INFORMS TutORials in Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Assessing Solution Quality in Stochastic Programs via Sampling

Güzin BayraksanDavid P. Morton

To cite this entry: Güzin BayraksanDavid P. Morton. Assessing Solution Quality in Stochastic Programs via Sampling. *In* INFORMS TutORials in Operations Research. Published online: 14 Oct 2014; 102-122.
https://doi.org/10.1287/educ.1090.0065

TUTORIALS IN
OPERATIONS RESEARCH
INFORMS 2009

 | ISBN 978-1-877640-24-7
DOI 10.1287/educ.1090.0065

Assessing Solution Quality in Stochastic Programs via Sampling

Güzin Bayraksan
Systems and Industrial Engineering, University of Arizona, Tucson, Arizona 85721,
guzinb@sie.arizona.edu

David P. Morton
Graduate Program in Operations Research, University of Texas at Austin, Austin, Texas 78712,
morton@mail.utexas.edu

Abstract Determining if a solution is optimal or near optimal is fundamental in optimization theory, algorithms, and computation. For instance, Karush-Kuhn-Tucker conditions provide necessary and sufficient optimality conditions for certain classes of problems, and bounds on optimality gaps are frequently used as part of optimization algorithms. Such bounds are obtained through Lagrangian, integrality, or semidefinite programming relaxations. An alternative approach in stochastic programming is to use Monte Carlo sampling-based estimators on the optimality gap. In this tutorial, we present a simple, easily implemented procedure that forms a point and interval estimator on the optimality gap of a given candidate solution. We then discuss methods to reduce the computational effort, bias, and variance of our simplest estimator. We also provide a framework that allows the use these optimality gap estimators in an algorithmic way by providing rules to iteratively increase the sample sizes and to terminate. This scheme can be used as a stand-alone sequential sampling procedure, or it can be used in conjunction with a variety of sampling-based algorithms to obtain a solution to a stochastic program with a priori control on the quality of that solution.

Keywords stochastic programming; Monte Carlo simulation

1. Introduction

Assessing whether a candidate solution is an optimal or near-optimal solution plays a prominent role in optimization. Depending on the nature of its objective function and constraints, a stochastic program can often be categorized as a linear program, a smooth nonlinear program, a nonsmooth convex program, etc. So, optimality conditions for stochastic programs can frequently be borrowed from their deterministic counterparts. The pitfall with this approach is that for all but the simplest of stochastic programs, the function values and (sub)gradients needed to test these conditions cannot be computed.

Our tutorial circumvents this difficulty. The price to be paid is that the resulting tests are inexact, in three senses. First, instead of testing whether a solution satisfies conditions that ensure optimality, we bound a candidate solution's optimality gap. In optimization, optimistic bounds arise via relaxations, e.g., a Lagrangian relaxation, an integrality relaxation, or a semidefinite programming relaxation. These help bound, in a deterministic manner, the gap between the objective function value of a feasible candidate solution, $\hat{x}$, and the optimal value, with a typical output being

- `Output 1:` $\hat{x}$ `has an objective function value within 1% of the optimal value.`

However, we cannot evaluate the objective function exactly in our stochastic programs, even for a fixed candidate solution, $\hat{x}$. So, the second sense in which our results are inexact

is that errors arise from the Monte Carlo (MC) sampling schemes we use to form statistical estimators. We view the true optimality gap as a deterministic unknown parameter, which we estimate with both point and interval estimators. The typical form for an exact probabilistic statement on an interval estimator would be

- `Output 2: the probability that` $\hat{x}$`'s objective function value is suboptimal by more than 1% is at most 0.05.`

Even in much simpler settings, such statements with *exact* probabilities can be difficult to achieve. Instead, an interval estimator is justified by an asymptotically valid probability statement, i.e., a probabilistic statement that holds as the sample size grows large. In this case, we either say that Output 2 holds for sufficiently large sample sizes, or, we say

- `Output 3: the probability that` $\hat{x}$`'s objective function value is suboptimal by more than 1% is approximately 0.05.`

Quantifying what constitutes a "sufficiently large" sample size or quantifying the accuracy of the "approximate" probability in Output 3 is usually done through empirical tests. Such tests involve problems whose objective functions can be evaluated exactly and whose optimal values are known. It is not our goal here to perform such tests, but there has been significant work in this vein (Bayraksan and Morton [3], Freimer et al. [15], Janjarassuk and Linderoth [28], Keller and Bayraksan [29], Linderoth et al. [37], Mak et al. [39], Partani et al. [49], Santoso et al. [55], Verweij et al. [61]).

