Comprehensive Exam Paper

David Love

April 29, 2010

Abstract

In this paper, we review a published result on the use of Monte Carlo sampling to solve the chance constrained problem in stochastic programming by Calafiori and Campi. The method of Monte Carlo sampling is shown to generate solutions that are feasible for the chance constrained problem with high probability. Then, we present an original result from our current work that uses Monte Carlo sampling to assess solution quality in another class of stochastic programs.

1 Introduction to Stochastic Programming

In an optimization problem, one seeks the "best" way to accomplish some result, subject to some constraints on the system that affects the decisions. The general optimization problem may be stated as

$$\min_{\mathbf{x}\in\mathcal{X}(\xi)\subseteq\mathbb{R}^n} f(\mathbf{x},\xi) \tag{1}$$

where $f:\mathcal{X}\to\mathbb{R}$ is the objective function, $\mathcal{X}$ is the set of allowed solutions, called the feasible set, $\mathbf{x}$ is a vector of decision variables and ξ is a vector of problem parameters which may affect the objective function or the set of feasible solutions. When $\mathcal{X}$ is a convex set and $f(\mathbf{x},\xi)$ is a convex function in $\mathbf{x}$, then (1) is called a convex program. Convex programs are amenable to numerical solution techniques and also have desired theoretical properties. The deterministic programming problem assumes that this vector ξ is known exactly, and proceeds to solve (1) via one of many known solution techniques. One important special case is the linear program, when f is linear in $\mathbf{x}$, and the feasible set is $\mathcal{X}=\{\mathbf{x}:A\mathbf{x}\leq b,\mathbf{x}\geq 0\}$, for $A\in\mathbb{R}^{m\times n}$ and $b\in\mathbb{R}^m$. Throughout this paper, the notation $\mathbf{u}\leq\mathbf{v}$ for vectors $\mathbf{u}$ and $\mathbf{v}$ denotes element-wise comparison.

The exact value of ξ is often unknown in practice. In this case, one may wish to approximate ξ with a single value (e.g., a point estimate of its "average" value), and proceed with the deterministic programming solution. However, this solution method may result in a solution that is infeasible when the results of the problem are transcribed back into the real world. Another method, reviewed briefly in Section 2.1.2, assumes only that ξ belongs to some set, and proceeds to optimize the worst case scenario. This method suffers from the drawback that all values of ξ are considered to be equal, even those that are believed to be unlikely.

The approach of stochastic programming is to consider the parameter ξ as a random variable $\tilde{\xi}$. (A small notational note: in this paper, we follow the convention that random variables are denoted with a tilde.) This paper examines methods of solving and evaluating solutions to the optimization problem (1) by methods of Monte Carlo sampling of the parameter $\tilde{\xi}$. The original research presented here examines a method of assessing the quality of a proposed solution and assumes that the feasible set $\mathcal{X}$ is independent of $\tilde{\xi}$, while the published work reviewed here seeks to solve problems where the objective function, but not the feasible region, is independent of $\tilde{\xi}$.

The structure of this paper is as follows: Section 2 introduces the uncertain convex program, whose results will be reviewed by this paper; Section 3 reviews some of these recent theoretical results and Section 4 discusses some computational results. Both Section 3 and Section 4 summarize the contributions of Calafiori and Campi [5]. Section 5 briefly discusses some results that have followed from the principal paper reviewed here and Section 6 discusses a research project that has recently been completed. Finally, Section 7 comments on the future path of research.

2 The Uncertain Convex Program

The Uncertain Convex Program (UCP) [5] is a generalization of convex programming where the constraints are not known exactly. The uncertainty is assumed to be parameterized by an instance parameter $\xi \in \Xi \subseteq \mathbb{R}^s$, and we seek to solve the optimization problem

$$\min_{\mathbf{x}\in\mathcal{X}\subseteq\mathbb{R}^n} \left\{c^T\mathbf{x} : g(\mathbf{x},\xi) \leq 0, \forall \xi \in \Xi\right\} \tag{UCP}$$

where $\mathbf{x} \in \mathcal{X}$ is the vector of decision variables, $\mathcal{X}$ is closed, convex and independent of ξ, and $g(\mathbf{x},\xi) : \mathcal{X} \times \Xi \to \mathbb{R}$, the constraint that stores all dependence on the instance parameter, is convex in $\mathbf{x}$ for all ξ. We can assume, without loss of generality, that g is scalar, since we can otherwise replace it with $g(\mathbf{x},\xi) = \max_i\{g_i(\mathbf{x},\xi)\}$. The objective function is assumed to be linear, but can be expanded to a convex function $c(\mathbf{x})$, for example, by epigraphic methods of translating the problem into

$$\min_{\mathbf{x}\in\mathcal{X}\subseteq\mathbb{R}^n} \left\{\gamma : g(\mathbf{x},\xi) \leq 0, \xi \in \Xi, c(\mathbf{x}) - \gamma \leq 0\right\}.$$

2.1 Solutions Methods

2.1.1 Chance Constrained Problem

The Chance Constrained Program (CCP), first introduced for linear problems by Charnes and Cooper in 1959 [7], introduces a probability measure $\mathbb{P}$ to the uncertain convex program (UCP). This measure could represent a known distribution of the stochastic instance parameter $\tilde{\xi}$, or it could represent a weighting of the relative importance of the possible values of $\tilde{\xi}$. Let $\epsilon \in [0,1]$ be the level of acceptable risk of violating the constraint g. Then the chance constrained problem is given by

$$\min_{\mathbf{x}\in\mathcal{X}\subseteq\mathbb{R}^n} \left\{c^T\mathbf{x} : \mathbb{P}\left[g(\mathbf{x},\tilde{\xi}) > 0\right] \leq \epsilon\right\}. \tag{CCP}$$

One immediate difficulty of solving (CCP) is that evaluating $\mathbb{P}\left[g(\mathbf{x},\tilde{\xi}) > 0\right]$ typically involves computing an integral in multiple dimensions, a computationally intensive task when s, the dimension of $\tilde{\xi}$, is sufficiently large. Another significant difficulty to solving (CCP) is that, even with the assumptions placed on (UCP), the chance constrained problem may be non-convex. Sen and Higle [19] provide a simple example of such a problem,

$$\min_{\mathbf{x}\in\mathbb{R}^2} \left\{x_1 + x_2 : \mathbb{P}\left[2x_1 + x_2 \geq \tilde{\xi}_1, x_1 + 2x_2 \geq \tilde{\xi}_2\right] \geq 0.5\right\} \tag{2}$$

where the dependent random variables $\tilde{\xi}_1$ and $\tilde{\xi}_2$ each have a Bernoulli distribution, with $\tilde{\xi}_2 = 1 - \tilde{\xi}_1$. The feasible region is shown in Figure 1. The upper right region of the plot contains values of $\mathbf{x}$ that are feasible with probability 1, and satisfy the equations $2x_1 + x_2 \geq \max\tilde{\xi}_1 = 1$ and $x_1 + 2x_2 \geq \max\tilde{\xi}_2 = 1$. The two remaining strips represent regions where one of $\tilde{\xi}_1$, $\tilde{\xi}_2$ assumes a value of zero.

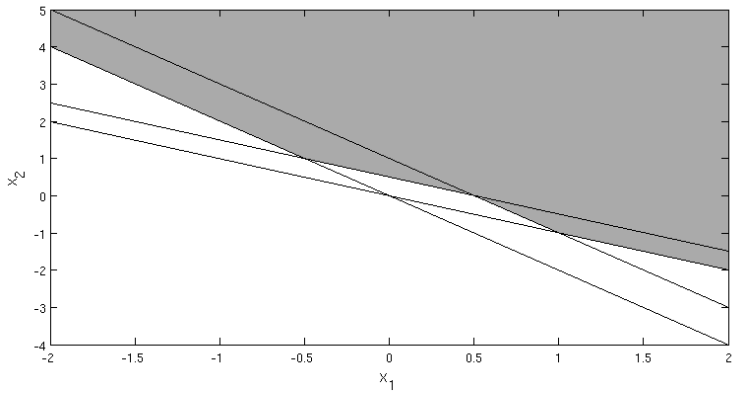


Figure 1: A graphical representation of feasible region of problem (2). Note that in this chance constrained programming problem, the feasible region is nonconvex.

