sigma$, go to Step 3, else go to Step 6.

Step 3. (Feasibility tests). Check if $x^k \in X$. If no, go to Step 4a. If yes, find if any of the constraints $c^i(x) \leq 0, i = 1, \ldots m$ is violated at x^k by checking $\bar{c}^i(x^k) \leq 0$, where $\bar{c}(.)$ represents an estimate of $c(.)$ generated by using Monte-Carlo simulation. If $\bar{c}^i(x^k) \leq 0$ for $i = 1, \ldots m$, go to Step 4b, otherwise go to Step 4c.

Step 4a. (Compute subgradient for feasibility cut). Call an oracle which returns a vector $g_X \in \Re^n$ such that ${g_X}^T x \leq {g_X}^T x^k$ for all $x \in X$. Let $\bar{a} = -g_X$ and go to Step 5.

Step 4b. (Compute subgradient for optimality cut). Call the oracle which returns an estimate, $\bar{g}^0$, of subgradient vector $g^0 \in \Re^n$. Let $\bar{a} = -\bar{g}^0$ and go to Step 5.

Step 4c. (Compute subgradient for expected value constraint). Let $\bar{g}^i$ be an estimate of subgradient vector $g^i \in \Re^n$ of the constraint satisfying $\bar{c}^i(x^k) > 0$. Let $\bar{a} = -\bar{g}^i$ and go to Step 5

Step 5. (Add a cut). Let $s^k = A^k x^k - b^k$, $S^k = diag(s^k)$. Add the constraint $\bar{a}^T x \geq \beta$ to the existing constraint system. Here $\beta = \bar{a}^T x^k - (\bar{a}^T ({A^k}^T (S^k)^{-2} A^k)^{-1} \bar{a}/\tau)^{1/2}$. Represent the new constraint system by (A^{k+1}, b^{k+1}), $m_k = m_k + 1$. Go to Step 7.

Step 6. (Delete a constraint). Suppose that $\sigma_{\min}(x^k) = \sigma_j(x^k) < \sigma$. Let (A^{k+1}, b^{k+1}) be the constraint system obtained by removing the jth row of (A^k, b^k), $m_k = m_k - 1$. Go to Step 7.

Step 7. (Centering steps). Let $\bar{x}^0 = x^k$. Starting from $\bar{x}^0$ take a sequence of damped Newton-like steps of the form $\bar{x}^{j+1} = \bar{x}^j - \alpha Q^{-1}(\bar{x}^j) g(\bar{x}^j)$, $j \geq 0$, until $\mu(\bar{x}^J) \|g(\bar{x}^J)\|_{Q^{-1}(\bar{x}^J)} \leq \hat{\delta}/6$. Let $x^{k+1} = \bar{x}^J$, $k = k + 1$, and go to Step 1

The analysis of Algorithm 3 is similar to the analysis in Section 4.2, except for Step 4c followed by Step 5. In this case we need to account for the possibility of error in estimating $c^i(.)$ together with the error in its subgradient estimate. Below we show how this can be accomplished. Let g^i be an exact subgradient of $c^i(x)$ at $\bar{x}$. Recall from Proposition 19 that

$$g^{i^T} x \leq g^{i^T} \bar{x} + c^i(x) - c^i(\bar{x})$$

is a valid inequality. Let $g^i = \bar{g}^i + \epsilon^i$ and $c^i(\bar{x}) = \bar{c}^i(\bar{x}) + \epsilon_c^i$, where ϵ^i is error in estimation of g^i and $\epsilon_c^i = c^i(x) - \bar{c}^i(x)$ is error in the estimation of constraint function value. Since for $\hat{x} \in \mathcal{C}_\rho$, $c^i(\hat{x}) \leq 0$,

$$\bar{g}^{i^T} x \geq \bar{g}^{i^T} \bar{x} + \epsilon^{i^T}(\bar{x} - x^*) - \epsilon_c^i$$

gives a valid cut. We add the feasibility cut if $\bar{c}^i(\bar{x}) > 0$. This means that in this case the cut added in Step 5 is valid as long as

$$\max_{\hat{x} \in \mathcal{C}_\rho} \epsilon^{i^T}(\bar{x} - \hat{x}) - \epsilon_c^i \leq (\bar{g}^{i^T}(A^T S(\bar{x})^{-2} A)^{-1} \bar{g}^i / \tau)^{1/2}. \tag{48}$$