The above discussion is meant to frame what the reader can expect in terms of output from the procedures we describe. We will consider the following stochastic program

$$z^* = \min_{x \in X} \mathbb{E} f(x, \xi). \tag{SP}$$

Here, f is a real-valued function measuring the performance of the system of interest, x is a decision vector constrained to obey physical and policy rules represented by the set $X \subset \mathbb{R}^{d_x}$, ξ is a d_ξ-dimensional random vector, and $\mathbb{E}$ is the associated expectation operator. We denote an optimal solution and the optimal value of (SP) as x^* and z^*, respectively, recognizing that x^* may not be unique.

Example 1. Let f be defined as the optimal value of a linear program, given x and ξ, i.e.,

$$\begin{aligned} f(x, \xi) = cx + \min_{y \geq 0} \quad & gy \\ \text{s.t.} \quad & Dy = Bx + d. \end{aligned} \tag{1}$$

Here, ξ is a vector of random elements from d, g, B, and D. □

The class of models of type (SP) with f defined in (1) has been widely studied and is frequently employed in applications. A prototypical two-stage program of this nature designs a system via x under system-operating conditions known only through a probability distribution on ξ. Then, y represents an operational recourse decision that is made *after* those operating conditions become known, i.e., after ξ is realized. The system design, x, could involve continuous decisions that allocate capacity, discrete decisions that locate facilities or construct a network, or some mix.

Example 2. Let ξ be a random row vector of costs incurred by participating in activities via x, let c_o denote a cost threshold, and let $f(x, \xi) = \mathbb{I}(\xi x \geq c_o)$, where $\mathbb{I}(\cdot)$ is the indicator function that takes value one if its argument is true and is zero otherwise. □

Example 2 points to the fact that $\mathbb{E} f(x, \xi)$ can capture performance measures not usually thought of as a "mean." In this case, (SP) is $\min_{x \in X} \mathbb{P}(\xi x \geq c_o)$.

Example 3. Define $f(x, \xi)$ as in (1) except that $c = 0$ and the "min" is replaced by "max," although the "min" in (SP) remains. Let the linear program in (1) correspond to Player II selecting an origin-destination path via y in a transportation network that maximizes the probability he evades detection. Player I seeks to minimize that evasion probability

by installing sensors via x on the network, knowing only a probability distribution governing Player II's origin-destination pair. □

Stochastic programs can capture adversarial models under uncertainty, as indicated by Example 3. Here, Player II's origin-destination pair is governed by a known probability distribution, a typical assumption in stochastic programming, whereas his path is selected in a worst-case manner from Player I's perspective, as is typical in robust optimization.

Stochastic programs model many important systems that require decisions to be made under uncertainty, with applications emerging from a wide variety of areas including electric power systems, homeland security, finance, production supply chains, communications networks, transportation systems, and water resources management; see Wallace and Ziemba [62]. Example 2 is closely related to models with probabilistic constraints (Prékopa [52]). For more on Example 3, see Morton et al. [42] and Pan and Morton [47].

Unless ξ has a small number of realizations or f has a particularly simple structure, it is usually impossible to solve (SP) exactly. For problems in which ξ is of moderate to high dimension and is continuous or has a large number of realizations, Monte Carlo simulation is widely regarded as the method of choice for estimating $\mathbb{E}f(x,\xi)$, when x is fixed. So, one approach for approximately optimizing (SP) is to sample n independent and identically distributed (i.i.d.) observations $\xi^1,\xi^2,\ldots,\xi^n$ from the distribution of ξ and then solve the approximating problem

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

Of course, non-i.i.d. sampling schemes are also possible, as we will later discuss.