Definition 1. *A set S is said to be a **cone** if $\exists v \in S$ and, for every $\mathbf{x} \in S$ and $\lambda > 0$, $v + \lambda(\mathbf{x} - v) \in S$.*

Note that the feasible region in Figure 1 has the form of a union of two cones. Dencheva [9] shows that this form of the feasible region generalizes to a wider class of problems, where the randomness is on the right-hand side of the constraints, i.e.,

$$\max_{\mathbf{x} \in \mathcal{X}} c^T \mathbf{x} \tag{3}$$
$$\text{subject to } \mathbb{P}\left[g(\mathbf{x}) < \tilde{\xi}\right] \leq \epsilon,$$

where $g(\mathbf{x}) = (g_1(\mathbf{x}), \ldots, g_s(\mathbf{x}))$ and $g_i(\mathbf{x})$ are convex functions for $i = 1, \ldots, s$. Define the p-level set of the distribution function of $\tilde{\xi}$

$$\mathcal{S}_p = \left\{ d \in \mathbb{R}^s : \mathbb{P}\left[\tilde{\xi} \leq d\right] \geq p \right\}.$$

Then, letting $p = 1 - \epsilon$, we can rewrite (3) as

$$\max_{\mathbf{x} \in \mathcal{X}} c^T \mathbf{x} \tag{4}$$
$$\text{subject to } g(\mathbf{x}) \in \mathcal{S}_{1-\epsilon}. \tag{5}$$

This changes our focus from avoiding "bad" values of $\tilde{\xi}$ to seeking "good" values of $\tilde{\xi}$, but this change is necessary for the next definition (Definition 6 from [9]).

Definition 2. *Let $p \in (0, 1]$. A point $v \in \mathbb{R}^s$ is called a p-efficient point of the probability distribution function F if $F(v) \geq p$ and there is no $z \leq v$, $z \neq v$ such that $F(z) \geq p$.*

Then the set of p-efficient points of the p-level set $\mathcal{S}_p$ is the set of the "smallest" points in $\mathcal{S}_p$ with respect to the partial order of $\mathbb{R}^s$ given by component-wise comparisons. Clearly, the set of p-efficient points is a singleton for every $p \in (0, 1]$ if $\tilde{\xi}$ is scalar valued. However, for higher dimensions, there is no limit on the number of p-efficient points for a general $\tilde{\xi}$—indeed, there could be infinitely many p-efficient points (i.e., the set of p-efficient points could be unbounded and not closed) [9].

Let $v^j, j \in J$ be an enumeration of all p-efficient points of $\mathcal{S}_p$, and $K_j = \{\mathbf{x} \in \mathbb{R}^s : \mathbf{x} \geq v^j\}$ be the cones centered at v^j and extending into the positive orthant, then Theorem 14 of [9] shows that $\mathcal{S}_p = \bigcup_{j \in J} K_j$. This clearly explains the structure of the feasible set of (2) in Figure 1. The feasible set of (2) is given by the union of two cones with centers $v^1 = (-1/3, 2/3)$ and $v^2 = (2/3, -1/3)$, given by $K_j = \left\{\mathbf{x} : \mathbf{x} = v^j + \lambda_1(1,2)^T + \lambda_2(2,1)^T, \lambda_1, \lambda_2 \geq 0\right\}$.

2.1.2 Robust Optimization

A more direct method of obtaining a solution to (UCP) is the Robust Convex Program (RCP)

$$\min_{\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^n} \left\{c^T \mathbf{x} : \mathbf{x} \in \mathcal{X} \cap K\right\} \tag{RCP}$$

where

$$K \equiv \bigcap_{\tilde{\xi} \in \Xi} \{\mathbf{x} : g(\mathbf{x}, \xi) \leq 0\}.$$

The (RCP) has the advantage that $\mathcal{X} \cap K$ is always a convex set, unlike the feasible region of (CCP). However, (RCP) is in general a semi-infinite program, i.e., a problem requiring an infinite number of constraints but a finite number of decision variables. Such problems are difficult to solve numerically, and methods rely on the use of some relaxations of the problem.

One obvious problem with the robust approach is that $\mathcal{X} \cap K$, the set of solutions that are feasible for every instance parameter, might be empty. This may be a problem if, for example, at least one component of $\tilde{\xi}$ is supported on the entire real line. Calafiori and Campi in the paper reviewed here [5] assume that $\mathcal{X} \cap K$ is nonempty.

2.2 Sampled Convex Program

Calafiori and Campi propose the use of the Sampled Convex Program (SCP) as a method of ameliorating the difficulties involved in the above two solution methods. The problem is given by

$$\min_{\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^n} \left\{c^T \mathbf{x} : g(\mathbf{x}, \tilde{\xi}^{(i)}) \leq 0, i = 1, \ldots, N\right\}, \tag{SCP$_N$}$$

where $\tilde{\xi}^{(i)}, i = 1, \ldots, N$ are independent and identically distributed (iid) samples from the distribution of the random variable $\tilde{\xi}$.

In their examination of (SCP$_N$), Calafiori and Campi initially assume that each sampled program has a unique optimal solution, and relax the assumption to cases when (SCP$_N$) has multiple optimal solutions, or is unbounded in a later extension. The assumption in Section 2.1.2, $\mathcal{X} \cap K \neq \emptyset$ guarantees that (SCP$_N$) will have a feasible solution, since (SCP$_N$) uses a subset of the constraints of (RCP).

3 Recent Theoretical Results

We begin with a probability space $(\Omega, \mathcal{F}, \hat{\mathbb{P}})$, and the random variables $\tilde{\xi} : (\Omega, \mathcal{F}) \to (\Xi, \mathcal{D})$, with distribution $\mathbb{P}$. Then, we define the probability of violating the constraint, (Definition 1 of [5]),

Definition 3. *Let $\mathbf{x} \in \mathcal{X}$ be a candidate solution to the uncertain convex program (UCP). The violation probability is*

$$V(\mathbf{x}) \equiv \mathbb{P}\left[\xi \in \Xi : g(\mathbf{x}, \xi) > 0\right]$$

Calafiori and Campi here use the additional assumption that $\{\xi \in \Xi : g(\mathbf{x}, \xi) > 0\}$ is a measurable set for all $\mathbf{x} \in \mathcal{X}$. This assumption is not particularly strong, and will be satisfied in the natural case of $\mathcal{D} = \mathcal{B}(\mathbb{R}^s)$, with g Borel in $\tilde{\xi}$ for every $\mathbf{x}$. Note that g is already Borel in $\mathbf{x}$, since it is assumed convex in $\mathbf{x}$ for every $\tilde{\xi}$. Clearly, the sampled program (SCP_N) seeks, in part, to determine a candidate solution with sufficiently small $V(\mathbf{x})$. This leads us to the next definition (Definition 2 of [5]):

Definition 4. *Let $\epsilon \in [0, 1]$. $\mathbf{x} \in \mathcal{X}$ is an ϵ-level solution if $V(\mathbf{x}) \leq \epsilon$.*

Note that the set $\{\mathbf{x} \in \mathcal{X} : V(\mathbf{x}) \leq \epsilon\}$ is the feasible set for the chance constrained problem (CCP), so any ϵ-level solution is feasible for (CCP).

Let $\mathbf{x}_N^*$ be the optimal solution to (SCP_N), a random variable that depends on the sampled constraints $\tilde{\xi}^{(1)}, \ldots, \tilde{\xi}^{(N)}$. The two key results of the paper (Theorem 1 and Corollary 1 of [5]) are provided next.

Theorem 5. *Let $\mathbf{x}_N^*$ be the optimal solution to (SCP_N). Then*

$$\mathbb{E}_{\mathbb{P}^N}\left[V(\mathbf{x}_N^*)\right] \leq \frac{n}{N+1}$$

where n is the size of the decision vector $\mathbf{x}$, and $\mathbb{P}^N (= \mathbb{P} \times \cdots \times \mathbb{P}, N$ times) is the probability measure on the space Ξ^N of the multi-sample extraction $\tilde{\xi}^{(1)}, \ldots, \tilde{\xi}^{(N)}$.

Corollary 6. *Fix two real numbers $\epsilon \in [0, 1]$ (level parameter) and $\beta \in [0, 1]$ (confidence parameter) and let*

$$N \geq \frac{n}{\epsilon\beta} - 1. \tag{6}$$

Then, with probability no smaller than $1 - \beta$, the randomized problem (SCP_N) returns an optimal solution $\mathbf{x}_N^*$ which is ϵ-level robustly feasible.

3.1 Discussion of Results

Two things are particularly striking about these results. First, the expected solution level of $\mathbf{x}_N^*$ decreases hyperbolically with the number of samples N, and increases only with the size of the decision vector $\mathbf{x}$. Second, neither of these results depends on the probability measure $\mathbb{P}$. One aspect of particular interest is that the sampled program can be applied to find ϵ-level solution even if the probability distribution of $\mathbb{P}$ is not known exactly—assuming that the distribution can be sampled from. Some further results, briefly discussed in Section 5, have been dedicated to finding good (though likely non-optimal) solutions without the ability to sample from an unknown distribution $\mathbb{P}$.

The number of constraints required by the algorithm increases linearly with $1/\epsilon\beta$, as well as with the size of the decision vector. For example, a standard statistical choice of $\epsilon = \beta = 0.05$ yields approximately 400 constraints per dimension of $\mathbf{x}$. This number, though it can be large for some problems, can be managed by current convex solvers for reasonable values of ϵ and β.

The probability measure in the sampled problem inherits its role of governing the sampling of $\tilde{\xi}$ from the chance constrained problem (CCP), and takes on the additional role in the sampled problem (SCP_N) as the probability with which the quality of the solution is assessed. The role of quality assessment may be particularly important when $\mathbb{P}$ is determined by weighing the relative importance of the instance parameters, rather than the frequency with which instance parameters occur.