We can now take our error vector (ϵ) to be $\begin{pmatrix} \epsilon^i \\ \epsilon_c^i \end{pmatrix}$ and perform an analysis similar to that in Section 4.2. A theorem similar to Theorem 25 can be stated for Algorithm 3. We leave this to the reader.

6 Conclusions

We developed a variant of Vaidya's volumetric center cutting plane method that is suitable for stochastic convex programming problems where the subgradient to generate a cut is computed using sampling. For this variant we showed how multiple cuts and bulk cut translation can be done. We showed how a subgradient used to generate cuts in our algorithm is computed for the two-stage stochastic convex program. For the two-stage and general stochastic convex programming problem we showed that the proposed variant ensures certain performance guarantees. In particular, we provided an estimate of the sample size needed to generate a cut ensuring that the near-central cut variant of VCM will give an optimal solution of the stochastic convex program with any desirable probability. It is also possible to analyze the cutting plane algorithm using the analytic centers instead of the volumetric centers. The computations at each iteration in the analytic center approach are simpler, however, in the worst case analysis the algorithm requires $O(nL^2)$ calls to the oracle [29], as compared with $O(nL)$ calls for the volumetric center method. The practical evaluation of the two approaches, and their overall efficiency require a computational study, which we are currently undertaking.

7 Acknowledgements

The author will like to thank Professor Ajit Tamhane at Northwestern University for a discussion on sampling techniques. He would like to thank Jeff Linderoth at Argonne National Lab for bringing reference [65] to his attention. He would like to thank Muhittin Ozevin Gokhan at Northwestern University for discussions that motivated the development of this paper.

References

[1] S. Ahmed and A. Shapiro (2002), "The sample average approximation method for stocahstic programs with integer recourse," technical report, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332-0205.

[2] Andersen, E.D., J.E. Mitchell, C. Roos, and T. Terlaky (1999), "A Homogenized Cutting Plane Method To Solve The Convex Feasibility Problem", in: Optimization Methods and Applications, publisher: Kluwer.

[3] K. M. Anstreicher (1996), "Large step volumetric potential reduction algorithm for linear programming", *Annals of Operations Research*, **62**, 521-538.

[4] K. M. Anstreicher (1997), "On Vaidya's Volumetric Cutting Plane Method For Convex Programming", *Mathematics of Operations Research*, **22**(1), 63-89.

[5] K. M. Anstreicher (1998), "Towards A Practical Volumetric Cutting Plane Method for Convex Programming", *SIAM Journal of Optimization*, **9**(1), 190-206.

[6] K. M. Anstreicher (1999), "Ellipsoidal Approximations of Convex Sets Based On The Volumetric Barrier", *Mathematics of Operations Research*, **24**(1), 193-203.

[7] K. A. Ariyawansa and P.L. Jiang (1996), "Polynomial Cutting Plane Algorithms for Two-Stage Stochastic Linear Programs Based on Ellipsoids, Volumetric Centers and Analytic Centers", Technical Report, Department of Pure and Applied Mathematics, Washington State University, Pullman, WA 99164-3113.

[8] Atkinson, D.S. and P.M. Vaidya (1995), "A Cutting Plane Algorithm for Convex Programming that uses Analytic Centers", *Mathematical Programming*, **69**, 1-43.

[9] K. T. Au, J. L. Higle and S. Sen, "Inexact subgradient methods with applications in stochastic programming", *Mathematical Programming*, **63**, 65-82.

[10] E. M. L. Beale, "On minimizing a convex function subject to linear inequalities", *Journal of the Royal Statistical Society*, B47, 173-184.

[11] A. J. Berger, J. M. Mulvey, A. Ruszczyński (1994), "An Extension of the DQA Algorithm for Convex Stochastic Programs", *SIAM J. Optimization*, **4**(4), 735-753.