Solving (SP$_n$) in place of (SP) is justified by large sample size results that establish conditions on f, X, ξ, and the sampling procedure under which solutions (x_n^*, z_n^*) to (SP$_n$) are optimal to (SP) as the sample size n grows to infinity. See, for example, the survey by Shapiro [56]. These consistency results are clearly needed, but in our view, they are not enough because they tell us nothing about the quality of a solution, x_n^*, obtained by solving (SP$_n$) for finite n; i.e., they do not yield a statement like "Output 3" discussed above.

Imagine that we have successfully addressed what we have laid out so far but we are unsatisfied with Output 3, e.g., because instead of a 1%-near optimality statement, we have a 20%-near optimality statement. The excessive width of this approximate confidence interval (CI) could be because the candidate solution is not of sufficiently high quality, or it could be due to our assessment procedure. Although we will address both issues, assume for the moment the former is the cause. Then, we could solve another instance of (SP$_n$) under a larger sample size in attempt to improve the candidate solution. Or, perhaps better, we could instead employ a sampling-based algorithm to solve (SP) such as a stochastic approximation algorithm (Lan et al. [32]) or a sampling-based cutting-plane method (Higle and Sen [23], Infanger [27]). Such an algorithm can produce a sequence of iterates that has limit points that solve (SP), with probability one (w.p.1). Again, although such convergence results are clearly needed, they are insufficient when we must terminate finitely. Statistical lower bounds have been used to guide termination (Higle and Sen [19, 23], Infanger [27], Lan et al. [32]), but the sequential issues inherent in repeatedly testing termination criterion have received little attention.

So, in this tutorial we describe a simple sampling-based procedure for assessing the quality of a candidate solution to a stochastic program. The procedure is easy to implement and yields a statement like Output 3. After additional background and motivation in the next section, we describe our simple procedure for assessing solution quality in §3. Then, the three sections that follow present enhancements that can be employed if our initial procedure yields inadequate results for a given computational budget. Finally, §7 addresses the sequential issues that arise in an algorithmic setting.

2. Background

Our primary goal is to have efficient methods for assessing the quality of a feasible candidate solution $\hat{x} \in X$, or of a sequence of such candidate solutions. The candidate solutions may come from solving (SP_n), from an algorithm for solving (SP), or from applying a heuristic. The optimal value, z_n^*, to (SP_n) plays an important role in accomplishing our goal. So, before proceeding, we give a simple example that illustrates a number of properties of z_n^*.

Example 4. Define (SP) through $X = [-1, 1]$, $\xi \sim N(0, 1)$, and $f(x, \xi) = \xi x$. Clearly, $z^* = 0$, and every feasible solution is an optimal solution to this instance of (SP). Forming the approximating problem

$$z_n^* = \min_{-1 \le x \le 1} \left(\frac{1}{n} \sum_{j=1}^{n} \xi^j \right) x,$$

we find $x_n^* = 1$ if $\sum_{j=1}^{n} \xi^j < 0$, and $x_n^* = -1$ if $\sum_{j=1}^{n} \xi^j > 0$. The objective function coefficient is an average of n i.i.d. standard normals and so $z_n^* = -|N(0, 1/n)|$. In this example, z_n^* has the following properties:

1. $\mathbb{E}z_n^* \le z^*$, $\forall\, n$ — negative bias;
2. $\mathbb{E}z_n^* \le \mathbb{E}z_{n+1}^*$ — monotonically shrinking bias;
3. $z_n^* \to z^*$, w.p.1 — strong consistency;
4. $\sqrt{n}(z_n^* - z^*) = -|N(0, 1)|$ — nonnormal errors; and
5. $b(z_n^*) \equiv \mathbb{E}z_n^* - z^* = a_1/\sqrt{n}$ — $O(n^{-1/2})$ bias. □

