

Conditional Value-at-Risk Approximation to Value-at-Risk Constrained Programs: A Remedy via Monte Carlo

L. Jeff Hong

Department of Industrial Engineering and Logistics Management, The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong, China, hongjl@ust.hk

Zhaolin Hu

School of Economics and Management, Tongji University, Shanghai 200092, China, huzhaolin@gmail.com

Liwei Zhang

School of Mathematical Science, Dalian University of Technology, Dalian 116024, China, lwzhang@dlut.edu.cn

We study optimization problems with value-at-risk (VaR) constraints. Because it lacks subadditivity, VaR is not a coherent risk measure and does not necessarily preserve the convexity. Thus, the problems we consider are typically not provably convex. As such, the conditional value-at-risk (CVaR) approximation is often used to handle such problems. Even though the CVaR approximation is known as the best convex conservative approximation, it sometimes leads to solutions with poor performance. In this paper, we investigate the CVaR approximation from a different perspective and demonstrate what is lost in this approximation. We then show that the lost part of this approximation can be remedied using a sequential convex approximation approach, in which each iteration only requires solving a CVaR-like approximation via certain Monte Carlo techniques. We show that the solution found by this approach generally makes the VaR constraints binding and is guaranteed to be better than the solution found by the CVaR approximation and moreover is empirically often globally optimal for the target problem. The numerical experiments show the effectiveness of our approach.

Keywords: value-at-risk; conditional value-at-risk; Monte Carlo; CVaR-like approximation

History: Accepted by Marvin K. Nakayama, Area Editor for Simulation; received July 2012; revised July 2013; accepted July 2013. Published online in *Articles in Advance* February 13, 2014.

1. Introduction

Financial risk management has emerged as one of the major concerns of investors. To quantify the potential risks and guide risk management, a number of so-called risk measures have been proposed in academia as well as in practice in recent years. Among them, value-at-risk (VaR) has reached a high status and gained the most attention. VaR, also known as quantile, can be interpreted as the maximum loss that is not exceeded with a given probability. Mathematically, for a specified confidence level $\alpha \in (0, 1)$, the $(1 - \alpha)$ -VaR of a random loss η is defined as $\text{VaR}_{1-\alpha}(\eta) := \inf\{v \in \Re: \Pr\{\eta \leq v\} \geq 1 - \alpha\}$. It has been incorporated in Basel II Accord and is widely used among international banks and financial institutions; for instance, the Bank for International Settlement uses the 10-day VaR at the 99% level to measure the adequacy of bank capital. For a comprehensive review of the VaR, see Duffie and Pan (1997) and Jorion (2010).

In this paper, we consider the following decision model:

$$\underset{x \in X}{\text{minimize}} \quad h(x) \quad (1)$$

$$\text{subject to} \quad \text{VaR}_{1-\alpha}(c(x, \xi)) \leq 0 \quad (2)$$

because of the central role and wide acceptance of VaR in financial practice. We call problem (1) a “VaR-constrained program” throughout the analysis. In problem (1), x is a d -dimensional decision vector, ξ is a k -dimensional vector of random parameters, and the support of ξ , denoted as Ξ , is a closed subset of \Re^k , X is a subset of \Re^d , $h: \Re^d \rightarrow \Re$ is the objective function, $c: \Re^{d+k} \rightarrow \Re$ is the loss function that models the potential risks, and $\text{VaR}_{1-\alpha}(c(x, \xi))$ denotes the $(1 - \alpha)$ -VaR of the random variable $c(x, \xi)$ given x . Note that $\text{VaR}_{1-\alpha}(c(x, \xi))$ is a function of x . We assume throughout this paper that X is a convex compact set, $h(x)$ is continuously differentiable and convex (e.g., linear) in x , and $c(x, \xi)$ is convex in x for every $\xi \in \Xi$.

The optimization model (1) appears in a number of areas with various backgrounds. Portfolio optimization, credit risk management, and reliability-based optimal design are well-known representatives of these areas. This model is also often studied in another equivalent form known as chance-constrained programs (CCP) (see, e.g., Charnes et al. 1958, Prékopa 2003, Hong et al. 2011, Hu et al. 2013). As an example of problem (1), consider a portfolio optimization problem where an investor aims to maximize the expected future payoff of the portfolio, while controlling the risk (the VaR of the portfolio) below a certain threshold. Suppose the investor has d assets with random returns. Let $r = (r_1, \dots, r_d)^T$ and $x = (x_1, \dots, x_d)^T$, where r_j is the return of asset j and x_j is the capital invested in this asset. Let $\mathbf{1}$ be the $d \times 1$ vector with all elements being 1. Suppose further that w is the allowed risk limit, the total wealth to be invested is 1, and short selling of the assets is not allowed. Then the investor's problem can be straightforwardly formulated as the following instance of problem (1):

$$\begin{aligned} & \text{minimize} && -(E[r])^T x \\ & \text{subject to} && \text{VaR}_{1-\alpha}(-r^T x) \leq w, \\ & && \mathbf{1}^T x \leq 1, \quad x \geq 0. \end{aligned} \quad (3)$$

Problem (3) is a typical mean-VaR portfolio selection formulation. The implication and impact of introducing a VaR constraint in mean-variance models, like problem (3), have been studied widely in the portfolio literature (see, e.g., Alexander and Baptista 2004, Gaivoronski and Pflug 2005).

Because closed-form expressions for VaR functions are typically not available except for some simple cases, evaluations of VaR functions are typically done using Monte Carlo simulation (e.g., Chu and Nakayama 2012). This is motivation to solve problem (1) using a Monte Carlo method, in which the VaR function is substituted by its Monte Carlo estimate. However, even for the Monte Carlo counterpart of problem (1), there exist certain difficulties when applying an optimization procedure. First, VaR (quantile) is a probability (expectation of an indicator function)-based risk measure. Thus, the sampling-based estimation/approximation will typically result in optimization problems that have certain discontinuity, which makes optimization procedures difficult. Second, VaR fails to possess the so-called "coherency." The seminal paper by Artzner et al. (1999) defines a set of axioms and calls a risk measure that satisfies these axioms a *coherent risk measure*. The authors argue that a "good" risk measure should be coherent. They also show that VaR does not always satisfy the subadditivity axiom and thus is not a coherent risk measure. The lack of subadditivity implies that VaR

does not encourage diversification, whereas diversification is generally considered a reasonable pursuit to control the risks. From the computational aspect, the lack of subadditivity implies that $\text{VaR}_{1-\alpha}(c(x, \xi))$ may not be a convex function of x even though the original loss function $c(x, \xi)$ is convex in x for every ξ . As a consequence, problem (1) may not be a convex optimization problem.

The aforementioned discontinuity nature and the nonconvexity become major obstacles in designing efficient algorithms for solving problem (1) or its Monte Carlo counterpart. Among the literature discussing Monte Carlo-based approaches, Pagnoncelli et al. (2009) studied both theoretical and computational aspects of the sample average approximation (SAA) for CCPs. Theoretically, they showed the convergence of the SAA. To practically solve the SAA, they reformulated the problem into some mixed-integer program (MIP) and suggested certain MIP tools. The MIP approach can handle a reasonably large sample size—for example, a few hundreds—but will encounter computational difficulty for Monte Carlo-scaled sample size, say, 10,000. Indeed, little progress has been made in the computational aspect for the general VaR optimization models in the literature thus far. The existing methods are either suitable for small problems or under some special assumptions on the distributions of the random parameters.

One possible compromise to the aforementioned difficulties is to seek some convex conservative approximation (CCA) for the VaR-constrained program (e.g., Ben-Tal and Nemirovski 2000, Nemirovski and Shapiro 2006, Ben-Tal et al. 2009, Chen et al. 2010, Rockafellar and Uryasev 2000). The conditional value-at-risk (CVaR) approximation, popularized by Rockafellar and Uryasev (2000), has been a representative of this kind of CCA. CVaR, conceptually interpreted as the mean of tail loss beyond VaR, was formally defined in Rockafellar and Uryasev (2000) for a random loss η with some continuous distribution. Rockafellar and Uryasev (2002) further generalize the notion to the general distributions (continuous, discrete, or other mixed forms). In contrast to VaR, CVaR satisfies the subadditivity and is a coherent risk measure (Rockafellar and Uryasev 2002). In Nemirovski and Shapiro (2006), it was established that the CVaR approximation is the "best" among all the CCAs of the VaR-constrained program under the so-called generating function scheme, in the sense that it finds less conservative solutions than other CCAs. The CVaR approximation is in general a stochastic program, as the CVaR functions are still typically not in analytical expressions. Rockafellar and Uryasev (2000) show that the Monte Carlo (sample) counterpart of the classical portfolio optimization with a CVaR objective can be reformulated into a linear program. This approach

has been further developed or applied by Rockafellar and Uryasev (2002), Krokmal et al. (2002), Trindade et al. (2007), and many others. Together with the Monte Carlo methods, the CVaR approximation has been widely used in practice as well as in research to tackle VaR-related optimization problems.

Although using CVaR approximation to handle problem (1) could be very convenient in the real applications, it has to be admitted that there exists a gap between the VaR constraint and corresponding CVaR constraint, and solving the CVaR approximation does not really solve the original VaR-constrained program. The solution obtained by the CVaR approximation does not satisfy any optimality conditions and is generally an interior point of problem (1). However, in real applications the optimal solutions of problem (1) often make the VaR constraint tight (binding) because the VaR constraint is often the critical constraint that restricts the choice of decisions. Moreover, in many situations, problem (1) is already a convex optimization problem, though we do not know this a priori. In this circumstance, implementing a convex approximation to approximate a convex problem seems like a waste of the convexity of the original problem.

These concerns draw our interests and motivate our research. In this paper, we are especially interested in the gap between VaR and CVaR that has scarcely been studied; we aim to investigate the possible connections between VaR and CVaR when they are involved in risk constraints in optimization models. To accomplish this, we rederive the CVaR approximation using the formulation of Hong et al. (2011) that represents the VaR constraint as a limit of the difference of convex functions (DC) constraint and demonstrate what is lost in the CVaR approximation. Based on the conservatism of the CVaR approximation, we convexify the DC function via a Taylor approximation and show that the newly formulated optimization problem is equivalent to a “CVaR-like approximation” problem. We then use a sequential convex approximation (SCA) procedure to recover the part that is discarded by the CVaR approximation. Our approach starts from the solution found by the CVaR approximation and makes improvements at each iteration by solving a CVaR-like approximation; it finally leads to an optimal solution of the original VaR constrained program, or a feasible solution essentially tightening the VaR constraint (the notion of “essentially” will be introduced in §3.2).