[12] D. P. Bertsekas and J. N. Tsitsiklis (1989), "Parallel and Distributed Computation", publisher: Prentice Hall.

[13] J. R. Birge, "Decomposition and partitioning methods for stochastic linear programs", *Operations Research*, **33**, 989-1007.

[14] J. R. Birge, "Stochastic Programming Computation and Applications", *INFORMS Journal on Computing*, **9**(2), 111-133.

[15] J. R. Birge and L. Qi (1995), "Subdifferential Convergence in Stochastic Programs", *SIAM J. Optimization*, **5**(2), 436-453.

[16] J. R. Birge and L. Qi (1988), "Computing Block-Angular Karmarkar Projections with Applications to Stochastic Programming", *Management Science*, **56**, 15-38.

[17] J. R. Birge, L. Qi, and Z. Wei (1998), "Convergence Analysis of Some Methods for Minimizing a Non-smooth Convex Function", *Journal of Optimization Theory and Applications*, **97**(2), 357-383.

[18] R. G. Bland, D. Goldfarb, and M.J. Todd (1981), "The ellipsoid method: A survey", Operations Research, **29**, 1039-1091.

[19] T.J. Carpenter, I.J. Lustig and J.M. Mulvey (1991), "Formulating Two-Stage Stochastic Programs for Interior Point Methods", *Operations Research*, **39**, 757-770.

[20] B.J. Chun and S.M. Robinson (1995), "Scenario analysis via bundle decomposition", *Annals of Operations Research*, **56**, 39-63.

[21] J. Czyzyk, R. Fourer and S. Mehrotra (1995). "A Study of the Augmented-System and Column-Splitting Approaches for Solving Two-Stage Stochastic Linear Programs by Interior-Point Methods," *ORSA Journal on Computing,* 474-490.

[22] J. Czyzyk, R. Fourer, S. Mehrotra (1995), "Parallel solutions of multi-stage stochastic linear programs by interior-point methods", Report 95-69, Dept. of IE/MS, Northwestern University.

[23] G. B. Dantzig (1955), "Linear Programming Under Uncertainty", *Management Science*, **1** (3), 197-206.

[24] G. B. Dantzig and P. Glynn (1990), "Parallel Processor for Planning Under Uncertainty", *Annals of Operations Research*, **22**, 1-21.

[25] G. B. Dantzig and G. Infanger (1991), "Large-Scale Stochastic Linear Programs-Importance Sampling and Benders Decomposition", in: *Computational and Applied Mathematics*, editors: C. Brezinski and U. Kulisch, publisher: North Holland, 111-120.

[26] A. Dembo and O. Zeitouni (1998), "Large Deviations Techniques and Applications", second edition, publisher: Springer.

[27] Y. Ermoliev (1988), "Stochastic quasigradient methods", in: Numerical Techniques for Stochastic Optimization, editors: Y. Ermoliev and R. J.-B. Wets, publisher: Springer.

[28] O. Güler (1994), "New Proximal Point Algorithms for Convex Minimization", *SIAM Journal on Optimization*, **4**, 649-664.

[29] J.-L. Goffin, Z.-Q Luo and Y. Ye (1996), "Complexity Analysis Of An Interior Cutting Plane Method For Convex Feasibility Problems", *SIAM Journal on Optimization*, **6**(3), 638-652.

[30] J. L. Higle and S. Sen (1996), "Stochastic Decomposition: An Algorithm for Two Stage Linear Programs with Recourse", *Mathematics of Operations Research*, **16**, 650-669.

[31] J. L. Higle and S. Sen (1996), "Stochastic Decomposition: A Statistical Method for Large Scale Stochastic Linear Programming", publisher: Kluwer.

[32] J. L. Higle and S. Sen (1996), "Duality and statistical tests of optimality for two stage stochastic programs", *Mathematical Programming*, **75**, 257-275.

[33] J. L. Higle and S. Sen (1999), "Statistical approximations for stochastic linear programming problems", *Annals of Operations Research*, **85**, 173-192.