In addition to the possibility that $\mathcal{X} \cap K = \emptyset$ (which was ignored for the purposes of the paper), one downside of solving the robust program (RCP) over (CCP) (or approximating its solution with (SCP_N)) is that a small subset of Ξ (with respect to $\mathbb{P}$) may significantly decrease the size of the feasible set or otherwise change the objective value significantly. Calafiori and Campi provide the simple example

$$\min_{x \in \mathbb{R}} \{x : g(\mathbf{x}, \tilde{\xi}) \leq 0\}$$

where

$$g(x, \tilde{\xi}) = \begin{cases} \frac{1}{\alpha} - x, & \tilde{\xi} \in [0, \alpha] \\ -x, & \tilde{\xi} \in (\alpha, 1], \end{cases}$$

$\tilde{\xi}$ is uniformly distributed over $\Xi = [0, 1]$, and α is a small positive number. Choosing a probability of infeasibility $\epsilon \geq \alpha$, it is clear that $x^* = 0$ is an ϵ-level feasible solution. However, restricting the feasible region to that of (RCP), the feasible region becomes $x \geq 1/\alpha$, leading to an optimal solution of $x^* = 1/\alpha$. Then, for smaller values of α, one pays an increasingly steep price for the inclusion of the set of "bad" parameters $\tilde{\xi} \in [0, \alpha]$.

An optimal solution to the ϵ-level problem (CCP) will always outperform (i.e., will have a smaller value of $c^T\mathbf{x}$) an optimal solution to the robust problem, because the set of ϵ-level solutions contains the entire feasible region for (UCP). Thus, the choice of ϵ is dictated by the competing desires of feasibility (with high probability) and performance of the optimal solution.

In light of these competing desires, it is important to remember that the solution generated with $N \geq n/\epsilon\beta - 1$ samples may exceed the guarantee of being an ϵ-level robustly feasible solution with probability at least $1 - \beta$. The level of the candidate solution $\mathbf{x}_N^*$ can be efficiently tested with Monte Carlo sampling. Let $\hat{V}_M(\mathbf{x}_N^*)$ be an empirical probability of constraint violation, i.e., $\hat{V}_M(\mathbf{x}_N^*) = \frac{1}{M} \sum_{i=1}^{M} \mathbf{1}_{g(\mathbf{x}_N^*, \tilde{\xi}^{(i)}) \leq 0}$, where $\mathbf{1}$ is the indicator function, and $\tilde{\xi}(i)$, $i = 1, \ldots, M$ is batch of iid samples of $\tilde{\xi}$ that is independent of the samples used to determine $\mathbf{x}_N^*$. Then the Hoeffding inequality states that

$$\mathbb{P}^N \left[\left| \hat{V}_M(\mathbf{x}_N^*) - V(\mathbf{x}_N^*) \right| \leq \epsilon' \right] \geq 1 - 2e^{-2(\epsilon')^2 M}.$$

Then $\left| \hat{V}_M(\mathbf{x}_N^*) - V(\mathbf{x}_N^*) \right| \leq \epsilon'$ holds with probability at least $1 - \beta'$ if $M \geq \frac{\log 2/\beta'}{2(\epsilon')^2}$. This test is an efficient means of assessing solution quality, since no additional optimization problems need to be solved. An example case where the the solution found by solving (SCP_N) is significantly better than ϵ-level robustly feasible is presented in Section 4.2.

3.2 Proof of Theorem

To show the main result, Calafiori and Campi prove a theorem bounding the number of support constraints for a convex program. Define the convex programs

$$\min_{\mathbf{x} \in \mathbb{R}^n} \left\{ c^T\mathbf{x} : \mathbf{x} \in \bigcap_{i=1}^{m} \mathcal{X}_i \right\}, \tag{P}$$

where $\mathcal{X}_i$, $i = 1, \ldots, m$ are closed convex sets. Then the problems P_k, $k = 1, \ldots, m$ are obtained by removing the constraint $\mathbf{x} \in \mathcal{X}_k$, i.e.,

$$\min_{\mathbf{x} \in \mathbb{R}^n} \left\{ c^T\mathbf{x} : \mathbf{x} \in \bigcap_{i=1, i \neq k}^{m} \mathcal{X}_i \right\}. \tag{P_k}$$

Let $\mathbf{x}^*$ and $\mathbf{x}_k^*$ be optimal solution to problems (P) and (P_k), respectively.

Definition 7. *The k-th constraint $\mathcal{X}_k$ is a support constraint for (P) if problem (P_k) has an optimal solution $\mathbf{x}_k^*$ such that $c^T\mathbf{x}_k^* < c^T\mathbf{x}^*$.*

Then Calafiori and Campi go on to prove (Theorem 2 of [5])

Theorem 8. *The number of support constraints for problem (P) is at most n.*

The proof of this theorem is not instructive, so it is omitted. However, we can gain some intuition about why it should be true. Consider the special case of the linear program, i.e., when $\mathcal{X}_i = \{\mathbf{x} : a_i^T\mathbf{x} \leq b_i\}$, for $a_i \in \mathbb{R}^n, b_i \in \mathbb{R}$. It is well known that the optimal solution $\mathbf{x}^*$ occurs on a vertex of $\cap_{i=1}^m \mathcal{X}_i$, and that there are at least n linearly independent constraints $\mathcal{X}_i$ such that $a_i^T\mathbf{x}^* = b_i$ (see Definition 2.9 and Theorem 2.3 of [4]). Without loss of generality, suppose these constraints are $\mathcal{X}_1, \ldots, \mathcal{X}_n$, and the other constraints satisfied at equality (if they exist) are $\mathcal{X}_{n+1}, \ldots, \mathcal{X}_r$. Then constraints $\mathcal{X}_{n+1}, \ldots, \mathcal{X}_r$ can be removed, but $\mathbf{x}^*$ will still be a vertex of the polyhedron. If we consider a general convex feasible set, this intuition is modified to note that (1) the optimal solution need not occur at a vertex (so there will be fewer than n supporting constraints) and (2) convex sets have at least one tangent hyperplane at every point on the boundary.

Using the above theorem, we prove Theorem 5.

Proof of Theorem 5. The proof proceeds in two steps. First, we wish to find an unbiased statistical estimator of $\mathbb{E}[V(\mathbf{x}_N^*)]$, where $\mathbf{x}_N^*$ is the optimal solution of (SCP_N), which we denote as $\hat{V}_N$. We then show that $\hat{V}_N \leq \frac{n}{N+1}$ a.s.

Let $\tilde{z}^{(1)}, \ldots, \tilde{z}^{(N+1)}$ be $N+1$ independent samples from the distribution $\mathbb{P}$ on Ξ, which are independent of the samples $\tilde{\xi}^{(1)}, \ldots, \tilde{\xi}^{(N)}$ used to formulate (SCP_N). We look at the $N+1$ instances of the problem (SCP_N)

$$\min_{\mathbf{x}\in\mathcal{X}\subseteq\mathbb{R}^n} \left\{ c^T\mathbf{x} : g(\mathbf{x}, \tilde{z}^{(i)}) \leq 0, i = 1, \ldots, k-1, k+1, \ldots, N+1 \right\}, \qquad (\text{SCP}_N^k)$$

for each $k = 1, \ldots, N+1$. Let $(\mathbf{x}_N^*)^k$ be the optimal solution to (SCP_N^k), and note that $(\mathbf{x}_N^*)^k$ may not satisfy the omitted constraint, $g((\mathbf{x}_N^*)^k, \tilde{z}^{(k)}) \leq 0$. Let the indicator function $v_k \equiv \mathbf{1}_{g((\mathbf{x}_N^*)^k, \tilde{z}^{(k)})>0}$ be a random variable that determines when the omitted constraint is violated. Then

$$\begin{aligned}
\mathbb{E}_{\mathbb{P}^{N+1}}[v_k] &= \mathbb{E}_{\mathbb{P}^N}\left[\mathbb{E}_{\mathbb{P}}\left[v_k | \tilde{z}^{(1)}, \ldots, \tilde{z}^{(k-1)}, \tilde{z}^{(k+1)}, \ldots, \tilde{z}^{N+1)}\right]\right] \\
&= \mathbb{E}_{\mathbb{P}^N}\left[\mathbb{P}\left[z \in \Xi : g((\mathbf{x}_N^*)^k, \tilde{z}^{(k)}) > 0\right]\right] \\
&= \mathbb{E}_{\mathbb{P}^N}\left[V((\mathbf{x}_N^*)^k)\right] \\
&= \mathbb{E}_{\mathbb{P}^N}[V(\mathbf{x}_N^*)],
\end{aligned}$$

where the last equality follows from the fact that (SCP_N^k) is simply one realization of (SCP_N). Then we define our estimator

$$\hat{V}_N = \frac{1}{N+1}\sum_{k=1}^{N+1} v_k$$

and we can clearly see that our estimator is unbiased, i.e.,. $\mathbb{E}_{\mathbb{P}^{N+1}}\left[\hat{V}_N\right] = \mathbb{E}_{\mathbb{P}^N}[V(\mathbf{x}_N^*)]$.