Note that this result leads to two properties for our approach. First, our solution is better than any CCAs of the generating function types, as the CVaR approximation is the best among them. Second, our approach can ensure either that a global optimal solution is found or that the VaR constraint essentially becomes

tight. We implement our approach to the mean VaR portfolio selection problem introduced earlier and a credit risk optimization model. The numerical results show that our approach works well and achieves significantly improved decisions for both problems.

Briefly, the main contributions of this paper can be clarified as follows: (1) We clarify the relation between the CVaR approximation and various CCAs of the VaR-constrained program, place CVaR properly when Monte Carlo techniques are brought in, and make the picture more complete. (2) We build an interesting linkage between VaR and CVaR constraints and demonstrate the conservatism of the CVaR approximation. (3) We introduce various optimization techniques and approaches for solving the CVaR-related optimization problems, make comparisons for these approaches, and provide guidance on how to select and apply these techniques in different situations. (4) We design efficient procedures to remedy the CVaR approximation and use CVaR optimization techniques to solve the VaR-constrained program.

The remainder of this paper is organized as follows. In §2 we introduce the CVaR approximation and discuss its conservatism. In §3 we show how to remedy the lost part of the CVaR approximation via iteratively solving a sequence of CVaR-like approximations. In §4 we implement Monte Carlo methods and introduce multiple approaches to solve the CVaR-like approximations. The numerical experiments are conducted in §5, which show the performances of our method. Section 6 concludes the paper. All the proofs are included in the online supplement (available as supplemental material at <http://dx.doi.org/10.1287/ijoc.2013.0572>).

2. Conservatism of CVaR Approximation

First, we impose the following assumptions to make problem (1) more clearly defined.

ASSUMPTION 1. For every $x \in X$, $\Pr\{c(x, \xi) = 0\} = 0$.

ASSUMPTION 2. For every $x \in X$, $c(x, \xi)$ is continuously differentiable at x for almost every $\xi \in \Xi$.

ASSUMPTION 3. There exists a random function $M(\xi)$ with $E[M(\xi)] < \infty$ such that

$$|c(x, \xi)| \leq M(\xi), \quad \forall x \in X, \xi \in \Xi.$$

Furthermore, there exists a random function $K(\xi)$ with $E[K(\xi)] < \infty$ such that

$$|c(x_1, \xi) - c(x_2, \xi)| \leq K(\xi) \|x_1 - x_2\|, \quad \forall x_1, x_2 \in X, \xi \in \Xi.$$

Assumption 1 means that $c(x, \xi)$ has no mass at 0. It is simply satisfied when $c(x, \xi)$ is a continuous random variable. Assumption 2 states that $c(x, \xi)$ is

smooth in x . It can also be easily satisfied. Assumption 3 requires that $c(x, \xi)$ is dominated by an integrable random variable and satisfies the Lipschitz condition. It is a standard assumption in stochastic optimization literature; see, for instance, Shapiro et al. (2009) and Hong and Liu (2009). It is also a critical assumption for the expectations considered next to be well-defined and differentiable. Assumptions 1–3 are imposed on the original loss function $c(x, \xi)$ within the VaR constraint. They are used to clearly specify the problems we target. The mean-VaR portfolio selection problems serve as simple examples that satisfy the assumptions.

2.1. CVaR Approximation and Its Relation to Various Convex Conservative Approximations

We first introduce the CVaR approximation and briefly review its relation to the various CCAs in the literature. The conventional conditional expectation definition of the CVaR could be quite subtle as it requires differentiating continuous and noncontinuous distributions. Rockafellar and Uryasev (2002) show that the CVaR for general distributions can be expressed in a unified way, as the following stochastic program:

$$\text{CVaR}_{1-\alpha}(\eta) = \inf_{t \in \mathfrak{R}} \left\{ \frac{1}{\alpha} E[\eta + t]^+ - t \right\}, \quad (4)$$

where $[z]^+ = \max\{z, 0\}$. Let $\text{VaR}_{1-\alpha}^+(\eta) := \inf\{v \in \mathfrak{R}: \Pr\{\eta \leq v\} > 1 - \alpha\}$. Then $\text{VaR}_{1-\alpha}(\eta) \leq \text{VaR}_{1-\alpha}^+(\eta)$ and the two coincide except when the distribution function $\Pr\{\eta \leq v\}$ is the constant $1 - \alpha$ over a certain v interval. Rockafellar and Uryasev (2002) further show that the infimum in (4) is (only) attained at any point of the interval $[-\text{VaR}_{1-\alpha}^+(\eta), -\text{VaR}_{1-\alpha}(\eta)]$. Plugging $-\text{VaR}_{1-\alpha}(\eta)$ in (4), we immediately see that CVaR is an upper bound of VaR; therefore, minimizing CVaR also results in low VaR. The idea of minimizing CVaR to lower VaR is from Rockafellar and Uryasev (2000). Nemirovski and Shapiro (2006) noted that this idea can also be used to handle the VaR constraint. In particular, using a generating function scheme, they derived the following so-called ‘‘CVaR approximation’’:

$$\text{minimize}_{x \in X} h(x) \quad (5)$$

$$\text{subject to } \text{CVaR}_{1-\alpha}(c(x, \xi)) \leq 0. \quad (6)$$

As discussed, because CVaR bounds VaR from above, problem (5) is a conservative approximation of problem (1); because CVaR is a coherent risk measure, $\text{CVaR}_{1-\alpha}(c(x, \xi))$ is a convex function of x . Thus, problem (5) is a CCA of problem (1). Nemirovski and Shapiro (2006) further demonstrated that under their generating function scheme, problem (5) is the best CCA.

From (4) it is not difficult to show that problem (5) is equivalent to

$$\text{minimize}_{x \in X, t \in \mathfrak{R}} h(x) \quad (7)$$

$$\text{subject to } E[[c(x, \xi) + t]^+] - \alpha t \leq 0,$$

in the sense that x is an optimal solution of problem (5) if and only if there exists t such that (x, t) is an optimal solution of problem (7) (e.g., Krokhmal et al. 2002). Suppose (x, t) is a feasible solution of problem (7). Then we must have $t \geq 0$. Furthermore, given any feasible solution (x, t) , let $t(x) = -\text{VaR}_{1-\alpha}(c(x, \xi))$. Then $(x, t(x))$ is also a feasible solution of problem (7). Note that from Assumption 3 we have for every $x \in X$,

$$\begin{aligned} t(x) &= -\inf\{v \in \mathfrak{R}: \Pr(c(x, \xi) \leq v) \geq 1 - \alpha\} \\ &\leq -\inf\{v \in \mathfrak{R}: \Pr(-M(\xi) \leq v) \geq 1 - \alpha\} \\ &= -\text{VaR}_{1-\alpha}(-M(\xi)). \end{aligned}$$

This suggests we can further assume in problem (7) that t is constrained in a bounded closed interval T , e.g., $T = [0, -\text{VaR}_{1-\alpha}(-M(\xi))]$. This treatment does not affect our analysis and computation and is only used to make the feasible region compact. Reformulating problem (5) as problem (7) allows us to solve problem (5) using a standard SAA approach that has been studied extensively in the literature. In §4 we will introduce different approaches to handle problems (5) and (7).

The CVaR approximation, the best CCA of the generating function type, turns out to have a close relation with the various CCAs in the literature. The multiple CCAs can often be derived by finding a certain convex function $\pi(\cdot)$ to bound $E[[\cdot]^+]$ from above and then replacing $E[[\cdot]^+]$ with $\pi(\cdot)$ in the stochastic representation (4) of the CVaR. An interesting construction of $\pi(\cdot)$ is to use the exponential function $\mu \exp(\mu^{-1}(z + t) - 1)$, where $\mu > 0$ to bound the piecewise linear function $[z + t]^+$. Replacing $[c(x, \xi) + t]^+$ in (4) with $\mu \exp(\mu^{-1}(c(x, \xi) + t) - 1)$, we clearly have the following constraint:

$$\inf_{t \in \mathfrak{R}} \inf_{\mu > 0} \left\{ \frac{1}{\alpha} E \left[\mu \exp \left(\frac{c(x, \xi) + t}{\mu} - 1 \right) \right] - t \right\} \leq 0, \quad (8)$$

which is a CCA of constraint (6). Suppose the settings and assumptions are the same as in Nemirovski and Shapiro (2006). It can be proven that (8) is equivalent to the Bernstein approximation. Actually, the two infima in (8) can be interchanged freely. Switching the order, the infimum with respect to $t \in \mathfrak{R}$ is attained at

$$t^* = -\mu \log E \left[\exp \left(\frac{c(x, \xi)}{\mu} \right) \right] + \mu \log \alpha + \mu.$$

Substituting t^* into (8) yields

$$\inf_{\mu > 0} \mu \log E \left[\exp \left(\frac{c(x, \xi)}{\mu} \right) \right] - \mu \log \alpha \leq 0,$$

which is exactly the Bernstein approximation of Nemirovski and Shapiro (2006). This shows the Bernstein approximation is indeed a CCA of the CVaR approximation (see online supplement for more discussion). Taking a step further, the quadratic approximation in Ben-Tal and Nemirovski (2000) can also be derived using the above bounding scheme. Actually, Nemirovski and Shapiro (2006) pointed out that the quadratic approximation of Ben-Tal and Nemirovski (2000) can be derived by further bounding the logarithmic moment generating function in the Bernstein approximation, and thus can be viewed as a less accurate version of Bernstein approximation. Because the Bernstein approximation is a CCA of the CVaR approximation, there exists a certain gap between the two approximations. To bridge this gap, Ben-Tal et al. (2009) suggested replacing the single exponential function in the Bernstein approximation with an exponential polynomial, which results in a bridged Bernstein-CVaR approximation. Ben-Tal et al. (2009) and Nemirovski (2012) showed numerically that this bridged Bernstein-CVaR approximation can fill the gap to a large extent.