The first two properties on z_n^* hold much more generally (Mak et al. [39]), as does the third, under mild conditions (e.g., Shapiro [56]). The fourth property is not in a form that holds for more general instances of (SP). Instead of "=" we usually have "⇒," i.e., convergence in distribution, and the limiting distribution will differ. That said, the result is representative of the more general case both with respect to the $n^{-1/2}$ rate of convergence and the "folded" normal random variables that arise. The fifth property concerns the rate at which z_n^*'s bias shrinks to zero. When (SP) has multiple optimal solutions, as in this example, $O(n^{-1/2})$ bias arises in a very general setting. This is important because problems having multiple optimal solutions, or having a "large" set of ϵ-optimal solutions, are pervasive in optimization. The canonical rate at which sampling error in Monte Carlo methods shrinks to zero is $O(n^{-1/2})$, as illustrated here in the fourth property, and the fifth property suggests that the rate at which bias shrinks to zero can be as slow as that of sampling error.

These results contrast sharply with those that arise when the optimization operator "$\min_{x \in X}$" is not present in (SP) and (SP_n). If $x \in X$ is fixed, then under very mild conditions we have an unbiased estimator that is strongly consistent with normally distributed errors. Optimization leads to biased estimators with nonnormal errors. Even consistency can be lost as seen by replacing $X = [-1, 1]$ with $X = \mathbb{R}$ in Example 4. Finally, that the optimization operator can yield an $O(n^{-1/2})$ bias result is also atypical relative to what one usually obtains when other "smoother" types of operators are applied to sample means. For example, suppose ϕ is a smooth nonlinear function and we seek to estimate $\phi(\mathbb{E}Y)$ by $\phi(\bar{Y}_n)$, where $\bar{Y}_n$ is a sample-mean estimator of the scalar population mean $\mathbb{E}Y$. Then $\phi(\bar{Y}_n) \approx \phi(\mathbb{E}Y) + \phi'(\mathbb{E}Y)(\bar{Y}_n - \mathbb{E}Y) + \frac{1}{2}\phi''(\mathbb{E}Y)(\bar{Y}_n - \mathbb{E}Y)^2$ implies $\mathbb{E}\phi(\bar{Y}_n) \approx \phi(\mathbb{E}Y) + \frac{1}{2}\phi''(\mathbb{E}Y)\operatorname{var}Y/n$; i.e., the estimator has bias that shrinks to zero with $O(n^{-1})$. Thus, the optimization operator qualitatively changes the nature of results arising when employing Monte Carlo sampling. This suggests that we may need to exercise care when developing point and interval estimators that are rooted in (SP_n).

A key observation is that z_n^* gives a lower bound, in expectation, on the optimal solution value z^* (Property 1 in Example 4). The intuition is as follows. In solving the original problem (SP), we seek a decision, x, that hedges against *all* realizations of ξ. When solving (SP_n), we optimize with respect to a subset of ξ's support, selected according to the distribution of ξ. Because of this "inside information," we overoptimize and, on average, obtain

an optimistic objective function value. Based on this same intuition, we expect the value of the bound to grow as n increases (Property 2 in Example 4).

As indicated above, the z_n^* bound is analogous to lower bounds that arise from optimizing relaxations in other areas of optimization, except that it is statistical in nature. Therefore, it will yield a different type of optimality statement compared to deterministic problems as discussed in §1. Now, suppose we take as a candidate solution the solution $\hat{x} = x_n^*$ of (SP_n) along with its optimal value z_n^*. The following question then arises: In what sense should we seek to characterize x_n^* and/or z_n^* as being "good" estimators of their population counterparts x^* and z^* from (SP)? We could pursue a number of different goals regarding such inference. For example, we could try to establish any of the following.

Goal 1. $x_n^* \to x^*$, w.p.1, and $\sqrt{n}(x_n^* - x^*) \Rightarrow Y_x$ (for some limiting random variable Y_x).
Goal 2. $z_n^* \to z^*$, w.p.1, and $\sqrt{n}(z_n^* - z^*) \Rightarrow Y_z$ (for some limiting random variable Y_z).
Goal 3. $\mathbb{E}f(x_n^*, \xi) \to z^*$, w.p.1.
Goal 4. $\lim_{n\to\infty} \mathbb{P}(\mathbb{E}f(x_n^*, \xi) \le z^* + \epsilon_n) = 1 - \alpha$, where the random width satisfies $\epsilon_n \to 0$, w.p.1.