To show that the estimator $\hat{V}_N$ is bounded above, let $z^{(1)}, \ldots, z^{(N+1)}$ be a realization of $\tilde{z}^{(1)}, \ldots, \tilde{z}^{(N+1)}$, and consider the sampled convex program with all constraints

$$\min_{\mathbf{x}\in\mathcal{X}\subseteq\mathbb{R}^n} \left\{ c^T\mathbf{x} : g(\mathbf{x}, z^{(i)} \leq 0, i = 1, \ldots, N+1 \right\}, \qquad (\text{SCP}_{N+1})$$

with $\mathbf{x}^*_{N+1}$ being its optimal solution. Removing a single constraint $g(\mathbf{x}, z^{(k)}) \leq 0$ from (SCP_{N+1}) will improve the objective, and thus change the optimal solution, only if $g(\mathbf{x}, z^{(k)}) \leq 0$ is a support constraint. We know by Theorem 8 that this occurs for at most n values of $k \in \{1, \ldots, N+1\}$. Then for at most n values of k, the optimal solution to (SCP^k_N) differs from (SCP_{N+1}). Finally, this difference can only occur if the missing constraint $g(\mathbf{x}, z^{(k)}) \leq 0$ is violated. Thus $v_k = 1$ at most n times, and $\hat{V} \leq \frac{n}{N+1}$, a.s. Therefore $\mathbb{E}_{\mathbb{P}^N}[V(\mathbf{x}^*_N)] \leq \frac{n}{N+1}$. □

4 Computational Results

4.1 Robust Linear Programs

Calafiori and Campi first test their method on an uncertain convex program for which the exact solution can be easily computed. For this, they choose an uncertain linear program

$$\min_{x \in \mathcal{X} = \mathbb{R}^n} \left\{ c^T \mathbf{x} : A(\xi)\mathbf{x} \leq b, \forall \xi \in \Xi \right\}$$

where $A(\xi) \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$. In the case when $A(\xi)$ is affine in ξ, and Ξ is the direct product of ellipsoids, the uncertain linear program can be exactly represented as a convex program with a finite number of decision variables and constraints. Calafiori and Campi assume that each row a_i^T of $A(\xi)$ has the form $a_i(\xi) = \hat{a}_i + E_i\xi_i$, $i = 1, \ldots, m$, with $\|\xi_i\| \leq 1$, where $\hat{a}_i$ is the center of the ellipsoid and $E_i \in \mathbb{R}^{n \times n}$ is the symmetric "shape" matrix. Then the ith constraint becomes

$$\hat{a}_i^T \mathbf{x} + \xi_i^T E^i \mathbf{x} \leq b_i, \forall \|\xi_i\| \leq 1$$

which holds if and only if

$$\max_{\|\xi_i\| \leq 1} \{\hat{a}_i^T \mathbf{x} + \xi_i^T E^i \mathbf{x}\} \leq b_i$$

which holds if and only if $\hat{a}_i^T \mathbf{x} + \|E^i \mathbf{x}\| \leq b_i$. This formulation then turns the uncertain linear program into a second order cone program, with m constraints and n decision variables.

To implement the sampled convex program, Calafiori and Campi assume that the random variable $\tilde{\xi}$ is distributed uniformly over the ball $\|\xi\| \leq 1$, and choose $n = 2$ decision variables. They use the matrix

$$A(\xi) = \begin{bmatrix} -1 & 0 \\ 0 & -1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} + 0.2 \begin{bmatrix} \xi_1^T \\ \xi_2^T \\ \xi_3^T \\ \xi_4^T \end{bmatrix},$$

with $b = [0, 0, 1, 0]^T$ and $c = [-1, -1]^T$. The solution to the robust problem is then computed as $\mathbf{x}^* = [0.7795, 0.7795]^T$ with optimal cost $c^T\mathbf{x}^* = -1.5590$. The sampled convex program is generated using $\epsilon = \beta = 0.01$, which requires $N = 19,999$ samples, for a random linear program

$$\min_{\mathbf{x} \in \mathbb{R}^2} \{c^T \mathbf{x} : A(\tilde{\xi}^{(i)})\mathbf{x} \leq b, i = 1, \ldots, N\},$$

which found an optimal solution of $\mathbf{x}^*_N = [0.7798, 0.7795]^T$ and optimal cost $c^T\mathbf{x}^*_N = -1.5594$, a mere 0.025% different from the optimal solution to the robust problem.

4.2 Robust Least-Squares

In the general least-squares problem, we wish to find a polynomial of degree of at most $n-1$ that interpolates m given points (a_i, y_i), $i = 1, \ldots, m$, such that it has the minimal squared error $\|Ax - y\|^2$, where

$$A = \begin{bmatrix} 1 & a_1 & \cdots & a_1^{n-1} \\ \vdots & \vdots & & \vdots \\ 1 & a_m & \cdots & a_m^{n-1} \end{bmatrix}, \mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}, y = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}.$$

The solution to this problem is well known when the data (a_i, y_i) are known exactly. However, there is no known efficient algorithm for data that follows a distribution. Calafiori and Campi assume that the y_i values are deterministic, while the $a_i(\tilde{\xi})$ are uniformly distributed on the interval $[a_i - \rho, a_i + \rho]$, and that they would like to solve the robust program

$$\min_{\mathbf{x} \in \mathbb{R}^n} \max_{\tilde{\xi} \in \Xi} \|A(\tilde{\xi})\mathbf{x} - y\|^2.$$

Calafiori and Campi compare an upper bound solution technique from [10] to the results of their chance constrained sampling algorithm. For $m = n = 3$, $\rho = 0.2$ and $(a_1, y_1) = (1, 1)$, $(a_2, y_2) = (2, -0.5)$ and $(a_3, y_3) = (4, 2)$. Choosing $\epsilon = \beta = 0.1$, requires $N = 299$ samples of $\tilde{\xi}$. The sampled convex program resulted in a residual error that was roughly 40% better than the upper bound solution, and the a posteriori technique described above gives high confidence that the chance of constraint violation is less than 0.006—significantly better than the ϵ that was chosen to start the algorithm.

5 Further Results

The paper reviewed here [5] has inspired a number of further results on the use of Monte Carlo sampling in chance constrained problems. I briefly discuss some of these results below.

Calafiori and Ghaoui have continued to investigate problems of this nature, for example in [6]. Here, the assumptions on the problem structure are significantly tightened, so that the feasible region is now described by the linear system $g(\mathbf{x}, \tilde{\xi}) = \tilde{A}\mathbf{x} + \tilde{b} \leq 0$, where $\tilde{A} \in \mathbb{R}^{m \times n}$, $\tilde{b} \in \mathbb{R}^m$, and the instance parameter is $\tilde{\xi} = (\tilde{A}, \tilde{b})$. Calafiori and Ghaoui focus on a single constraint of this form and want it to be feasible with probability at least $1 - \epsilon$, i.e., $\mathbb{P}\left[\tilde{a}^T\mathbf{x} + \tilde{b} \leq 0\right] \geq 1 - \epsilon$. They show that, in the case that $\tilde{d} = [\tilde{a}^T, \tilde{b}]^T$ has a Q-radial distribution, the chance constrained linear program is equivalent to a deterministic program. By a Q-radial distribution, they mean that the random variable $\tilde{d} - \mathbb{E}\left[\tilde{d}\right] = Q\tilde{w}$, where Q is a deterministic matrix, and $\tilde{w}$ has a distribution that depends only on its Euclidean norm, $\|\tilde{w}\|$—i.e., $\phi_w(w) = h(\|w\|)$.

More significantly, Calafiori and Ghaoui begin to investigate the case when the distribution of $\tilde{\xi}$ is not known exactly. They investigate the case when the mean and covariance of the random vector $\tilde{d}$ are known exactly, and the case when $\mathbb{E}\left[\tilde{d}\right]$ is known, the $\tilde{d}_i$ belongs to the finite interval $\left[\mathbb{E}\left[\tilde{d}\right] - l_i^-, \mathbb{E}\left[\tilde{d}\right] + l_i^+\right]$. In each of these cases, they find deterministic programs whose optimal solutions satisfy the chance constraint with probability 1. The deterministic constraints are not linear, but they are second order cone constraints, which have the form

$$\kappa_\epsilon \|B(\mathbf{x}^T, 1)^T\| + \mathbb{E}\left[d^T\right]\mathbf{x} \leq 0$$

where κ_ϵ and $B \in \mathbb{R}^{n+1\times n+1}$ are parameters that differ in the two cases listed above.