In the more general robust optimization framework, the interesting relation between CVaR and various robust approximations was uncovered by Chen et al. (2010). They showed that the multiple robust approximations of the VaR-constrained program can be obtained from bounding the CVaR function by incorporating all kinds of information on the random parameters, e.g., the mean, the variance, and the forward and backward deviations. They also demonstrated how to combine those robust approximations to reduce the conservatism of the robust optimization framework.

In summary, this analysis shows that the various CCAs of the VaR-constrained program are actually CCAs of the CVaR approximation. The CCA framework works on one side of the CVaR, proposes to find a good deterministic CCA of the CVaR, and tries to get close to the CVaR.

2.2. Conservatism of Conditional Value at Risk Approximation: An Example

Despite being the best CCA within a certain category, the CVaR approximation is not an accurate approximation of problem (1). The solution obtained by the CVaR approximation does not satisfy any optimality conditions and is generally an interior point of problem (1). Thus, the following question raises wide concerns: How conservative is the CVaR approximation?

This question is generally difficult to answer. In some cases, problem (5) provides an ideal approximation to problem (1), whereas in many situations it is rather conservative. Let us consider a simple

instance of problem (3). To derive problem (3) analytically, we assume r follows a multivariate normal distribution with mean vector μ and covariance matrix Σ . Simple computation shows that

$$\begin{aligned} \text{VaR}_{1-\alpha}(-r^T x) &= -\mu^T x + \kappa_1(\alpha) \sqrt{x^T \Sigma x}, \quad \text{with} \\ \kappa_1(\alpha) &= \Phi^{-1}(1 - \alpha); \\ \text{CVaR}_{1-\alpha}(-r^T x) &= -\mu^T x + \kappa_2(\alpha) \sqrt{x^T \Sigma x}, \quad \text{with} \\ \kappa_2(\alpha) &= \alpha^{-1} \phi(\Phi^{-1}(1 - \alpha)), \end{aligned}$$

where $\Phi^{-1}(z)$ is the inverse of the standard normal cumulative distribution function and $\phi(z)$ is the density function of the standard normal distribution (e.g., Rockafellar and Uryasev 2000). It follows that the VaR function in problem (3) is convex provided $\alpha \leq 1/2$, and the corresponding CVaR function is always convex no matter what value α takes. Note that this is consistent with the fact that the CVaR approximation is a convex optimization problem. Therefore, under the normality assumption, both problem (3) and its CVaR approximation can be converted to deterministic convex optimization problems. Such deterministic problems are known as second-order cone programs (SOCP), which can be solved efficiently using conventional software (e.g., CVX package) or more professional software (e.g., Mosek and SeDuMi).

The parameters $\kappa_i(\alpha)$, $i = 1, 2$ are often viewed as the risk factor (e.g., El Ghaoui et al. 2003). Obviously, the conservatism of the CVaR approximation depends on the difference between the risk factors $\kappa_1(\alpha)$ and $\kappa_2(\alpha)$. For instance, we assume $E[r_i]$ is evenly spread between 0.04 and 0.50 and increasing with i , the standard deviation $\text{std}[r_i] = E[r_i] + 0.05$ for all $i = 1, \dots, d$ and the correlation between r_i and r_j is 0.35 for any $i \neq j$ (the parameters are the same as that used in Hong and Liu (2009, Problem (25))), except that the variances are slightly enlarged). We consider different combinations of d , α , and w and compute optimal values for problem (3) and its CVaR approximation by solving the SOCPs, and summarize the results in Table 1.

Table 1 Conservatism of CVaR Approximation

		$\alpha = 0.10$	$\alpha = 0.05$	$\alpha = 0.01$
$d = 10$	$w = 0.05$	0.139/0.422	0.084/0.182	0.047/0.062
	$w = 0.10$	0.278/0.440	0.167/0.348	0.094/0.125
	$w = 0.15$	0.375/0.449	0.251/0.405	0.141/0.187
$d = 50$	$w = 0.05$	0.230/0.485	0.116/0.355	0.059/0.082
	$w = 0.10$	0.433/0.488	0.232/0.467	0.118/0.163
	$w = 0.15$	0.471/0.490	0.348/0.480	0.177/0.245
$d = 100$	$w = 0.05$	0.261/0.492	0.128/0.425	0.062/0.086
	$w = 0.10$	0.464/0.494	0.249/0.483	0.123/0.172
	$w = 0.15$	0.485/0.495	0.374/0.490	0.185/0.259

In Table 1 there are two values for each parameter combination; the second is the (negative) optimal value of problem (3) and the first is the (negative) optimal value of corresponding CVaR approximation (all results are rounded to three digits). From Table 1, we see clearly that for some parameter combinations, the CVaR approximation is close to the VaR constrained program (e.g., $d = 100$, $\alpha = 0.1$, and $w = 0.15$). However, for some combinations, the CVaR approximation is rather conservative (e.g., $d = 50$, $\alpha = 0.05$, and $w = 0.10$). Clearly, the investors concerned only about VaR risk wish to gain more than what the CVaR approximation can provide.

2.3. A Different Scheme

Let us look at the CVaR approximation from an angle other than the generating function scheme of Nemirovski and Shapiro (2006). It is not difficult to verify that the VaR constraint (2) is equivalent to $\Pr\{c(x, \xi) \leq 0\} \geq 1 - \alpha$. Consequently, problem (1) can be rewritten as the following problem:

$$\begin{aligned} & \underset{x \in X}{\text{minimize}} && h(x) \\ & \text{subject to} && p(x) \leq \alpha, \end{aligned} \tag{9}$$

with $p(x) := \Pr\{c(x, \xi) > 0\}$. Hong et al. (2011) proposed a new approach to handle problem (9), and their basic idea was to reformulate $p(x)$ as the infimum of a DC function. Specifically, define $\pi(z, t) = t^{-1}\{[z + t]^+ - [z]^+\}$ on $\Re \times \Re_{++}$, where \Re_{++} denotes the set of positive reals. Then

$$E[\pi(c(x, \xi), t)] = \frac{1}{t} \{E[[c(x, \xi) + t]^+] - E[[c(x, \xi)]^+]\}.$$

Moreover, for every z , $\pi(z, t)$ is nondecreasing in t and $\lim_{t \rightarrow \infty} \pi(z, t) = 1_{[0, +\infty)}(z)$, where $1_A(z)$ denotes the indicator function of A that equals 1 if $z \in A$, and 0 otherwise. It follows that for any $t > 0$,

$$E[\pi(c(x, \xi), t)] \geq E[1_{[0, +\infty)}(c(x, \xi))] = \Pr\{c(x, \xi) \geq 0\}.$$

By the monotone convergence theorem, we have $\Pr\{c(x, \xi) \geq 0\} = \inf_{t > 0} E[\pi(c(x, \xi), t)]$, from which it is immediately seen that $\inf_{t > 0} E[\pi(c(x, \xi), t)] \leq \alpha$ implies $p(x) \leq \alpha$. Noting the definition of $E[\pi(c(x, \xi), t)]$, we obtain the following problem:

$$\begin{aligned} & \underset{x \in X}{\text{minimize}} && h(x) \\ & \text{subject to} && \inf_{t > 0} \frac{1}{t} \{E[[c(x, \xi) + t]^+] \\ & && - E[[c(x, \xi)]^+]\} \leq \alpha, \end{aligned} \tag{11}$$

which is a conservative approximation of problem (9). Let Ω_0 and Ω denote the feasible sets of problem (9) and problem (10), respectively. Then $\Omega \subset \Omega_0$ and

$$\Omega_0 \setminus \Omega = \{x \in X: \Pr\{c(x, \xi) \geq 0\} > \alpha, \Pr\{c(x, \xi) > 0\} \leq \alpha\}.$$

By Assumption 1, $\Omega_0 \setminus \Omega$ is an empty set. Therefore, problem (10) is equivalent to Problem (9).

It seems that reformulating problem (1) into problem (10) does not simplify the analysis. However, problem (10) has a structure that is worthwhile noting: both $E[[c(x, \xi) + t]^+]$ and $E[[c(x, \xi)]^+]$ are convex functions of x ; consequently, $E[[c(x, \xi) + t]^+] - E[[c(x, \xi)]^+]$ is a DC function of x . Given this DC structure, one possible approach to handle problem (10) is to find a concave function $g(x)$ bounding $E[[c(x, \xi)]^+]$ from below, i.e., $g(x) \leq E[[c(x, \xi)]^+]$ for all $x \in X$, and then use

$$\inf_{t > 0} \frac{1}{t} \{E[[c(x, \xi) + t]^+] - g(x)\} \leq \alpha$$

to approximate constraint (11). Naturally, different $g(x)$ functions will result in different conservative approximations of problem (10). Perhaps the simplest choice under this scheme is to use the lower bound $g(x) \equiv 0$ to approximate $E[[c(x, \xi)]^+]$. This results in the following optimization problem:

$$\underset{x \in X}{\text{minimize}} h(x) \tag{12}$$

$$\text{subject to } \inf_{t > 0} \frac{1}{t} E[[c(x, \xi) + t]^+] \leq \alpha. \tag{13}$$

It turns out that this optimization problem is directly linked to the CVaR approximation, as shown in the following proposition.

PROPOSITION 1. *Suppose Assumption 1 is satisfied. Then problem (12) is equivalent to problem (5).*

Proposition 1 shows that we can rederive the CVaR approximation using the DC reformulation approach under certain conditions, which is different from the generating function scheme of Nemirovski and Shapiro (2006) and Ben-Tal et al. (2009). This new approach also suggests that the CVaR approximation actually drops a term $E[[c(x, \xi)]^+]$ from the original constraint (11). Thus, roughly speaking, the value of function $E[[c(x, \xi)]^+]$ determines the conservatism of the CVaR approximation.