[34] J.-B. Hiriart-Urruty (1976), "About properties of the mean value functional and the continuous inf-convolution in stochastic convex analysis", In *Optimization Techniques Modeling and Optimization in the Service of Man,* Ed. J. Cea, Springer-Verlag Lecture Notes in Computer Science, Berlin, 763-789.

[35] J.-B. Hiriart-Urruty and C. Lemaréchal (1993), "Convex Analysis and Minimization Algorithms I,II", publisher: springer.

[36] Hoffmann-Jorgensen, J. (1994), "Probability With A View Towards Statistics", publisher: Chapmann and Hall.

[37] Horn, R.A. and C.R. Johnson (1985), *Matrix Analysis*, Cambridge University Press, Cambridge.

[38] Infanger, G. (1994), "PLANNING UNDER UNCERTAINTY Solving large-Scale Stochastic Linear Programs", publisher: boyd and fraser.

[39] A. D. Ioffe and V. L. Levin (1972), "Subdifferentials of Convex Functions", *Trudy Moskov. Mat. Obšč.*, (Trans. Moscow Mathematical Society), **26**, 1-72.

[40] Y. M. Kaniovski, A. J. King, and R. J.-B. Wets (1995), "Probabilistic bounds (via large deviations) for the solutions of stochastic programming problems", *Annals of Operations Research*, **56**, 189-208.

[41] A. J. King and R.T. Rockafellar (1993), "Asymptotic theory for solutions in statistical estimation and stochastic programming", *Mathematics of Operations Research*, **18**, 148-162.

[42] J. T. Linderoth, A. Shapiro and S. J. Wright (2002), "The empirical behavior of sampling methods for stochastic programs," OTC Report 02-01, Computer Science Dept., University of Wisconsin-Madison.

[43] A. Madansky (1960), "Inequalities for Stochastic Linear Programming", *Management Science*, **6**, 197-204.

[44] A. Madansky (1963), "Dual variables in two-stage linear programming under uncertainty", *J. Math. Anal. Appl.*, **6**, 98-108.

[45] R. Mifflin (1996), "A quasi-second-order proximal bundle algorithm", *Mathematical Programming*, **73**, 51-72.

[46] J. M. Mulvey and A. Ruszczyński, "A diagonal quadratic approximation method for large scale linear programs", *Operations Research Letters*, **12**, 205-215.

[47] Y. Nesterov and J.-PH. Vial (1999), "Homogeneous Analytic Center Cutting Plane Methods For Convex Problems And Variational Inequalities", *SIAM Journal on Optimization*, **9**(3), 707-728.

[48] S. Ramaswamy and J.E. Mitchell (1995), "A Long Step Cutting Plane Algorithm That Uses the Volumetric Barrier", *Technical Report*, Department of Mathematical Sciences, Rensselaer Polytechnic Institute.

[49] R. T. Rockafellar (1968), "Integrals Which Are Convex Functionals", *Pacific Journal of Mathematics*, **24**(3), 525-539.

[50] R. T. Rockafellar (1969), "Measurable Dependence of Convex Sets and Functions on Parameters", *Journal of Mathematical Analysis and Applications*, **28**, 4-25.

[51] R. T. Rockafellar (1970), "Convex Analysis," publisher: Princeton University Press.

[52] R. T. Rockafellar (1971), "Integrals Which Are Convex Functionals, II", *Pacific Journal of Mathematics*, **39**(2), 439-469.

[53] R. T. Rockafellar and R. J.-B. Wets (1975), "Stochastic Convex Programming: Kuhn-Tucker Conditions", *Journal of Mathematical Economics*, **2**, 349-370.

[54] R. T. Rockafellar and R. J.-B. Wets (1976), "Stochastic Convex Programming: Basic Duality", *Pacific Journal of Mathematics*, **62(1)**, 173-195.

[55] R. T. Rockafellar and R. J.-B. Wets (1976), "Stochastic Convex Programming: Relatively Complete Recourse And Induced Feasibility", *SIAM J. Control and Optimization*, **14(3)**, 574-589.