Goal 1 would provide a route to forming an approximate CI on the optimal solution x^* using the limiting distribution Y_x. Similarly, we could attempt to use Goal 2 to form an approximate CI on z^*. As we have seen above, the limiting distributions Y_x and Y_z may be nonnormal. If (SP_n) were a maximum-likelihood estimation problem, then Goal 1 would be appropriate, and if (SP) were a model to price a financial option, then Goal 2 would be appropriate. However, when (SP) is a decision-making model, then we believe that Goal 1 is more than we need, and Goal 2 is of secondary interest. In this case, Goals 3 and 4 are arguably what we should try to achieve. Note that the expectation in Goal 3 is with respect to ξ given x_n^*; i.e., $\mathbb{E}f(x_n^*, \xi)$ is a random variable. The probability in Goal 4 is with respect to the random width ϵ_n and is also conditional on x_n^*. We further note that we cannot expect $\{x_n^*\}_{n=1}^{\infty}$ to converge when (SP) has multiple optimal solutions. In this case, we weaken the consistency result in Goal 1 to the following: every limit point of $\{x_n^*\}_{n=1}^{\infty}$ solve (SP). When X is compact and $\mathbb{E}f(x, \xi)$ is continuous, and if we achieve this "limit points result," then we obtain Goal 3. Finally, as noted via the references given above, these goals—particularly in the form of 1–3—have been pursued and established under various conditions on f, ξ, and X. In this tutorial, our focus is on results along the lines of Goal 4. In particular, with $\hat{x} = x_n^*$ fixed, and ϵ_n denoting the CI width on the optimality gap of $\hat{x}$, we will obtain results of the type in "Output 3."

Finally, with respect to background, we recall some basic notions associated with CIs (see, e.g., Casella and Berger [8, §9.1]). An *interval estimator*, I_n, of a real-valued parameter is a random set based on sample size n. For example, for finite n with $\hat{x} = x_n^*$, $I_n = [0, \epsilon_n]$ in Goal 4 is an interval estimator for the *optimality gap*, $\mu_{\hat{x}} = \mathbb{E}f(\hat{x}, \xi) - z^*$. The *coverage probability*, or simply the *coverage*, of an interval estimator is the probability that the random interval I_n contains the parameter of interest, e.g., $\mathbb{P}(\mu_{\hat{x}} \in I_n)$. Typically, asymptotic results such as that in Goal 4 provide theoretical justification for a specific procedure to form the interval estimator I_n, provided n is sufficiently large. As indicated in §1, practical interpretation of what is meant by sufficiently large is usually guided by empirical testing, specifically, by empirically assessing coverage probabilities. We are now ready to present our first point and interval estimator of a given candidate solution's optimality gap formed via Monte Carlo sampling.

3. Multiple Replications Procedure

In the spirit of Goal 4 of the previous section, we will measure the quality of a candidate solution $\hat{x}$, e.g., $\hat{x} = x_n^*$, by the optimality gap, $\mu_{\hat{x}} = \mathbb{E}f(\hat{x}, \xi) - z^*$. If the gap is sufficiently

small, then $\hat{x}$ is of high quality. An upper bound on the optimality gap for $\hat{x}$ is given by $\mathbb{E}f(\hat{x},\xi) - \mathbb{E}z_n^*$, because $\mathbb{E}z_n^* \le z^*$. We estimate this quantity by

$$G_n(\hat{x}) = \frac{1}{n}\sum_{j=1}^{n} f(\hat{x},\xi^j) - \min_{x\in X}\frac{1}{n}\sum_{j=1}^{n} f(x,\xi^j), \tag{2}$$

where $\xi^1,\xi^2,\ldots,\xi^n$ are i.i.d. from the distribution of ξ. The second term in (2) is simply z_n^* and, as indicated in the previous section, is not asymptotically normal, and so neither is $G_n(\hat{x})$. We can circumvent this issue via a multiple replications procedure to construct a CI of the form

$$\mathbb{P}(\mathbb{E}f(\hat{x},\xi) - z^* \le \epsilon) \approx 1-\alpha. \tag{3}$$