REMARK 1. Constraint (13) is closely related to constraint (2.8), i.e.,

$$\inf_{t > 0} [E[[c(x, \xi) + t]^+] - t\alpha] \leq 0. \tag{14}$$

As is easily seen, constraint (14) can be obtained by multiplying both sides of constraint (13) by t and placing $t\alpha$ on the left-hand side. Note that Nemirovski and Shapiro (2006) deduced constraint (14) using the generating function scheme as the best CCA of the VaR constraint (2) and showed further that (14) is equivalent to the CVaR constraint, i.e., constraint (6). We may conjecture that constraint (13) is

also equivalent to constraint (6) without any conditions. However, we should be cautious here, as the equivalence of the two constraints might be destroyed by the operation of multiplying t when the infimum is attained at $t=0$. A simple one-dimensional example, $\inf_{t>0}\{x^2+1-x^2\} \leq 0$, is not equivalent to $\inf_{t>0} t\{x^2+1-x^2\} \leq 0$. Actually, (13) and (6) are not equivalent in general. For instance, if $x \in X$ satisfies $c(x, \xi) \equiv 0$ for every $\xi \in \Xi$, then x satisfies constraint (6) but violates constraint (13). Therefore, Assumption 1 is required for the equivalence to hold. This nonequivalence is indeed counterintuitive. In the online supplement, we discuss the relationship between constraints (13) and (6) in a more general setting. As will be seen, Proposition 1 can actually be embedded into that general setting.

3. Remedy of CVaR Approximation

We have rewritten problem (9) as problem (10) under certain assumptions. Our major concern about the CVaR approximation here, is that using $g(x) \equiv 0$ to approximate function $E[[c(x, \xi)]^+]$ may be rather conservative. Can we improve on this bound, and can we find the lost term $E[[c(x, \xi)]^+]$, provided that the computational budget is not critical? In this section we focus on these questions. Before the discussion, we first introduce some properties of the CVaR and its stochastic program representation. We summarize the properties in the following proposition.

PROPOSITION 2. For any $x \in X$,

$$\text{CVaR}_{1-\alpha}(c(x, \xi)) \leq \alpha^{-1}E[[c(x, \xi)]^+]. \quad (15)$$

Suppose that Assumption 1 is satisfied. Then $\alpha^{-1}E[[c(x, \xi) + t]^+]$ is differentiable at $t=0$ and

$$\nabla_t \left\{ \frac{1}{\alpha} E[[c(x, \xi) + t]^+] - t \right\} \Big|_{t=0} = p(x)/\alpha - 1. \quad (16)$$

Furthermore, the following are equivalent:

- (a) $p(x) = \alpha$.
- (b) $t=0$ is a minimizer of (4).
- (c) Equality holds in (15).
- (d) $\text{VaR}_{1-\alpha}(c(x, \xi)) \leq 0 \leq \text{VaR}_{1-\alpha}^+(c(x, \xi))$.

From Proposition 2 we see that the kink point 0 is critical in the formulations of the CVaR and its stochastic program representation. That is the main reason we impose Assumption 1. Proposition 2 explores the inherent connection between the probability (VaR) value and the optimal solution of the stochastic program representation of CVaR. We will frequently use these results in the following analysis.

3.1. CVaR-like Approximation

Let us still focus on the convex function $E[[c(x, \xi)]^+]$ that is embedded in the infimum of constraint (11). Suppose that Assumptions 1–3 are satisfied. Then we can easily verify the conditions (A1), (A2), and (A4) in Shapiro et al. (2009, Theorem 7.44) for $[c(x, \xi)]^+$. It follows from the theorem that $E[[c(x, \xi)]^+]$ is differentiable at any $x \in X$ and $\nabla_x E[[c(x, \xi)]^+] = E[\nabla_x [c(x, \xi)]^+]$, where ∇_x denotes the gradient of a function with respect to x . Clearly, the best (least conservative) concave approximation of the convex function $E[[c(x, \xi)]^+]$ is the hyperplane that supports the convex function $E[[c(x, \xi)]^+]$ at certain point (such a hyperplane is called a tangent hyperplane). This motivates us to linearize function $E[[c(x, \xi)]^+]$ at some specific point. Now suppose $y \in X$ is any interior point of problem (9); i.e., $p(y) < \alpha$. Let

$$L(x, y) = E[[c(y, \xi)]^+] + \nabla_x E[[c(y, \xi)]^+]^T(x - y)$$

be the first-order Taylor approximation of $E[[c(x, \xi)]^+]$ at point y . We can then make the DC function convex in constraint (11) using $L(x, y)$ and propose a solution for the following problem:

$$\begin{aligned} & \text{minimize}_{x \in X} h(x) \\ & \text{subject to} \quad \inf_{t>0} \frac{1}{t} \{E[[c(x, \xi) + t]^+] - L(x, y)\} \leq \alpha. \end{aligned} \quad (17)$$

The linearization approach that makes the DC function convex and leads to Problem (17) is a standard approach in the DC program literature. At first glance, problem (17) seems still intractable, since it involves taking the infimum. However, problem (17) turns out to be a convex optimization problem in this context. Besides, the following theorem states that problem (17) is actually equivalent to the optimization problem:

$$\begin{aligned} & \text{minimize}_{x \in X} h(x) \\ & \text{subject to} \quad \text{CVaR}_{1-\alpha}(c(x, \xi)) - \alpha^{-1}L(x, y) \leq 0. \end{aligned} \quad (18)$$

THEOREM 1. Suppose that $y \in X$ satisfies $p(y) < \alpha$, and Assumptions 1–3 are satisfied. Then problems (17) and (18) are equivalent.

Theorem 1 serves as the basis for designing the remedy procedure. Since $L(x, y)$ is linear in x , problem (18) is a convex optimization problem. Note that problem (18) takes a similar form of problem (5). In this paper we refer to problem (18) as a CVaR-like approximation. Provided that the values and the gradients of the function $E[[c(x, \xi)]^+]$ at specified points can be estimated efficiently, the complexity of such CVaR-like approximations is essentially the same as that of the CVaR approximation. In §4 we will discuss how to solve the CVaR-like approximations efficiently.

REMARK 2. Some careful observation suggests that problem (18) can actually be derived by linearizing the second term of the following DC constraint at point y :

$$\text{CVaR}_{1-\alpha}(c(x, \xi)) - \alpha^{-1}E[[c(x, \xi)]^+] \leq 0 \quad (19)$$

There is a plausible way from (11) to (19). That is, (11) is equivalent to

$$\inf_{t>0} \frac{1}{t} \{ \alpha^{-1}E[[c(x, \xi) + t]^+] - \alpha^{-1}E[[c(x, \xi)]^+] \} - 1 \leq 0. \quad (20)$$

Multiplying both sides of (20) by t yields

$$\inf_{t>0} \{ \alpha^{-1}E[[c(x, \xi) + t]^+] - t \} - \alpha^{-1}E[[c(x, \xi)]^+] \leq 0, \quad (21)$$

which is very close to (19). This naturally leads to the conjecture that constraint (2) is equivalent to constraint (19). However, (15) tells us that inequality (19) always holds and actually defines the whole space and thus is not equivalent to (2). Indeed, letting $t \rightarrow 0$ in (21), we can see that (21) also always holds, and thus (21) is not equivalent to (20). Then where is the error? The error hides in the fact that the infimum in (20) is attained at $t = 0$: multiplying 0 on both sides of (20) results in an invalid inequality, i.e., (21), that always holds. This is similar to the issue discussed in Remark 1. It is somewhat tricky that before the linearization, the DC constraint (19) defines the whole space, but once we convexify it using a linear approximation, the new convex constraint only defines a subset of the region defined by constraint (2). To show that this phenomenon could indeed happen, consider a simple one-dimensional example. Suppose that our feasible region is $X = \{x \in \mathfrak{R} : -5 \leq x \leq 5, x^2 - (x^2 + 1) \leq 0\}$. Note that $x^2 - (x^2 + 1) \leq 0$ always holds and thus defines the whole space \mathfrak{R} . However, if we linearize $x^2 + 1$ at $x = 1$, then the new convex constraint $x^2 - 2x \leq 0$ will define a convex set $[0, 2]$. The convex set $[0, 2]$ obtained is a subset of our original feasible set X .

The following proposition further states that the feasible region \mathcal{C}_y or \mathcal{D}_y lies in the interior of the feasible region of Problem (9).

PROPOSITION 3. Suppose that $y \in X$ satisfies $p(y) < \alpha$ and Assumptions 1–3 are satisfied. Then

$$\text{CVaR}_{1-\alpha}(c(y, \xi)) - \alpha^{-1}L(y, y) < 0,$$

and for any $x \in \mathcal{C}_y$, $p(x) < \alpha$.

3.2. Remedy Procedure

Suppose we have solved problem (5) and obtained its optimal solution x_0 . Because x_0 is a feasible solution

of problem (5), we have $\text{CVaR}_{1-\alpha}(c(x_0, \xi)) \leq 0$, which implies $p(x_0) \leq \alpha$. Furthermore, $p(x_0) = \alpha$ can be excluded, because if this is the case, from Proposition 2 we have $\alpha^{-1}E[[c(x_0, \xi)]^+] = \text{CVaR}_{1-\alpha}(c(x_0, \xi)) \leq 0$, which contradicts $p(x_0) = \alpha$. Therefore, we have $p(x_0) < \alpha$, which suggests we can start from x_0 and set $y = x_0$ in problem (18). From Proposition 3, x_0 is a feasible solution of problem (18). Solving problem (18) we can obtain a solution x_1 that is at least as good as x_0 ; i.e., $h(x_1) \leq h(x_0)$. Moreover, from Proposition 3 we also have $p(x_1) < \alpha$. Note that once we obtain x_1 , we do not need to stop there. We can repeat this procedure at x_1 and search for better solutions. This suggests we can use a SCA procedure to gradually recover the lost part of the CVaR approximation. We propose the following algorithm to handle problem (9).

Algorithm (CVaR-SCA)

Step 0. Solve problem (5) and denote the solution as x_0 . Set $k = 0$.

Step 1. Solve

$$\begin{aligned} (C_k) \quad & \underset{x \in X}{\text{minimize}} \quad h(x) \\ & \text{subject to} \quad \text{CVaR}_{1-\alpha}(c(x, \xi)) \\ & \quad \quad \quad - \alpha^{-1}L(x, x_k) \leq 0 \end{aligned}$$

to obtain its optimal solution x_{k+1} .

Step 2. Set $k = k + 1$ and go to Step 1.

Algorithm CVaR-SCA is easy to implement, as we only need to solve a CVaR-like approximation in each iteration. Besides this, Algorithm CVaR-SCA has the desired properties. We summarize these properties in the following theorem.