Another important result comes from Erdoğan and Iyengar [11], who investigate the case when the distribution of $\tilde{\xi}$ is not known exactly, and we are unable to sample from it, called the "ambiguous chance constrained problem" (ACCP). In this case, the distribution $\mathbb{P}$ is known only to belong to some family of distributions $\mathcal{P}$, the set of probability measures sufficiently similar to some central distribution $\mathbb{P}_0$. This family is defined by the Prohorov metric ρ_p, i.e., $\mathcal{P} = \{\mathbb{P} : \rho_p(\mathbb{P}, \mathbb{P}_0) \leq \beta\}$ for some $\beta > 0$. The central distribution can be chosen, for example, to be one that is theoretically well known that fits empirical observations well. The problem then is

$$\min_{\mathbf{x}\in\mathcal{X}\subseteq\mathbb{R}^n} \left\{ c^T\mathbf{x} : \mathbb{P}\left[g(\mathbf{x},\tilde{\xi}) > 0\right] \leq \epsilon, \forall \mathbb{P} \in \mathcal{P} \right\}. \tag{ACCP}$$

The algorithm of Erdoğan and Iyengar uses Monte Carlo sampling of the central distribution $\mathbb{P}_0$ to construct a result similar to (6) for (ACCP). The theoretical result draws from several areas of probability theory. In particular, the notion of Vapnik-Chervonenkis (VC) dimension from statistical learning theory is heavily employed in the algorithm's method of finding the feasible region of the chance constrained problem. In this context, the VC dimension functions as a way to measure the complexity of the sampled problem.

The theoretical justification of the algorithm comes from the Strassen–Dudley Representation Theorem, dealing with couplings of random variables. Random variables $\tilde{\xi}_1, \tilde{\xi}_2$ are said to be coupled if and only if they have the same distribution. The Strassen–Dudley theorem (Theorem 4 of [11]) is

Theorem 9. *Let $\tilde{\xi}_1 \sim \mathbb{P}_1$ and $\tilde{\xi}_2 \sim \mathbb{P}_2$ be two random variables taking values in Ξ. If $\rho_p(\mathbb{P}_1, \mathbb{P}_2) \leq \beta$, then there exists a coupling $(\tilde{\xi}_1', \tilde{\xi}_2')$ of $(\tilde{\xi}_1, \tilde{\xi}_2)$ such that $\mathbb{P}\left[\|\tilde{\xi}_1' - \tilde{\xi}_2'\|\right] \leq \beta$.*

6 Overlapping Batches

6.1 Introduction

We now turn our attention to an original result on the use of Monte Carlo sampling to determine the quality of a proposed optimal solution to a (non-chance constraint) stochastic program. We consider a stochastic program of the form

$$z^* = \min_{\mathbf{x}\in X} \mathbb{E}\left[f(\mathbf{x},\tilde{\xi})\right], \tag{SP}$$

where $\mathbf{x}$ is a vector of decision variables, $\tilde{\xi}$ is a vector of random variables, and X is the feasible set, which is assumed to be independent of $\tilde{\xi}$. Further, it is assumed that $X \neq \emptyset$ and compact, the distribution of $\tilde{\xi}$ is known and that we can sample from it. Even though we will impose a more restrictive moment condition later, we assume that $\mathbb{E}\left[f(\mathbf{x},\tilde{\xi})\right]$ is well defined and finite for all $\mathbf{x} \in X$ and $f(\cdot,\tilde{\xi})$ is lower semicontinuous on X, almost surely (a.s.). This implies that (SP) has a finite optimal solution achieved on X.

Let $\tilde{\xi}^1, \tilde{\xi}^2, \ldots, \tilde{\xi}^n$ be an independent and identically distributed (iid) sample from the distribution of $\tilde{\xi}$. A sampling approximation of (SP) is given by

$$z_n^* = \min_{\mathbf{x}\in X} \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x},\tilde{\xi}^i). \tag{SP$_n$}$$

We denote an optimal solution to (SP) as $\mathbf{x}^*$ and an optimal solution to (SP$_n$) as $\mathbf{x}_n^*$.

We are interested in assessing the quality of a candidate solution $\hat{\mathbf{x}} \in X$ (e.g., $\hat{\mathbf{x}} = \mathbf{x}_n^*$), defined by its optimality gap $\mathbb{E}\left[f(\hat{\mathbf{x}}, \tilde{\xi})\right] - z^*$. This is important in practice since (SP) cannot be solved exactly, so one only has an approximate solution $\hat{\mathbf{x}}$ without verification of its quality. Assessing solution quality is also an indispensable part of stopping criteria in algorithms. For stochastic programs, one way to estimate optimality gaps is through Monte Carlo sampling. A straightforward estimate of $\mathbb{E}\left[f(\hat{\mathbf{x}}, \tilde{\xi})\right]$ is the sample mean, $\frac{1}{n}\sum_{i=1}^{n} f(\hat{\mathbf{x}}, \tilde{\xi}^i)$, so, optimality gap estimators depend on a lower bound estimator on z^*. A widely applicable lower bound that does not depend on a particular solution method is $\mathbb{E}[z_n^*] \leq z^*$ and this bound improves as the sample size increases, e.g., $\mathbb{E}[z_n^*] \leq \mathbb{E}\left[z_{n+1}^*\right] \leq z^*$ [15, 17].

A point estimator of the optimality gap of $\hat{\mathbf{x}} \in X$ can be obtained using sample size m through the above bounds, $G_m(\hat{\mathbf{x}}) = \frac{1}{m}\sum_{i=1}^{n} f(\hat{\mathbf{x}}, \tilde{\xi}^i) - z_m^*$. Here, we assume the same observations $\tilde{\xi}^1, \ldots, \tilde{\xi}^m$ are used in both terms of $G_m(\hat{\mathbf{x}})$. Computing $G_m(\hat{\mathbf{x}})$ involves solving an optimization problem (SP_n) to obtain a lower bound estimator z_m^*, which complicates the statistical analysis. To enable statistical inference, the *multiple replications procedure* (MRP) of Mak, Morton and Wood [15] generates k independent estimators of $G_m(\hat{\mathbf{x}})$ and averages them to obtain a point estimator. To form confidence intervals, the sample variance of these estimators is calculated. This is essentially a (nonoverlapping) batch means estimator (k batches of size m) commonly used in simulation [14].

6.2 Background

6.2.1 Overlapping Batch Means

Consider a covariance stationary stochastic process which has mean μ and variance σ^2. The task is to estimate μ given some realization of the stochastic process $\mathbf{y} = (y^1, y^2, \ldots, y^n)$. Typically, this involves forming a confidence interval of the form $[L_\alpha(\mathbf{y}), U_\alpha(\mathbf{y})]$ for a given significance level $1-\alpha$ such that $\mathbb{P}[L_\alpha(\mathbf{y}) \leq \mu \leq U_\alpha(\mathbf{y})] = 1-\alpha$. The usual estimator for μ is the sample mean $\bar{\bar{\mathbf{y}}} = \frac{1}{n}\sum_{i=1}^{n} y^i$, which is an unbiased estimator even in the presence of correlated data. However, the usual variance estimator $\frac{1}{n-1}\sum_{i=1}^{n}(y^i - \bar{\bar{\mathbf{y}}})^2$ can be severely biased in this case, resulting in inaccurate interval estimators. Note that output from simulation analysis is typically correlated. To overcome this difficulty, various variance estimators have been proposed in the simulation literature, including nonoverlapping and overlapping batch means [14].

Nonoverlapping batch means takes n observations, $y^1, \ldots, y^n$ of the stochastic process and splits them into k batches of size m, where $k = \lfloor \frac{n}{m} \rfloor$. The sample mean of each batch is then computed as

$$\bar{\mathbf{y}}_j = \frac{1}{m}\sum_{i=1}^{m} y^{m(j-1)+i},$$

for $j = 1, \ldots, k$, and the overall sample mean $\bar{\bar{\mathbf{y}}} = \frac{1}{k}\sum_{j=1}^{k} \bar{\mathbf{y}}_j = \frac{1}{n}\sum_{i=1}^{n} y^i$ provides a point estimator of μ. The sample variance of the batch means is calculated as

$$\widehat{\mathrm{Var}}(\bar{\bar{\mathbf{y}}}) = \frac{m}{n}\frac{1}{k-1}\sum_{j=1}^{k}\left(\bar{\mathbf{y}}_j - \bar{\bar{\mathbf{y}}}\right)^2,$$

and the $(1-\alpha)$-level approximate confidence interval is then formed by $\bar{\bar{\mathbf{y}}} \pm t_{k-1,\alpha/2}\sqrt{\widehat{\mathrm{Var}}(\bar{\bar{\mathbf{y}}})}$, where $t_{k-1,\alpha/2}$ denotes the $1-\alpha/2$ quantile of the Student's t distribution with $k-1$ degrees of freedom.