[56] R. T. Rockafellar and R. J.-B. Wets (1982), "On the Interchange of Subdifferentiation and Conditional Expectation for Convex Functionals", *Stochastics*, **7**, 173-182.

[57] R. T. Rockafellar and R. J.-B. Wets (1986), "A Lagrangian Finite Generation Technique for Solving Linear-Quadratic Problems in Stochastic Programing", *Mathematical Programming Study*, **28**, 63-93.

[58] R. T. Rockafellar and R. J.-B. Wets (1991), "Scenarios and policy aggregation in optimization under uncertainty", *Mathematics of Operations Research*, **16**, 1-23.

[59] A. Ruszczyński (1986), "A regularized decomposition method for minimizing a sum of polyhedral functions", *Mathematical Programming*, **35**, 309-333.

[60] A. Ruszczyński (1999), "Some advances in decomposition methods for stochastic linear programming", *Annals of Operations Research*, **85**, 153-172.

[61] A. Ruszczyński and W. Syski (1986), “A Method of Aggregate Stochastic Subgradients with On-Line Stepsize Rules for Convex Stochastic Programming Problems”, *Mathematical Programming Study*, **28**, 113-131.

[62] A. Shapiro (1993), “Asymptotic behavior of optimal solutions in stochastic programming”, *Mathematics of Operations Research*, **18**, 829-845.

[63] A. Shapiro (2000), “Stochastic programmming by monte-carlo simulation method,” technical report, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332-0205.

[64] A. Shapiro and T. Homem-de-Mello (1998), “A simulation-based approach to two-stage stochastic programming with recourse”, *Mathematical Programming*, **81**, 301-325.

[65] A. Shapiro and T. Homem-de-Mello (1999), “On rate of convergence of Monte Carlo approximations of stochastic programs”, technical report, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332-0205.

[66] A. Shapiro, J. Kim and T. Homem-de-Mello (2000), “Conditioning of Stochastic Programs,” technical report, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332-0205.

[67] Shor, N.Z. (1985), “Minimization Methods for Non-Differentiable Functions”, publisher: Springer-Verlag.

[68] P. M. Vaidya (1996), “A new algorithm for minimizing convex functions over convex sets”, *Mathematical Programming*, **73**, 291-341.

[69] R. Van Slyke and R. J.-B. Wets (1969), “L-shaped linear programs with applications to optimal control and stochastic programming”, *SIAM Journal on Applied Mathematics*, **17**, 638-663.

[70] Verweij, B., Ahmed, S., Kleywegt, A.J., Nemhauser, G. and Shapiro, A. (2002), ”The sample average approximation method applied to stochastic routing problems: a computational study”, technical report, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332-0205.

[71] R.J.-B. Wets (1974), “Stochastic Programs With Fixed Recourse: The Equivalent Deterministic Program”, *SIAM Review*, (**16**)(3), 309–339.

[72] R. J.-B. Wets (1988), “Large Scale Linear Programming Techniques”, in: Numerical Techniques for Stochastic Optimization, editors: U. Ermoliev and R. J.-B. Wets, publisher: Springer-Verlag.

[73] R. J.-B. Wets (1989), “Stochastic Programming,” in: Handbooks in Operations Research and Management Science”, Volume 1: Optimization, editors: G.L. Nemhauser, A.H.G. Rinnoy Kan, and M.J. Todd, publisher: North-Holland, Amsterdam, 573-630.

[74] R. J.-B. Wets and W. T. Ziemba (1999), "Preface," *Annals of Operations Research*, **85**.

[75] A. C. Williams (1966), "Approximation Formulas for Stochastic Linear Programming", *SIAM Journal of Applied Mathematics*, **4**(14), 668-677.

[76] D. Yang and S.A. Zenios (1995), "A Scalable Parallel Interior Point Algorithm for Stochastic Linear Programming and Robust Optimization", Report 95-07, Department of Public and Business Administration, University of Cyprus, Nicosia, Cyprus.

[77] Zangwill, W.I. (1969), "Nonlinear Programming: A Unified Approach", publisher: Prentice Hall.