THEOREM 2. Suppose that Assumptions 1–3 are satisfied. Then Algorithm CVaR-SCA satisfies the following properties:

1. For $k = 1, 2, \dots, x_k$ is a strictly feasible solution of problem (9), i.e., $p(x_k) < \alpha$.
2. $\{h(x_k), k = 1, 2, \dots\}$ is a convergent nonincreasing sequence; i.e., $h(x_{k+1}) \leq h(x_k)$. Moreover, if $h(x_{k+1}) = h(x_k)$ for some k , then x_k is a global optimal solution of problem (9).
3. The algorithm will either terminate after a finite number of iterations, or any cluster point \bar{x} of $\{x_k, k = 1, 2, \dots\}$ satisfies $p(\bar{x}) = \alpha$ and $\text{VaR}_{1-\alpha}(c(\bar{x}, \xi)) \leq 0 \leq \text{VaR}_{1-\alpha}^+(c(\bar{x}, \xi))$.

The first property in Theorem 2 shows that we always search solutions in the interior of the feasible region of problem (9). The second property states that we will make improvements at each iteration and the sequence of objective values will decreasingly converge to a certain value. Moreover, if the objective value no longer decreases at some iteration, then we have reached the global optimal solution. The third property implies that if all the stationary solutions

of the original VaR-constrained program make the VaR constraint tight, then our algorithm will converge to certain point at which the chance constraint is tight, and the VaR constraint is essentially tight; i.e., either the VaR constraint is tight, or the VaR constraint strictly holds but the corresponding random variable has no mass between the VaR value and 0.

Some other observations and discussions also follow Theorem 2. The theorem suggests that if the objective value of Problem (9) does not improve after a finite number of iterations, then we have actually found the global optimal solution. Now let us discuss the binding of the chance constraint. Consider problem (9) and the following optimization problem that is obtained by removing the chance constraint $p(x) \leq \alpha$ from problem (9):

$$\underset{x \in X}{\text{minimize}} \quad h(x). \quad (22)$$

Then we will fall in one of the following cases:

- Optimal values for problem (9) and problem (22) are equal and all the optimal solutions of problem (22) satisfy $p(x) \leq \alpha$. In this case, the chance constraint $p(x) \leq \alpha$ is redundant, and to solve (9) we only need to solve (22).

- Optimal values for problems (9) and (22) are equal, but some of the optimal solutions of problem (22) violate constraint $p(x) \leq \alpha$. In this case we cannot simply drop $p(x) \leq \alpha$, and solving (22) does not automatically solve (9), because the algorithm may return an optimal solution of (22) that violates $p(x) \leq \alpha$. In this case it is possible for Algorithm CVaR-SCA to stop at a global optimal solution of (9) where the chance constraint is not tight.

- Optimal values for problems (9) and (22) are not equal. In this case, problem (9) cannot attain optimal value at point x for which $p(x) < \alpha$, and our algorithm will not stop at point x for which $p(x) < \alpha$. Any cluster points will be on the boundary, i.e., will satisfy $p(x) = \alpha$.

Clearly, the third case is the typical situation that we encounter. In this case suppose \bar{x} is a cluster point of the sequence of solutions generated by Algorithm CVaR-SCA. Then we have $p(\bar{x}) = \alpha$. It follows from Proposition 2 that

$$\text{CVaR}_{1-\alpha}(c(\bar{x}, \xi)) - \alpha^{-1} \mathbb{E}[[c(\bar{x}, \xi)]^+] = 0.$$

Note that in §2.3, we show that the CVaR approximation loses a term $\mathbb{E}[[c(x, \xi)]^+]$. The previous equation shows that our approach can actually find this term.

REMARK 3. Implementing gradient-involved Monte Carlo methods to handle the VaR constraints or chance constraints typically requires certain differentiability for the constraint functions, and our remedy procedure is no exception. However, surprisingly, our approach only requires that the function $\mathbb{E}[[c(x, \xi)]^+]$

is differentiable in x , which can be ensured by Assumptions 1–3. Generally speaking, the requirement of differentiability of $\mathbb{E}[[c(x, \xi)]^+]$ should be much weaker than the differentiability of the VaR function or the probability function, and estimating the gradient for $\mathbb{E}[[c(x, \xi)]^+]$ is also relatively easier than that for the VaR function or the probability function. Of course, Theorem 2 does not build that the remedy procedure will finally guarantee certain stationary points for the VaR-constrained program. Whether the procedure will converge to some stationary point of the VaR-constrained program and under what conditions such convergence can be ensured are still open to the authors.

We end this section by briefly comparing our approach with the approach of Hong et al. (2011). To handle problem (10), Hong et al. (2011) propose to fix t as a small constant $\varepsilon > 0$ and call the resulted problem an ε approximation to problem (10). They show that the ε approximation is a DC program and is a conservative approximation to problem (10); moreover, under a certain set of strong conditions, both the optimal solutions and stationary points of the ε approximation converge to those of problem (10) as ε goes to 0. They then focus on solving the ε approximation via a SCA procedure.

The approach developed in this paper is quite different from Hong et al. (2011). We no longer propose any problem to approximate problem (10) as a whole at the first stage. Instead, we directly make problem (10) convex in each iteration and directly construct convex sets within the feasible region of problem (10) such that these sets contain better and better solutions. Clearly, the approximation scheme and convergence trajectory of our approach are different from that of the ε approximation approach. In some sense, our approach succeeds in removing the parameter ε . More importantly, it builds an interesting connection between VaR and CVaR. This allows us to obtain a certain sense of the conservatism of the CVaR approximation and to bridge the gap between the CVaR and the VaR constraints.

4. Monte Carlo Approaches to the CVaR-like Approximation

To implement the remedy procedure proposed in §3, we need to solve a sequence of CVaR-like approximations. In this paper we use Monte Carlo methods combined with deterministic convex optimization tools to solve the CVaR-like approximations. Suppose that we have n independent and identically distributed (i.i.d.) observations of the random vector ξ , denoted as $\xi_1, \xi_2, \dots, \xi_n$. Then $\mathbb{E}[[c(x_k, \xi)]^+]$ can be estimated by the sample mean $n^{-1} \sum_{j=1}^n [c(x_k, \xi_j)]^+$. Following

the discussion in §3.1, the gradient $\nabla_x E[[c(x_k, \xi)]^+]$ can be estimated by

$$\bar{\nabla}_x E[[c(x_k, \xi)]^+] := \frac{1}{n} \sum_{j=1}^n \nabla_x c(x_k, \xi_j) 1_{(0, +\infty)}(c(x_k, \xi_j)).$$

Consequently, we can estimate the linear function $L(x, x_k)$ by

$$L_n(x, x_k) := \frac{1}{n} \sum_{j=1}^n [c(x_k, \xi_j)]^+ + (\bar{\nabla}_x E[[c(x_k, \xi)]^+])^T (x - x_k).$$

After approximating $L(x, x_k)$ by $L_n(x, x_k)$, solving the CVaR-like approximation is essentially the same as solving the CVaR approximation, i.e., problem (5). In the past decade, CVaR optimization problems have attracted great attention. Various techniques have been developed to handle the CVaR function. In particular, Rockafellar and Uryasev (2000, 2002) suggested a linear approach to solve the CVaR optimization problem. Subsequently, Andersson et al. (2001) generalized the approach to handle credit risk optimization problems with CVaR objectives. Krokhmal et al. (2002) extended the approach to solve portfolio optimization problems with CVaR objectives and constraints. More recently, Rockafellar and Royset (2010) used the approach to solve the buffered failure probability-constrained problems in reliability engineering design. In the following sections we briefly introduce how to use the linear approach to solve the CVaR-like approximation and discuss the possible impediments of implementing this approach. We then suggest a number of approaches that may serve as alternatives to the linear approach.

4.1. Linear Approach

The linear approach is based on the following result that reformulates problem (C_k) as a $d + 1$ -dimensional optimization problem.

LEMMA 1. Problem (C_k) is equivalent to

$$(C'_k) \quad \begin{aligned} & \underset{x \in X, t \in T}{\text{minimize}} && h(x) \\ & \text{subject to} && E[[c(x, \xi) + t]^+] - \alpha t - L(x, x_k) \leq 0 \end{aligned}$$

in the sense that x is an optimal solution of problem (C_k) if and only if there exists t such that (x, t) is an optimal solution of problem (C'_k) .

Lemma 1 is a simple analogy to the equivalence between problems (5) and (7). We omit the proof since it is straightforward. As has been mentioned, problem (C'_k) is a standard convex stochastic optimization problem. Thus it can be solved by the SAA (see, e.g., Shapiro et al. 2009 for a comprehensive review about

this topic). The SAA of problem (C'_k) takes the following form:

$$\begin{aligned} & \underset{x \in X, t \in T}{\text{minimize}} && h(x) \\ & \text{subject to} && \frac{1}{n} \sum_{j=1}^n [c(x, \xi_j) + t]^+ - \alpha t - L_n(x, x_k) \leq 0. \end{aligned}$$

The linear approach suggests introducing a set of auxiliary decision variables $z_j, j = 1, \dots, n$ and reformulating the sample problem as the following problem:

$$\begin{aligned} & \text{minimize} && h(x) \\ & \text{subject to} && c(x, \xi_j) + t \leq z_j, \quad j = 1, \dots, n, \\ & && \frac{1}{n} \sum_{j=1}^n z_j - \alpha t - L_n(x, x_k) \leq 0, \\ & && x \in X, t \in T, z_j \geq 0, \quad j = 1, \dots, n. \end{aligned} \tag{23}$$

Note that when the loss function $c(x, \xi)$ is linear in x , problem (23) becomes a linear programming problem. That's why we call this approach a linear approach, though it does not necessarily yield a linear problem in our context. One of the merits of the linear approach is that it does not require differentiability conditions on the CVaR function. The convexity of the objective and constraint functions will ensure the uniform convergence of the random variables, which will further ensure the convergence of the SAA and thus the linear approach (see, e.g., Shapiro et al. 2009, Theorems 7.49 and 5.3). In contrast, the number of decision variables and the number of constraints in problem (23) are proportional to the sample size n , thus solving it could become quite slow, especially when n is relatively large, say $n \geq 10,000$. Alexander et al. (2006) considered the CVaR optimization in portfolio selection and observed that the linear programming problem of the sample CVaR problem of Rockafellar and Uryasev (2000) becomes ill conditioned and time consuming to solve when considerable samples are simulated. Undoubtedly, when $c(x, \xi)$ is nonlinear, problem (23) will be more time consuming to solve.