Overlapping batch means modifies this idea by taking batches given by

$$\bar{\mathbf{y}}_j(m) = \frac{1}{m}\sum_{i=1}^{m} y^{(j-1)+i},$$

for $j = 1, \ldots, n-m+1$. This is what we call the *maximally-overlapping* batch means method, because $m-1$ observations are common to, and only one observation changes between, adjacent batches. Given this, the sample variance estimator is updated to

$$\widetilde{\mathrm{Var}}\,(\bar{\bar{\mathbf{y}}}) = \sum_{j=1}^{n-m+1} \frac{(\bar{\mathbf{y}}_j(m) - \bar{\bar{\mathbf{y}}})^2}{(\frac{n}{m}-1)(n-m+1)}. \tag{7}$$

Meketon and Schmeiser [16] show several attractive properties of the overlapping batches variance estimator, the two most important of which are: (i) the overlapping variance estimator has nearly the same bias as the standard nonoverlapping variance estimator, and (ii) the overlapping estimator has only two-thirds of the asymptotic variance of the nonoverlapping one. That is,

$$\frac{\mathrm{Var}\left(\widetilde{\mathrm{Var}}\,(\bar{\bar{\mathbf{y}}})\right)}{\mathrm{Var}\left(\widehat{\mathrm{Var}}(\bar{\bar{\mathbf{y}}})\right)} \to \frac{2}{3} \tag{8}$$

in the limit as batch size m and then the number of batches (n/m) tend to infinity. Note that the degrees of freedom in (7) is slightly different than in [16]. The degrees of freedom in (7) makes $\widetilde{\mathrm{Var}}\,(\bar{\bar{\mathbf{y}}})$ an unbiased estimator for iid data for all finite m and n with the same asymptotic benefits [21].

Welch [23] established the relationship between OBM and spectral estimators and also considered partial overlap, see also [21]. In our application to assessment of solution quality, we also consider varying the amount of overlap between neighboring batches. We defer this discussion on variable overlap to Section 6.3.1 and continue with a brief review of MRP.

6.2.2 Multiple Replications Procedure

Given a candidate solution $\hat{\mathbf{x}} \in X$ to a stochastic programming problem (SP), the task is to estimate its optimality gap $\mathbb{E}\left[f(\hat{\mathbf{x}}, \tilde{\xi})\right] - z^*$. In many fields of optimization, it is common to evaluate an upper bound on this quantity through relaxations such as integrality or Lagrangian relations. In stochastic programming, given a sample size m, an upper bound on the optimality gap can be obtained by $\mathbb{E}\left[f(\hat{\mathbf{x}}, \tilde{\xi})\right] - \mathbb{E}\left[z_m^*\right] \geq \mathbb{E}\left[f(\hat{\mathbf{x}}, \tilde{\xi})\right] - z^*$ due to the inequality $\mathbb{E}\left[z_m^*\right] \leq z^*$. MRP uses this bound to construct a confidence interval (CI) on the optimality gap using the nonoverlapping batches method described above. Again, taking n observations of the random variables and splitting them into k batches of batch size m, we define (for a candidate solution $\hat{\mathbf{x}}$)

$$\bar{G}_j = \frac{1}{m}\sum_{i=1}^{m} f(\hat{\mathbf{x}}, \tilde{\xi}^{m(j-1)+i}) - \min_{\mathbf{x}\in X} \frac{1}{m}\sum_{i=1}^{m} f(\mathbf{x}, \tilde{\xi}^{m(j-1)+i}) \tag{9}$$

for $j = 1, \ldots, k$. We are assessing the quality of a given solution $\hat{\mathbf{x}} \in X$, therefore, we suppress it from the notation. As before, $\bar{\bar{G}} = \frac{1}{k}\sum_{j=1}^{k} \bar{G}_j$ provides a point estimator of $\mathbb{E}\left[f(\hat{\mathbf{x}}, \tilde{\xi})\right] - z^*$. The sample variance is obtained similarly by $\widehat{\mathrm{Var}}\left(\bar{\bar{G}}\right) = \frac{1}{k}\frac{1}{k-1}\sum_{j=1}^{k}(\bar{G}_j - \bar{\bar{G}})^2$, which results in a one-sided CI $\left[0, \bar{\bar{G}} + t_{k-1,\alpha}\sqrt{\widehat{\mathrm{Var}}\left(\bar{\bar{G}}\right)}\right]$ that has a significance level which is approximately $1-\alpha$.

Notice that the use of z_m^* in the $\bar{G}_j$ gives rise to a biased gap estimator. Thus we can expect that the true probability of the optimality gap residing within the confidence interval to be greater than the $1-\alpha$ suggested by the above calculation. This is shown empirically in [1], which also suggests additional methods for using a smaller number of replications (e.g., 1 or 2) with an alternative variance estimator to compute a confidence interval. See also [2] for variations of MRP aimed to reduce bias and variance.

An advantage of MRP is its applicability to a wide range of problems. With iid sampling, (SP) can be linear or nonlinear, X can include integrality constraints or not. It is also easy to implement, thus, has been applied to a variety of problems, see e.g. [3, 13, 18]. Recently, similar ideas have been used for assessing solution quality of stochastic programs with (finitely many) expected value [22] and stochastic dominance [12] constraints.

6.3 Overlapping Multiple Replications Procedure

Our aim is to apply the idea of overlapping batches to MRP. We note several differences in this setting compared to the simulation setting. In simulation, the point of interest is estimating the variance of the sample mean of a covariance stationary process. We are interested in estimating the variance of an optimality gap estimator. The main difference comes from the fact that an optimality gap estimator not only has a sample mean ($\frac{1}{m}\sum_{i=1}^{m} f(\hat{\mathbf{x}}, \tilde{\xi}^i)$) but also a minimized sample mean (z_m^*). First, minimization changes the statistical properties of sample means. For instance, the central limit theorem may not hold for a minimized sample mean even though it holds for each $x \in X$. We overcome this difficulty by approximating the optimality gap estimators by their non-optimized counterparts (see Section 6.4). The non-optimized counterparts have the desired statistical properties and we establish convergence (in mean square) of the optimality gap estimators to their non-optimized counterparts. Second, once the data is generated through a simulation, it can be reused without much additional computational effort to obtain the OBM variance estimator. In our setting, due to the solution of a sampling problem (SP_m), the computational effort can increase with data reuse. Fortunately, near-optimal variance reduction can be obtained by partially overlapping the batches. Partial overlap results in a fewer number of batches, hence fewer number of optimization problems need to be solved. Moreover, in many solution methods, warm-starting can be used to solve sampling approximations with overlapping samples, considerably reducing solution time.

We begin our discussion with variably overlapping batches and then define the estimators of OMRP.

6.3.1 Variably Overlapping Batches

As before, let m denote the batch size, n the total sample size and $k = \lfloor \frac{n}{m} \rfloor$ be the number of nonoverlapping batches. In this paper, we use the batch nonoverlap parameter $1 \leq \gamma \leq m$ to denote how much neighboring batches do not overlap. For instance, $\gamma = m$ corresponds to the classical case of nonoverlapping batches and $\gamma = 1$ corresponds to the maximally overlapping case of Meketon and Schmeiser. The sample mean of each batch estimator is calculated similarly, $\bar{\mathbf{y}}_j(m,\gamma) = \frac{1}{m}\sum_{i=1}^{m} y^{\gamma(j-1)+i}, j = 1, 2, \ldots, \lfloor \frac{n-m}{\gamma} \rfloor + 1$, where $\lfloor \frac{n-m}{\gamma} \rfloor + 1$ is the number of batches used given n, m and γ. The sample variance estimator given in (7) is changed to

$$\widetilde{\text{Var}}_\gamma\left(\bar{\bar{\mathbf{y}}}\right) = \sum_{j=1}^{\lfloor \frac{n-m}{\gamma} \rfloor + 1} \frac{(\bar{\mathbf{y}}_j(m,\gamma) - \bar{\bar{\mathbf{y}}})^2}{\left(\frac{n}{m} - 1\right)\left(\lfloor \frac{n-m}{\gamma} \rfloor + 1\right)}.$$

The amount of variance reduction in this estimator depends on the asymptotic ratio of γ/m, which we denote by $\bar{\gamma}$. For example, when $\gamma = 1$ (maximally overlapping case), the variance is reduced to two-thirds (66.67%) of the original nonoverlapping case, as given in (8). When only 75% of observations overlap (i.e., when $\bar{\gamma} = 1/4$), the variance is 33/48th of the original (68.75%), which is near-optimal. When only half of the observations overlap ($\bar{\gamma} = 1/2$) the variance is 75% of original. In general, from the spectral analysis given in [23], if $\bar{\gamma} = 1/N$ for integer N, then the variance is reduced to $\frac{2N^2+1}{3N^2}$ of the original.