Here, $\hat{x}\in X$ is a candidate solution, $\mathbb{E}f(\hat{x},\xi)$ is its "true" and unknown expected performance measure, ϵ is the (random) CI width, and $1-\alpha$ is the confidence level, e.g., 0.95. Let $t_{n,\alpha}$ be the $1-\alpha$ quantile of the t distribution with n degrees of freedom, and, for later use, let z_α be that of the standard normal. We summarize below our multiple replications procedure (MRP) for constructing (3) from Mak et al. [39].

MRP:
Input: Value $\alpha\in(0,1)$ (e.g., $\alpha=0.05$), sample size n, replication size n_g, and a candidate solution $\hat{x}\in X$.
Output: Approximate $(1-\alpha)$-level confidence interval on $\mu_{\hat{x}}$.

1. For $k=1,2,\ldots,n_g$:
   1.1. Sample i.i.d. observations $\xi^{k1},\xi^{k2},\ldots,\xi^{kn}$ from the distribution of ξ.
   1.2. Solve (SP_n) using $\xi^{k1},\xi^{k2},\ldots,\xi^{kn}$ to obtain x_n^{k*}.
   1.3. Calculate $G_n^k(\hat{x}) = n^{-1}\sum_{j=1}^{n}(f(\hat{x},\xi^{kj}) - f(x_n^{k*},\xi^{kj}))$.
2. Calculate gap estimate and sample variance by

$$\bar{G}_n(n_g) = \frac{1}{n_g}\sum_{k=1}^{n_g} G_n^k(\hat{x}) \quad\text{and}\quad s_G^2(n_g) = \frac{1}{n_g-1}\sum_{k=1}^{n_g}(G_n^k(\hat{x}) - \bar{G}_n(n_g))^2.$$

3. Let $\epsilon_g = t_{n_g-1,\alpha}s_G(n_g)/\sqrt{n_g}$, and output the one-sided CI on $\mu_{\hat{x}}$,

$$[0, \bar{G}_n(n_g) + \epsilon_g].$$

We know $G_n(\hat{x})$ can be nonnormal, so we produce n_g i.i.d. replicates in Step 1 and form the resulting sample mean $\bar{G}_n(n_g)$ and sample variance $s_G^2(n_g)$ in Step 2. The CI $[0,\bar{G}_n(n_g)+\epsilon_g]$ on $\mu_{\hat{x}} = \mathbb{E}f(\hat{x},\xi) - z^*$ is inferred from the central limit theorem (CLT)

$$\sqrt{n_g}[\bar{G}_n(n_g) - \mathbb{E}G_n(\hat{x})] \;\Rightarrow\; N(0,\sigma_g^2) \quad \text{as } n_g\to\infty, \quad \text{where } \sigma_g^2 = \operatorname{var} G_n(\hat{x}),$$

and the fact that $s_G^2(n_g)$ is a consistent estimator of σ_g^2. The CLT holds by requiring that the batches $\xi^{k1},\xi^{k2},\ldots,\xi^{kn}$, $k=1,2,\ldots,n_g$, be i.i.d. We have more freedom with respect to the random vectors within a batch. Specifically, for fixed values of x, we only require that the random vectors within a batch produce unbiased estimators; i.e., they satisfy $\mathbb{E}n^{-1}\sum_{j=1}^{n} f(x,\xi^{kj}) = \mathbb{E}f(x,\xi)$, $k=1,2,\ldots,n_g$. This freedom is later exploited in §6. Note that the random CI width consists of the point estimate of the optimality gap, $\bar{G}_n(n_g)$, which is biased because of the bias of the lower bound, z_n^*, plus the sampling error, ϵ_g. The random CI width ϵ in (3) is simply $\epsilon = \bar{G}_n(n_g) + \epsilon_g$ in the interval estimator produced by the MRP.