To speed up the computation, we looked at a number of alternative approaches. Ogryczak and Śliwiński (2011) suggested a dual approach. They formulated the dual problem of the sample CVaR optimization problem of Rockafellar and Uryasev (2000) and found that the dual problem is much easier to solve than the primal one. Specifically, they considered 50,000 scenarios and 100/200 instruments and compared the computational efforts of solving the primal and dual problems. They found that the efficiency of the dual approach is encouraging. Of course, the problem they considered has a much simpler structure than our CVaR-like approximation, and their dual problem takes a rather simple form. How to implement the dual approach to handle the sample CVaR-like

approximation, or the general CVaR-related optimization problems, can be a potential research topic. Besides the dual approach, Alexander et al. (2006) implemented the smoothing method to approximate the CVaR optimization problem. Lim et al. (2010) suggested a nondifferentiable optimization algorithm. Chung et al. (2010) proposed an active-set method. Iyengar and Ma (2013) developed a fast gradient descent method. The recent work of Basova et al. (2011) compared the linear approach, the active-set method, and the smoothing method. They found that the active-set method and the smoothing method are significantly faster than the linear approach for their reliability-based optimal design problems. Clearly, all these methods can be tailored to solve our CVaR-like approximation. This provides great freedom for implementing the remedy procedure.

4.2. Gradient-Based Approach

When the CVaR function is sufficiently smooth, then a gradient-based approach can be directly applied to solve the CVaR-like approximation. Hong and Liu (2009) provided a gradient-based Monte Carlo algorithm that directly solves the CVaR-constrained program. The gradient-based approach directly solves problem (C_k) at each iteration. It requires the evaluation of function values and gradients of the objective and constraint functions for a fixed feasible point $x \in X$. Suppose now we have an i.i.d. sample $\xi_1, \xi_2, \dots, \xi_n$ from the random vector ξ . For every $x \in X$, let $c_{[n(1-\alpha)] : n}(x)$ denote the $[n(1-\alpha)]$ th order statistic of $c(x, \xi)$ from n observations, where $[n(1-\alpha)]$ is the largest integer not exceeding $n(1-\alpha)$. Then $c_{[n(1-\alpha)] : n}(x)$ is a strongly consistent estimator of $\text{VaR}_{1-\alpha}(c(x, \xi))$ (see, for instance, Serfling 1980). We can estimate $\text{CVaR}_{1-\alpha}(c(x, \xi))$ by

$$\text{CVaR}_{1-\alpha, n}(c(x, \xi)) := \frac{1}{n\alpha} \sum_{j=1}^n c(x, \xi_j) \cdot 1_{\{c(x, \xi_j) \geq c_{[n(1-\alpha)] : n}\}}.$$

The estimator $\text{CVaR}_{1-\alpha, n}(c(x, \xi))$ is a consistent estimator of $\text{CVaR}_{1-\alpha}(c(x, \xi))$ under the assumption that $E[c(x, \xi)^2] < \infty$ (Trindade et al. 2007).

The gradient of CVaR can also be estimated directly. Specifically, we have the following result from Hong and Liu (2009).

LEMMA 2. *Suppose Assumptions 1–3 are satisfied. Suppose that $\text{VaR}_{1-\alpha}(c(x, \xi))$ is differentiable for any $x \in X$. Then for any $x \in X$,*

$$\begin{aligned} \nabla_x \text{CVaR}_{1-\alpha}(c(x, \xi)) &= E[\nabla_x c(x, \xi) \mid c(x, \xi) \\ &\geq \text{VaR}_{1-\alpha}(c(x, \xi))]. \end{aligned}$$

Based on Lemma 2, we can use

$$\bar{\nabla}_x \text{CVaR}_{1-\alpha}(c(x, \xi)) := \frac{1}{n\alpha} \sum_{j=1}^n \nabla_x c(x, \xi_j) 1_{\{c(x, \xi_j) \geq c_{[n(1-\alpha)] : n}\}}$$

to estimate $\nabla_x \text{CVaR}_{1-\alpha}(c(x, \xi))$. Following the analysis in Hong and Liu (2009), we can easily prove that $\bar{\nabla}_x \text{CVaR}_{1-\alpha}(c(x, \xi))$ is a consistent estimator of $\nabla_x \text{CVaR}_{1-\alpha}(c(x, \xi))$. Note that problem (C_k) is a convex optimization problem. We can then directly use a gradient-based algorithm to solve problem (C_k) with the estimated function values and gradients.

As has been observed in the literature (e.g., Hong and Liu 2009), the gradient-based method is at least an order of magnitude faster than the linear approach. This is reasonable because smooth optimization methods are generally faster than their non-smooth counterparts. However, it is worth noting that the gradient-based method requires more conditions on the function $c(x, \xi)$, to ensure the differentiability of the CVaR function, as shown in Lemma 2, and thus may be not as applicable as the linear approach.

The plentiful and relatively mature techniques for the CVaR optimization problems also reveal the advantages of our remedy procedure. Overall, we believe both the VaR and the CVaR, as risk measures, have their rationality and the decision maker may prefer one over the other in different situations. However, we do see that the CVaR is more considered and studied than the VaR in the aspect of computation. The optimization models built in the literature often evade the VaR not because of its rationality but because of the computation difficulties. In this paper, the problem we consider is VaR-constrained programs. But we convert handling VaR-constrained programs into solving CVaR optimization problems. Therefore, our approach can make full use of the techniques and tools developed for CVaR optimization.

4.3. Practical Consideration of Stopping Criteria

A major concern of using a Monte Carlo counterpart to approximate the true stochastic optimization problem, as suggested in this paper, may be that this approach does not necessarily guarantee a feasible solution of the true problem because of the simulation error. This is a common situation when using sampling-based methods to handle constraints involving random parameters. However, we argue that this is not a large problem, for the following reasons. First, our approach always makes approximation in the interior of the feasible region of the original problem and belongs to the inner approximation, and in most cases the solution found by our method is feasible even if some simulation error exists. When implementing Algorithm CVaR-SCA, we typically stop the algorithm if the difference between two consecutive iterations is less than a certain tolerance level. We find that when the tolerance level is comparable to the simulation error, our procedure will generally stop at solutions that rarely violate the original constraint. Second, once we obtain

a candidate solution at each iteration of the algorithm, we can always verify the feasibility via Monte Carlo estimation using a very large sample size, and we can stop the algorithm if the solution hits the boundary. For instance, we can generate a new set of samples that are independent of those used in Algorithm CVaR-SCA and use a standard sample mean to estimate the probability function at the candidate solution. This simulation approach is easy to implement and is computationally cheap compared to the optimization procedure. In the numerical experiments conducted in the following section, we suggest checking the feasibility of the solutions using Monte Carlo estimation during implementing the algorithm and embedding it in the stopping criteria.

5. Numerical Illustration

In this section, we study the performance of our method through two numerical examples. The first example is the mean-VaR portfolio management problem that we introduced in §2.2; the second is a VaR-constrained program abstracted from the credit risk model. We use the examples to demonstrate how to implement our approach in real applications.

5.1. Mean-VaR Portfolio Selection

Let us consider the mean-VaR portfolio selection problem in §2.2. Note that Table 1 summarizes the optimal values of problem (3) and its CVaR approximation for different parameter combinations. In the experiments, we use these results as the benchmark. We implement Algorithm CVaR-SCA to solve the problem for all the parameter combinations. Because the problem is sufficiently smooth, we apply the gradient-based approach introduced in §4 to solve the CVaR-like approximations. We use a sample size $n = 20,000$ for $d = 10$ and 50 and $n = 50,000$ for $d = 100$, and implement MATLAB function *fmincon* to conduct optimization with the estimated objective function values, constraint function values, and corresponding estimated gradients. We also use a sample size of 10^5 to check the feasibility of the solution obtained in each iteration. The algorithm is stopped if the difference of objective values of two consecutive iterations is less than or equal to 10^{-4} or if the solution reaches the boundary of the feasible region. For each parameter combination we first randomly conduct experiments five times. The experiments show that the performances across replications are similar. The algorithm typically converges in 5–20 iterations with total computational time varying from a few seconds to a few minutes. We found that even for $d = 100$ and $n = 50,000$, the computation is still fast. This shows the effectiveness of the gradient-based approach. We compute the average values of

Table 2 Recovery of CVaR Approximation

		$\alpha = 0.10$	$\alpha = 0.05$	$\alpha = 0.01$
$d = 10$	$w = 0.05$	0.139/0.420	0.084/0.179	0.048/0.061
	$w = 0.10$	0.280/0.438	0.168/0.345	0.095/0.120
	$w = 0.15$	0.375/0.445	0.253/0.402	0.141/0.184
$d = 50$	$w = 0.05$	0.233/0.486	0.116/0.323	0.062/0.078
	$w = 0.10$	0.435/0.486	0.229/0.465	0.119/0.158
	$w = 0.15$	0.465/0.492	0.349/0.480	0.179/0.240
$d = 100$	$w = 0.05$	0.238/0.493	0.125/0.421	0.053/0.083
	$w = 0.10$	0.460/0.493	0.248/0.483	0.124/0.167
	$w = 0.15$	0.475/0.494	0.381/0.491	0.184/0.255

the five replications and summarize the results in Table 2.

Comparing Table 2 with Table 1, we see that the objective values achieved by both the CVaR approximation and our approach are very close to the true ones. This suggests that the SAA approach is quite effective. It also shows that our approach succeeds to remedy the lost part of the CVaR approximation as well as to converge to the optimal solution of the original VaR-constrained program.

Next we conduct a more extensive numerical study to see the performance of the algorithm. Specifically, we make 1,000 replications for $d = 10$ and 100 replications for $d = 50$. Once we obtain a solution, we plug it into the true objective function to obtain its true objective value. We compute the average values, and the empirical 0.25, 0.50, 0.75 quantile of the replications, and report the results in Table 3. From Table 3 we see that the remedy procedure for the considered example is quite stable.