6.3.2 Definition of Estimators for OMRP

In order to apply overlapping batches to MRP, we need to keep track of solutions to sampling problem (SP_m) for each batch of size m. Toward this end, let $N(i)$ denote the set of batches $j \in \{1, 2, \ldots, \lfloor \frac{n-m}{\gamma} \rfloor + 1\}$ observation $\tilde{\xi}^i$ is used in, $i = 1, 2, \ldots, n$. $|N(i)|$ then gives the number of batches observation is $\tilde{\xi}^i$ is used in. As an example, consider $m = 5$ and $\gamma = 2$ (see Figure 2). The first batch uses observations $\tilde{\xi}^1, \tilde{\xi}^2, \tilde{\xi}^3, \tilde{\xi}^4, \tilde{\xi}^5$, the second batch uses $\tilde{\xi}^3, \tilde{\xi}^4, \tilde{\xi}^5, \tilde{\xi}^6, \tilde{\xi}^7$, and so on. So $N(1) = \{1\}$, $N(2) = \{1\}$ and $N(3) = \{1, 2\}$. We use $\mathbf{x}_j^*$ to denote an optimal solution to sampling problem (SP_m) formed using the jth batch; $\mathbf{x}_j^* \in \text{argmin}_{\mathbf{x}\in X} \frac{1}{m} \sum_{i=1}^{m} f(\mathbf{x}, \tilde{\xi}^{\gamma(j-1)+i})$, $j = 1, 2, \ldots, \lfloor \frac{n-m}{\gamma} \rfloor + 1$.

The results on overlapping batch means occur in the limit as $n, m, k \to \infty$ [8, 16, 21, 23]. In particular, we need $m = O(n^r)$ for some $r \in (0, 1/2)$ (see, e.g., [8]). Let n_l, m_l and k_l be sequences of numbers satisfying these requirements, then the limits will be taken as $l \to \infty$. In addition, we may desire that the batch nonoverlap parameter change with l, to ensure that $\gamma_l/m_l = \bar{\gamma}_l$ converges to the constant $\bar{\gamma}$. Now we are ready to define the estimators for OMRP.

$$\bar{G}_j(m_l, \gamma_l) = \frac{1}{m_l} \sum_{i=1}^{m_l} f(\hat{\mathbf{x}}, \tilde{\xi}^{\gamma_l(j-1)+i}) - \frac{1}{m_l} \sum_{i=1}^{m_l} f(\mathbf{x}_{i,l}^*, \tilde{\xi}^{\gamma_l(j-1)+1}), \tag{10}$$

$$\bar{\bar{G}}_l = \frac{1}{n_l} \sum_{i=1}^{n_l} \frac{1}{|N(i)|} \sum_{j \in N(i)} \left[f(\hat{\mathbf{x}}, \tilde{\xi}^i) - f(\mathbf{x}_j^*, \tilde{\xi}^i) \right], \tag{11}$$

$$VG_l = \frac{1}{(\frac{n_l}{m_l} - 1)(\lfloor \frac{n_l - m_l}{\gamma_l} \rfloor + 1)} \sum_{j=1}^{\lfloor \frac{n_l - m_l}{\gamma_l} \rfloor + 1} (\bar{G}_j(m_l, \gamma_l) - \bar{\bar{G}}_l)^2. \tag{12}$$

The optimality gap estimator for each batch $\bar{G}_j(m_l, \gamma_l)$ is defined like 9 for general values of the nonoverlap parameter γ_l. We just remove the minimization in 9 and use directly the optimal solution of $\mathbf{x}_j^*$ of batch j. The overall mean, $\bar{\bar{G}}_l$, is defined a little differently. Here, $\bar{\bar{G}}_l$ still uses each observation $i = 1, 2, \ldots, n_l$ but also makes use of all the information collected throughout the batches. That is, if observation $\tilde{\xi}^i$ is used in $|N(i)|$ batches, all the optimal solutions $\mathbf{x}_j^*$ corresponding to each batch $j \in N(i)$ are used for the lower bound estimator. Then, VG_l is defined in a similar fashion for the variable overlapping batches variance estimator.

Before we continue, we mention a notation we use for clarity in the rest of the paper. When it is necessary to specify the amount of nonoverlap explicitly, we will write VG_l as $VG_l(\gamma_l)$. For example, we might write the variance-reduction result from (8) as

$$\lim_{l \to \infty} \frac{\text{Var}\left(VG_l(1)\right)}{\text{Var}\left(VG_l(m_l)\right)} = \frac{2}{3}.$$

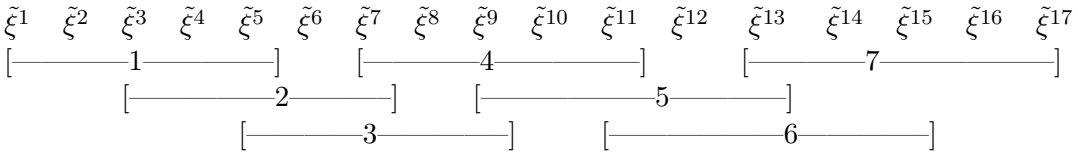


Figure 2: A visual representation of overlapping batches with $n = 17$, $m = 5$ and $\gamma = 2$. This illustrates the sets required in equations (11) and (12). The brackets represent show which random samples are used in each batch, and the numbers inside each bracket show the batch number j.

6.4 Theoretical Results

The internal optimization in the batches in (10) makes a straightforward statistical analysis of the behavior of the estimators difficult. To overcome this problem, we introduce the following unbiased optimality gap estimators

$$\bar{D}_j(m_l, \gamma_l) = \frac{1}{m_l} \sum_{i=1}^{m_l} f(\hat{\mathbf{x}}, \tilde{\xi}^{\gamma_l(j-1)+i}) - \frac{1}{m_l} \sum_{i=1}^{m_l} f(\mathbf{x}^*, \tilde{\xi}^{\gamma_l(j-1)+i}) \tag{13}$$

$$\bar{\bar{D}}_l = \frac{1}{n_l} \sum_{i=1}^{n_l} \left[f(\hat{\mathbf{x}}, \tilde{\xi}^i) - f(\mathbf{x}^*, \tilde{\xi}^i) \right] \tag{14}$$

$$VD_l = \frac{1}{(\frac{n}{m} - 1)(\lfloor \frac{n-m}{\gamma} + 1 \rfloor)} \sum_{j=1}^{\lfloor \frac{n_l - m_l}{\gamma_l} \rfloor + 1} (\bar{D}_j(m_l, \gamma_l) - \bar{\bar{D}}_l)^2. \tag{15}$$

These are essentially the same as the variably overlapping batches estimators in Section 6.3.1 with $y^i = f(\hat{\mathbf{x}}, \tilde{\xi}^i) - f(\mathbf{x}^*, \tilde{\xi}^i)$. These are also defined identically to the original estimators (10)-(12) with the exception that $\mathbf{x}_j^*$ from (10) is replaced by $\mathbf{x}^*$ in (13). With $\hat{\mathbf{x}}$ and $\mathbf{x}^*$ fixed, then estimators (13)-(15) have the same statistical properties as variably overlapping batches estimators in section 6.3.1. They are used only to show convergence of the estimators (10)-(12)—they are not necessary for carrying out the OMRP algorithm.

To show our theoretical results, we make the assumptions about our system:

A1 Samples of the random variable $\tilde{\xi}$ are iid.

A2 $z_n^* \to z^*$, a.s, as $n \to \infty$.

A3 $\exists \epsilon > 0$ such that $\mathbb{E}\left[(\sup_{\mathbf{x} \in X} f(\mathbf{x}, \tilde{\xi}))^{4+\epsilon} \right] < \infty$.

First, we restrict our attention to iid sampling (A1). Then, we require optimal objective function values from sampling problems z_n^* converge to $\mathbf{x}^*$, a.s. via A2. Assumption A2 is not uncommon in the literature and a significant amount of work has gone into documenting the conditions under which this assumption is true; see e.g. [20]. We are interested in variance of the variance estimator, so, we need (at least) the fourth moment of the objective function. Assumptions A3 warrants this, and is used as a sufficient condition for uniform integrability. Our proof proceeds by showing that $VG_l \xrightarrow{L^2} VD_l$, and showing that the probability that the true gap $\mu_{\hat{\mathbf{x}}}$ is covered by the confidence interval formed by $\bar{\bar{G}}$ and $\bar{\bar{D}}$ become asymptotically equal.

6.4.1 Convergence of Overlapping Variance Estimators

We wish to replicate the results on variance reduction from [16, 23] for stochastic programs. The standard results for overlapping batches apply to the variance estimators calculated in (15), so we wish to show that $VG_l - VD_l \xrightarrow{L^2} 0$ as $l \to \infty$, under the assumption that both variance terms use the same random variables and amount of overlap. For brevity, we will abbreviate the convergence statement as $VG_l \xrightarrow{L^2} VD_l$. We present below an outline of this proof below. We show in Theorem 13 that $VG_l \xrightarrow{L^2} VD_l$, using results derived in Lemmas 10, 11 and 12. Validity of confidence intervals is established in Theorem 16.