Norkin et al. [46] used the statistical lower bound z_n^* in global optimization of stochastic programs within a branch-and-bound method. Other algorithmic work that uses Monte Carlo simulation-based bounds and multiple replications includes Ahmed and Shapiro [1]

and Kleywegt et al. [31]. MRP has been applied to different kinds of problems in the literature including financial portfolio models (Bertocchi et al. [5], Morton et al. [43]), stochastic vehicle routing problems (Kenyon and Morton [30], Verweij et al. [61]), supply chain network design (Santoso et al. [55]), a stochastic water resources model to contain groundwater contaminants (Watkins et al. [63]), an employee and production scheduling problem (Morton and Popova [40]), a stochastic integer knapsack problem (Morton and Wood [41]), and a stochastic network interdiction model (Janjarassuk and Linderoth [28]).

There is other related work on assessing solution quality in stochastic programs via Monte Carlo methods, some being in the context of specific algorithms. Higle and Sen [20] derive a bound on the optimality gap for two-stage stochastic linear programs that is motivated by the Karush-Kuhn-Tucker optimality conditions (see also Shapiro and Homem-de-Mello [57]). Higle and Sen [21] have also proposed a statistical lower bound that is rooted in duality. Dantzig and Glynn [11], Dantzig and Infanger [12], Infanger [26, 27], and Higle and Sen [19, 23] use Monte Carlo versions of lower bounds obtained in sampling-based adaptations of deterministic cutting-plane algorithms. Throughout we focus on problems in which the feasible region, X, is simple in that feasibility of a candidate solution is easy to enforce and verify. This contrasts with related work on assessing solution quality in problems with probabilistic constraints (Luedtke and Ahmed [38]).

The MRP has a number of important advantages. Foremost is its wide applicability. The same approach is valid regardless of whether (SP) is a two-stage stochastic linear or nonlinear program, X includes integer restrictions on x, the second-stage optimization model (1) includes integer restrictions on y, or $\mathbb{E}f(x,\xi)$ is a "nonstandard" objective function as in Example 2. Of course, the techniques employed to solve (SP_n) in Step 1.2 in these cases will vary, as will the computational effort required to do so. That said, the (asymptotic) validity of the CI on the optimality gap produced by the MRP is not an issue. To carry out the method we must be able to generate i.i.d. observations of ξ, and for the CLT to apply, we require that $f(x,\xi)$ has finite second moments. The MRP is not tied to any particular algorithm so that (SP_n) can be solved by any number of means, and the effort to implement the MRP can be modest. For example, the MRP can be implemented with relative ease in a modeling language such as GAMS (Brooke et al. [6]). We have also observed that the MRP tends to be conservative in its coverage properties (see, e.g., Bayraksan and Morton [3], Partani [48]).

For many problems, one can carry out the MRP and obtain satisfactory results, in the sense of Output 3 from the introduction, with modest computational effort. The remainder of this tutorial is devoted to what one can do when this is *not* the case. Below we list four reasons that we might not obtain satisfactory results when running the MRP:

(a) the computational effort to solve (say) $n_g = 30$ instances of (SP_n) is prohibitive;
(b) the bias of z_n^* is large;
(c) the sampling error, ϵ_g, is large; or
(d) the candidate solution $\hat{x}$ is far from optimal to (SP).

Factors (b)–(d) contribute to the CI width, and any one of them can result in a CI width that is too large to be useful. Of course, if (a) occurs, then we are unable to produce a CI with reasonable computational effort. This could occur because n_g is large or because solving a modest number of instances of (SP_n) is too expensive. However, we usually fix $n_g = 20$–30, and multiple processors can be used to lessen the burden of solving n_g problems. So, it's typically the latter cause. We now briefly preview the ways in which the remainder of this tutorial addresses (a)–(d).

(a) Single- and Two-Replication Procedures (§4). The reason for performing n_g replications in MRP is that z_n^* can have nonnormal errors. Despite this, in §4 we describe an approach that enables us to assess the solution quality of $\hat{x}$ by solving either one or

two instances of (SP_n). We may pursue this option when the output of the MRP would be acceptable but the computational cost of achieving that result is excessive.