To further observe the behavior of the proposed remedy procedure, we take a representative combination $d = 50, \alpha = 0.05$, and $w = 0.10$ and report numerical results for this combination. Note that $\alpha = 0.05$ is the most popular risk level in financial risk management and $d = 50$ is a moderately large dimension for a portfolio. We plot a typical simulation run in Figures 1 and 2. In Figure 1, we plot the optimal objective

Table 3 Recovery of CVaR Approximation with Multiple Replications

		$\alpha = 0.10$		$\alpha = 0.05$		$\alpha = 0.01$	
		$d = 10$	$d = 50$	$d = 10$	$d = 50$	$d = 10$	$d = 50$
$w = 0.05$	0.25 quantile	0.420	0.484	0.173	0.309	0.060	0.074
	0.50 quantile	0.421	0.484	0.176	0.323	0.061	0.076
	0.75 quantile	0.422	0.485	0.180	0.336	0.062	0.078
	Average	0.421	0.484	0.176	0.320	0.061	0.076
$w = 0.10$	0.25 quantile	0.439	0.488	0.342	0.464	0.120	0.149
	0.50 quantile	0.440	0.488	0.345	0.465	0.122	0.154
	0.75 quantile	0.440	0.488	0.348	0.466	0.123	0.157
	Average	0.440	0.488	0.345	0.465	0.122	0.153
$w = 0.15$	0.25 quantile	0.448	0.489	0.403	0.479	0.181	0.223
	0.50 quantile	0.448	0.489	0.404	0.479	0.183	0.232
	0.75 quantile	0.449	0.490	0.405	0.479	0.185	0.235
	Average	0.448	0.489	0.404	0.479	0.183	0.229

Downloaded from informs.org by [144.214.42.26] on 02 September 2014, at 22:58 . For personal use only, all rights reserved.

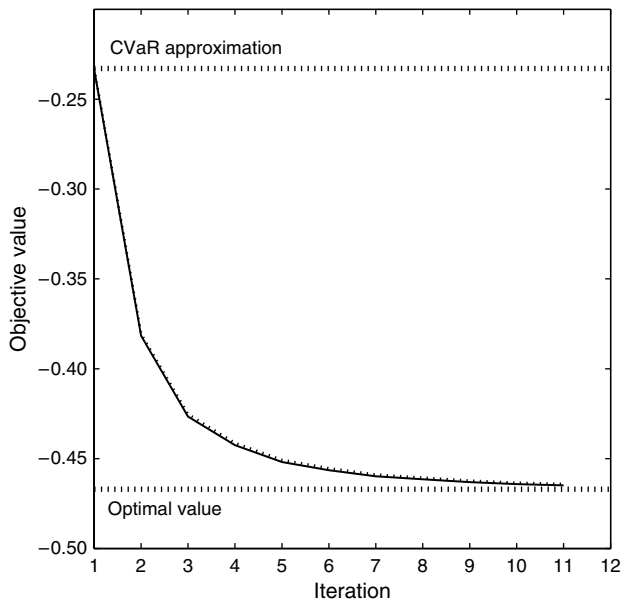


Figure 1 Performance of Algorithm CVaR-SCA for Portfolio Optimization

value of problem (3) and the objective value obtained by the corresponding CVaR approximation (see the dashed horizontal lines).

There are two curves in the figure. The dashed curve is the objective values for all iterations of the algorithm and the solid curve is the objective values obtained by plugging in the solutions generated in the true objective function $-\mu^T x$. Both curves start from the CVaR approximation and approach the optimal objective value. Of course there exists a minor difference between the two curves because of simulation error.

Figure 2 shows the constraint behavior for the algorithm. In the left panel we plot the values $\text{VaR}_{1-\alpha}(-r^T x)$ for the sequence of solutions generated by the algorithm. It can be seen that the VaR constraint is quite loose (about 0.035) for the solution of the CVaR approximation. While Algorithm CVaR-SCA keeps on tightening the VaR constraint until the VaR constraint becomes tight ($w = 0.1$). The right panel is about the probability levels $\Pr\{-r^T x \leq w\}$ for the generated solutions. Similarly, the probability level for the CVaR approximation is about 98%, which is relatively conservative, and our algorithm keeps on relaxing the level until it falls to the target level (95%). Note that in this example the correlations are chosen to be equal and positive. In the online supplement we also consider the case where some correlations are negative. Note also that in this example the closed form of the VaR/chance constraint can be derived. In cases where the constraints cannot be derived analytically, we can still run simulations to track the constraint behavior of the solutions generated by our algorithm.

5.2. Credit Risk Optimization

As a second example we consider a credit risk optimization problem. The example is constructed from Andersson et al. (2001). Suppose there are d obligors and the investor wants to allocate money among these obligors. Credit risk may occur if the obligors fail to fulfill the obligations in full on the due date or thereafter. The investor wants to maximize the expected loan portfolio return while keeping the potential credit loss from obligors' credit migration under some threshold. More specifically, the investor resorts to the following optimization model:

$$\begin{aligned} & \underset{l \leq x \leq u}{\text{minimize}} && -\frac{\sum_{i=1}^d q_i r_i x_i}{\sum_{i=1}^d q_i} \\ & \text{subject to} && \text{VaR}_{1-\alpha}((b - \xi)^T x) \leq w, \end{aligned} \quad (24)$$

$$\begin{aligned} & \sum_{i=1}^d q_i x_i = \sum_{i=1}^d q_i, \\ & q_i x_i \leq 0.20 \sum_{i=1}^d q_i, \quad i = 1, \dots, d. \end{aligned}$$

For a more detailed background of credit risk optimization, readers are referred to Andersson et al. (2001), or the general credit risk literature such as J.P. Morgan's CreditMetrics (Gupton et al. 1997). In problem (24), $x = (x_1, \dots, x_d)^T$ is the vector of obligor weights that the investor wants to determine. Thus, it is the decision vector of the problem. l and u are the trading limits that restrict the change of obligor weights. The vector $b = (b_1, \dots, b_d)^T$ denotes the future values from the obligors in the absence of credit migration, and ξ is the vector of future values from the obligors with credit migration. Therefore, $c(x, \xi) := (b - \xi)^T x$ is the credit loss of the loan portfolio. The first constraint in problem (24) requires that the VaR level of the portfolio loss be below the threshold w . The vector $q = (q_1, \dots, q_d)^T$ denotes the current mark-to-market counterparty values. The second constraint maintains the current value of the loan portfolio. The third constraint restricts each long individual position from exceeding 20% of the total current portfolio value. The expected return from obligor i in the absence of credit migration is r_i . Andersson et al. (2001) suggest that the expected portfolio return can be expressed as $\sum_{i=1}^n q_i r_i x_i / \sum_{i=1}^n q_i x_i$. With the second constraint of problem (24), the expected portfolio return is also equal to $\sum_{i=1}^n q_i r_i x_i / \sum_{i=1}^n q_i$. Thus in problem (24) we actually minimize the negative expected portfolio return.

We consider $d = 50$, $\alpha = 0.05$, $w = 250$, $l = (-2, \dots, -2)^T$, and $u = (2, \dots, 2)^T$, and let r_i evenly spread in $[0.02, 0.12]$. Since the magnitude of q does