Lemma 10. *Assume that assumptions A1 and A2 hold, $m = O(n^r)$ for some $r \in (0, 1/2)$ and $\bar{\gamma} = \gamma/m = 1/N$ for some integer N. Then $\bar{\bar{G}}_l \to \bar{\bar{D}}_l$ as $l \to \infty$*

Lemma 11. *Suppose assumption A3 holds. Then $(G_j(m_l, \gamma_l))^4$ is uniformly integrable.*

Lemma 12. *Suppose that the previous two results hold. Then $VG_l \xrightarrow{P} VD_l$*

Theorem 13. *Assume that the previous lemmas hold. Then $VG_l \xrightarrow{L^2} VD_l$.*

Corollary 14. *The variance reduction results from [16, 23] are valid for stochastic programs.*

Corollary 15. *The bias of VG_l remains essentially unchanged across values of $\bar{\gamma}_l$.*

The overlapping variance estimator for stochastic program displays the same desirable properties as in simulation (Theorem 13 and Corollaries 14 and 15 guarantee this) , but we must still check that it results in a valid confidence interval. Let $I_G^l = \bar{\bar{G}}_l + t_{d,\alpha} VG_l$ and $I_D^l = \bar{\bar{D}}_l + t_{d,\alpha} VD_l$ be the one-sided confidence intervals generated by each estimator. The degrees of freedom d is calculated using the method given in [23].

Theorem 16. $\mathbb{P}\left[\mu_{\hat{\mathbf{x}}} \leq I_G^l\right] \to \mathbb{P}\left[\mu_{\hat{\mathbf{x}}} \leq I_D^l\right]$.

Then Theorem 16 guarantees that our procedure produces asymptotically valid confidence intervals.

6.5 Computational Results

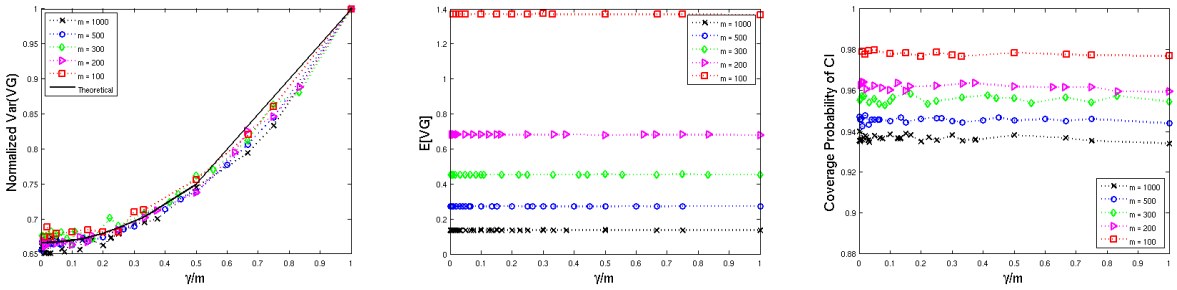
We tested the effectiveness of OMRP on the newsvendor problem. The newsvendor problem asks the optimal number of newspapers to buy, when newspapers are purchased from the publisher at a cost c, sold to the public at a price r, and the demand $\tilde{d}$ for papers is unknown. The problem is then formulated as

$$\max_{x \geq 0} \mathbb{E}\left[r \min\{x, \tilde{d}\} - cx\right].$$

For our test, we chose $c = 5$, $r = 15$, and $\tilde{d} \sim U(0, 10)$.

For all test problems we used the sampling scheme $n = 30m$, with a several values of the batch size m. Results were compiled over 10000 independent runs of the OMRP algorithm.

The reduction of variance for the newsvendor problem is shown in Figure 3a, with each term being normalized with respect to $\mathrm{Var}\,(VG(1))$. The solid line in the figure shows the theoretical result from Welch [23] (reviewed in Section 6.3.1), computed at the points $1, 1/2, 1/3, \ldots$. Empirical results agree with the theory extremely well. In Figure 3b we have estimates of $\mathbb{E}\,[VG]$ for changing batch size and amount of overlap. Consistent with previous results from simulation, we can see that the expectation of the variance estimator is not changed with increasing overlap. Figure 3c shows the coverage probability of confidence intervals generated by OMRP for several values of m. We can see that for many values of γ/m the coverage probability is not greatly affected by changing the amount of overlap.



(a) Variance reduction (b) Expected Variance (c) Coverage Probability

Figure 3: (a) Reduction in variance of VG for various values of γ. The estimate of each $\mathrm{Var}\left(VG(\gamma/m)\right)$ is normalized by $\mathrm{Var}\left(VG(1)\right)$. The solid line shows the variance reduction result given by [23] for a variable amount of overlap. Note that $\gamma/m = 1$ is the case of nonoverlapping batches. The amount of overlap increases for smaller values of γ/m. (b) Estimates of $\mathbb{E}\left[VG\right]$ as a function of γ. We can see that VG is nearly constant across multiple values of γ, which agrees with the results from simulation. (c) Coverage probability of the confidence intervals generated for different relative values of γ. The coverage probability is nearly constant over many different values of γ. This agrees with previous results from simulation, that we should get asymptotically valid confidence intervals for any choice of γ.

7 Ongoing and Future Research Plans

The first step in my plans is to submit my results on overlapping batches for publication. I will also continue my literature review of the previous results in chance constrained problems, especially those results briefly discussed in Section 5 and more recent results in ambiguous chance constrained problems. At the beginning of this phase, I will be paying attention to mastering the computational learning theory previously used, and the use of probability metrics.

References

[1] G. Bayraksan and D. P. Morton. Assessing solution quality in stochastic programs. *Mathematical Programming*, 108(2):495–514, 2006.

[2] G. Bayraksan and D.P. Morton. Assessing solution quality in stochastic programs via sampling. In *Tutorials in Operations Research*, volume 5, pages 102–122. 2009.

[3] M. Bertocchi, J. Dupačová, and V. Moriggia. Sensitivity of bond portfolio's behavior with respect to random movements in yield curve: A simulation study. *Annals of Operations Research*, 99:267–286, 2000.

[4] D. Bertsimas and J.N. Tsitsiklis. *Introduction to linear optimization*. Athena Scientific, Belmont, MA, 1997.

[5] G. Calafiore and MC Campi. Uncertain convex programs: randomized solutions and confidence levels. *Mathematical Programming*, 102(1):25–46, 2005.

[6] G.C. Calafiore and L.E. Ghaoui. On distributionally robust chance-constrained linear programs. *Journal of Optimization Theory and Applications*, 130(1):1–22, 2006.

[7] A. Charnes and WW Cooper. Chance-constrained programming. *Management Science*, 6(1):73–79, 1959.

[8] H. Damerdji. Strong consistency of the variance estimator in steady-state simulation output analysis. *Mathematics of Operations Research*, 19(2):494–512, 1994.

[9] D. Dentcheva. Optimization Models with Probabilistic Constraints. In G. Calafiore and F. Dabbene, editors, *Probabilistic and randomized methods for design under uncertainty*, pages 49–98. Springer Verlag, 2006.

[10] L. El Ghaoui and H. Lebret. Robust solutions to least-squares problems with uncertain data. *SIAM Journal on Matrix Analysis and Applications*, 18:1035–1064, 1997.

[11] E. Erdoğan and G. Iyengar. Ambiguous chance constrained problems and robust optimization. *Mathematical Programming*, 107(1):37–61, 2006.

[12] J. Hu, T. Homem de Mello, and S. Mehrotra. Sample average approximation for stochastic dominance constrained programs. *Optimization Online, http://www.optimization-online.org*, 2009.

[13] U. Janjarassuk and J. T. Linderoth. Reformulation and sampling to solve a stochastic network interdiction problem. *Networks*, 52:120–132, 2008.

[14] A.M. Law. *Simulation Modeling and Analysis*. McGraw-Hill, 4 edition, 2007.

[15] W. Mak, D. P. Morton, and R. K. Wood. Monte carlo bounding techniques for determining solution quality in stochastic programs. *Operations Research Letters*, 24(1-2):47 – 56, 1999.

[16] M.S. Meketon and B. Schmeiser. Overlapping batch means: Something for nothing? In S. Shapiro, U. Pooch, and D Pegder, editors, *Proceedings of the Winter Simulation Conference*, pages 226–230. IEEE Press, 1984.

[17] V.I. Norkin, G.Ch. Pflug, and A. Ruszczyński. A branch and bound method for stochastic global optimization. *Mathematical Programming*, 83:425–450, 1998.

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

[19] S. Sen and J.L. Higle. An introductory tutorial on stochastic linear programming models. *Interfaces*, pages 33–61, 1999.

[20] A. Shapiro. Monte Carlo sampling methods. In A. Ruszczyński and A. Shapiro, editors, *Handbooks in Operations Research and Management Science, Volume 10: Stochastic Programming*, pages 353–425. Elsevier, 2003.

[21] W. T. Song and B. W. Schmeiser. Variance of the Sample Mean: Properties and Graphs of Quadratic-Form Estimators. *Operations Research*, 41(3):501–517, 1993.

[22] W. Wang and S. Ahmed. Sample average approximation of expected value constrained stochastic programs. *Operations Research Letters*, 36:515–519, 2008.

[23] P. D. Welch. On the relationship between batch means, overlapping means and spectral estimation. In A. Thesen, H. Grant, and W. D. Kelton, editors, *Proceedings of the Winter Simulation Conference*, pages 320–323, New York, NY, USA, 1987. IEEE Press.