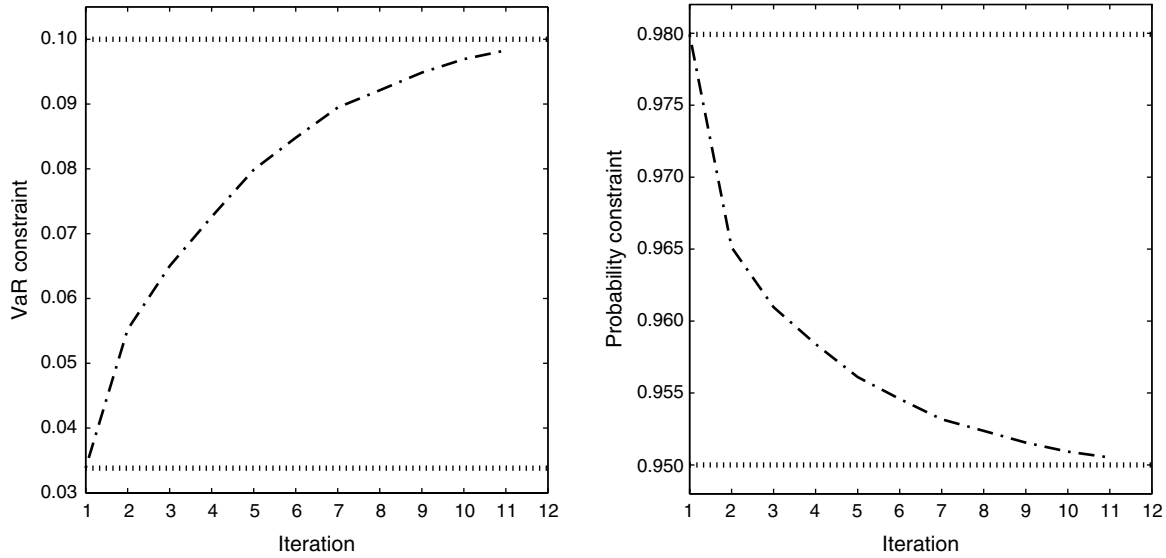


Figure 2 Constraint Behavior of Algorithm CVaR-SCA for Portfolio Optimization

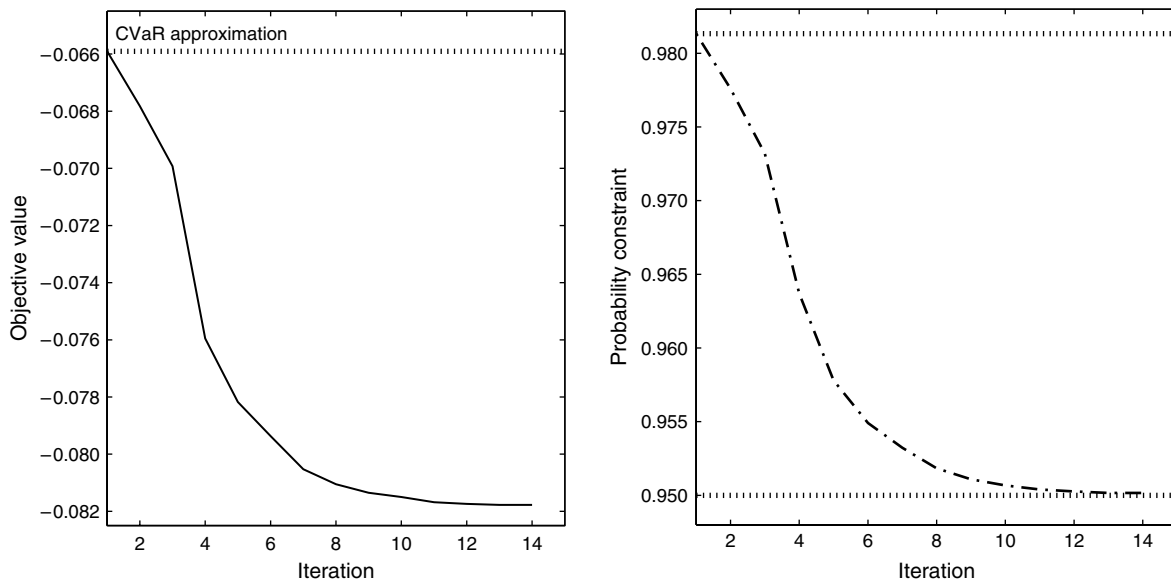


Figure 3 Performances of Algorithm CVaR-SCA for Credit Risk Optimization

not affect the model, we assume that the components of q spread evenly in $[0, 10]$. Following Andersson et al. (2001), we assume the future value vector ξ has a density $p(\xi)$. Because the loss distribution is often skewed and has a long, fat tail (Andersson et al. 2001), we assume $b - \xi$ follows a multivariate log-normal distribution. Specifically, we assume $b - \xi$ can be simulated by $\exp(\eta) - u$, where $u = (u_1, \dots, u_d)^T$ with u_i evenly spreading in $[1, 4]$ and η following a multivariate normal distribution $N(\mu, \Sigma)$. We assume the means $\mu_i, i = 1, \dots, d$ evenly spread in $[0.5, 2]$, the standard deviation of each component of η is half of its mean, and all the correlations between two components are 0.2. Note that in such a setting, it is difficult

to derive the analytical form of problem (24). But we can still use our approach to handle the problem.

As in the first example, we implement the gradient-based approach using the *fmincon* function in MATLAB. The sample size is set as $n = 50,000$ and the stopping rule is the same as that of the preceding example except that we reduce the tolerance level to 10^{-5} . We run Algorithm CVaR-SCA multiple times; the algorithm also shows similar performances.

In Figure 3 we report a typical run. The left panel of the figure shows the convergence of the function values, and the right panel shows the behavior of $\Pr\{(b - \xi)^T x \leq w\}$ values, estimated at the generated solutions. From the plot we see the CVaR

approximation provides an allocation vector of obligor weights whose expected portfolio return is about 6.6%. The probability constraint for this solution is greater than 98%, which is rather conservative. Our algorithm improves on the allocation vector found by the CVaR approximation and finally approaches a new allocation vector with expected portfolio return of approximately 8.2%. At the same time the probability constraint is gradually relaxed to the required level. The results again suggest that our remedy procedure may help achieve much better decisions.

6. Conclusion

In this paper, we have investigated the VaR-constrained program. We reexamine the CVaR approximation that is the best CCA of the VaR constrained program and propose a SCA approach to recover the lost part of the CVaR approximation. The approach starts from the solution found by the CVaR approximation and makes improvement at each iteration by solving a CVaR-like approximation; it finally leads to an optimal solution of the original VaR-constrained program, or a solution essentially binding the VaR constraint. We introduce various approaches that solve the CVaR-like approximation encountered at each iteration and make some comparisons among them. The comparisons provide information that may be useful for selecting techniques to solve CVaR optimization problems. The numerical experiments show that our approach works well.

We have also discussed the role of CVaR in the multiple approaches of handling the VaR/chance constraints. The picture becomes clear now. The CVaR turns out to be a watershed in solving stochastic optimization problems involving VaR/chance constraints. To handle the VaR/chance constraints, the various CCAs of the VaR-constrained program propose to find a good tractable CCA of the CVaR and try to get close to the CVaR. The merit of these CCAs is that they can be solved very efficiently by simple deterministic optimization techniques. However, they may suffer from conservatism. As can be seen, our work is on the other side of the watershed. We start from the CVaR and try to bring in Monte Carlo methods to improve on the CVaR. Compared to the deterministic CCA approach, our approach is not so neat and is often more computationally demanding. But if computational budget is allowed, our approach may obtain improved solutions. In many cases such improvement can be significant and encouraging. Our approach provides a new way of trading the computational effort and the quality of the solutions.

Supplemental Material

Supplemental material to this paper is available at <http://dx.doi.org/10.1287/ijoc.2013.0572>.

Acknowledgments

The authors thank Marvin Nakayama, the associate editor, and two anonymous referees for their valuable comments and suggestions. The research of L. Jeff Hong is partially supported by the Hong Kong Research Grants Council [Projects GRF613410, GRF613011, and N_HKUST626/10]. Zhaolin Hu is the corresponding author and his research is partially supported by National Natural Science Foundation of China (NSFC) [Grant 71201117], NSFC major program [Grant 71090404/71090400], and Shanghai Pujiang Program [Grant 13PJ101]. The research of Liwei Zhang is partially supported by NSFC [Grants 11071029 and 91130007].

References

- Alexander GJ, Baptista AM (2004) A comparison of VaR and CVaR constraints on portfolio selection with the mean-variance model. *Management Sci.* 50:1261–1273.
- Alexander S, Coleman TF, Li Y (2006) Minimizing CVaR and VaR for a portfolio of derivatives. *J. Bank Finance* 30:583–605.
- Andersson F, Mausser H, Rosen D, Uryasev S (2001) Credit risk optimization with conditional value-at-risk criterion. *Math. Programming* 89:273–291.
- Artzner P, Delbaen F, Eber J-M, Heath D (1999) Coherent measures of risk. *Math. Finance* 9(3):203–228.
- Basova HG, Rockafellar RT, Royset JO (2011) A computational study of the buffered failure probability in reliability-based design optimization. *Proc. 11th Conf. Appl. Statist. Probab. Civil Engrg., Zurich, Switzerland*.
- Ben-Tal A, Nemirovski A (2000) Robust solutions of linear programming problems contaminated with uncertain data. *Math. Programming* 88:411–424.
- Ben-Tal A, El Ghaoui L, Nemirovski A (2009) *Robust Optimization*, Princeton Series in Applied Mathematics (Princeton University Press, Englewood Cliffs, NJ).
- Charnes A, Cooper WW, Symonds GH (1958) Cost horizons and certainty equivalents: An approach to stochastic programming of heating oil. *Management Sci.* 4:235–263.
- Chen W, Sim M, Sun J, Teo C-P (2010) From CVaR to uncertainty set: Implications in joint chance-constrained optimization. *Oper. Res.* 58:470–485.
- Chu F, Nakayama MK (2012) Confidence intervals for quantiles when applying variance-reduction techniques. *ACM Trans. Modeling Comput. Simulation* 22(2):article 10.
- Chung H, Polak E, Sastry S (2010) On the use of outer approximations as an external active set strategy. *J. Optim. Theory Appl.* 146:51–75.
- Duffie D, Pan J (1997) An overview of value at risk. *J. Derivatives* 4:7–49.
- El Ghaoui L, Oks M, Oustry F (2003) Worst-case value-at-risk and robust portfolio optimization: A conic programming approach. *Oper. Res.* 51:543–556.
- Gaivoronski A, Pflug G (2005) Value at risk in portfolio optimization: Properties and computational approach. *J. Risk* 7:1–31.
- Gupton GM, Finger CC, Bhatia M (1997) CreditMetrics™—Technical document (J.P. Morgan & Co., New York).
- Hong LJ, Liu G (2009) Simulating sensitivities of conditional value at risk. *Management Sci.* 55:281–293.
- Hong LJ, Yang Y, Zhang L (2011) Sequential convex approximations to joint chance constrained programs: A Monte Carlo approach. *Oper. Res.* 59:617–630.
- Hu Z, Hong LJ, Zhang L (2013) A smooth Monte Carlo approach to joint chance constrained program. *IEEE Trans.* 45:716–735.
- Iyengar G, Ma AKC (2013) Fast gradient descent method for mean-CVaR optimization. *Ann. Oper. Res.* 205:203–212.

- Jorion P (2010) *Financial Risk Manager Handbook*, Sixth ed. (Wiley, New York).
- Krokhmal P, Palmquist J, Uryasev S (2002) Portfolio optimization with conditional value-at-risk objective and constraints. *J. Risk* 4:11–27.
- Lim C, Sherali HD, Uryasev S (2010) Portfolio optimization by minimizing conditional value-at-risk via nondifferentiable optimization. *Computational Optim. Appl.* 46:391–415.
- Nemirovski A (2012) On safe tractable approximations of chance constraints. *Eur. J. Oper. Res.* 219:707–718.
- Nemirovski A, Shapiro A (2006) Convex approximations of chance constrained programs. *SIAM J. Optim.* 17:969–996.
- Ogryczak W, Śliwiński T (2011) On solving the dual for portfolio selection by optimizing conditional value at risk. *Computational Optim. Appl.* 50:591–595.
- Pagnoncelli BK, Ahmed S, Shapiro A (2009) Sample average approximation method for chance constrained programming: Theory and applications. *J. Optim. Theory Appl.* 142:399–416.
- Prékopa A (2003) Probabilistic programming. Ruszczyński A, Shapiro A, eds. *Operations Research and Management Science: Stochastic Programming*, Vol. 10 (Elsevier Science, Amsterdam), 267–352.
- Rockafellar RT, Royset JO (2010) On buffered failure probability in design and optimization of structures. *Reliability Engrg. System Safety* 95:499–510.
- Rockafellar RT, Uryasev S (2000) Optimization of conditional value-at-risk. *J. Risk* 2:21–41.
- Rockafellar RT, Uryasev S (2002) Conditional value-at-risk for general loss distributions. *J. Banking Finance* 26:1443–1471.
- Serfling RJ (1980) *Approximation Theorems of Mathematical Statistics* (Wiley, New York).
- Shapiro A, Dentcheva D, Ruszczyński A (2009) *Lectures on Stochastic Programming: Modeling and Theory* (SIAM, Philadelphia).
- Trindade AA, Uryasev S, Shapiro A, Zrazhevsky G (2007) Financial prediction with constrained tail risk. *J. Banking Finance* 31: 3524–3538.