(b) Bias Reduction (§5). In integer programming, sometimes we have an optimal, or near optimal, $\hat{x}$, but considerable computational effort is required to tighten weak lower bounds and prove optimality. Issue (b) is analogous in stochastic programming. As we have discussed, $\mathbb{E}z_n^* \leq z^*$, and sometimes this bias is the largest contributor to the CI width. In §5, we use an adaptive jackknife estimator designed to tighten weak lower bounds.

(c) Variance Reduction (§6). The error as a result of sampling, ϵ_g, can dominate the CI width and make it impossible to recognize a high-quality solution. Decreasing ϵ_g by increasing n_g or n can be prohibitively expensive because the error shrinks at rate $n_g^{-1/2}$ or $n^{-1/2}$. Section 6 employs a randomized quasi-Monte Carlo (QMC) procedure to reduce ϵ_g.

(d) Sequential Sampling (§7). To alleviate problem (d), if $\hat{x}$ is obtained via (SP_n), we can increase n. If we instead (asymptotically) solve (SP) via a sampling-based algorithm, we can obtain the algorithm's next iterate. Practical implementation requires *finite* stopping, i.e., termination after a finite number of iterations using a finite sample size, and iterative testing of a statistical termination criterion requires care. Section 7 presents rules to iteratively increase the sample size and rules to terminate that ensure a priori control on solution quality.

The four remedies we suggest are not tied to specific solution algorithms and hence have broad applicability. We emphasize that the techniques we propose to address (a), (b), and (c) are not designed to be used in concert. Rather, they are a set of tools that can be employed to improve the efficiency of the MRP depending on the nature of the computational bottleneck for the specific problem under consideration. (Although we will not explore the issue here, the assessment of which issue is of primary importance can usually be made via a computationally inexpensive pilot study with relatively small sample sizes.) Indeed, the corresponding sections are written in a modular manner so that the reader can skip directly to the recommended remedy. The sequential sampling development for (d) provides a sufficiently general framework that it could be coupled with any of the other techniques. Here, we only point to how it can be coupled with the single- and two-replication procedures of the next section.

4. Single- and Two-Replication Procedures

When applying the MRP, the replication size, n_g, is taken to be, say, 20 or 30 in an attempt to have a valid statistical inference. This section is based on Bayraksan and Morton [3] and begins with a single-replication procedure (SRP) to make a valid statistical inference on the quality of a candidate solution.

As before, let the candidate solution $\hat{x} \in X$ be given. We use the following additional notation. For a feasible solution, $x \in X$, let $\bar{f}_n(x) = n^{-1}\sum_{j=1}^{n} f(x,\xi^j)$, $\sigma^2_{\hat{x}}(x) = \mathrm{var}[f(\hat{x},\xi) - f(x,\xi)]$, and $s_n^2(x) = (n-1)^{-1}\sum_{j=1}^{n}[(f(\hat{x},\xi^j) - f(x,\xi^j)) - (\bar{f}_n(\hat{x}) - \bar{f}_n(x))]^2$. Note that $G_n(\hat{x})$ given in (2) can be written as $\bar{f}_n(\hat{x}) - z_n^*$ with the understanding that the same n observations $\xi^1, \xi^2, \ldots, \xi^n$ are used in $\bar{f}_n(\hat{x})$ and z_n^*. The single-replication procedure follows.

SRP:
Input: Value $\alpha \in (0,1)$, sample size n, and a candidate solution $\hat{x} \in X$.
Output: Approximate $(1-\alpha)$-level confidence interval on $\mu_{\hat{x}}$.

1. Sample i.i.d. observations $\xi^1, \xi^2, \ldots, \xi^n$ from the distribution of ξ.
2. Solve (SP_n) to obtain x_n^*.
3. Calculate $G_n(\hat{x})$ as given in (2) and

$$s_n^2(x_n^*) = \frac{1}{n-1}\sum_{j=1}^{n}\left[\left(f(\hat{x},\xi^j) - f(x_n^*,\xi^j)\right) - \left(\bar{f}_n(\hat{x}) - \bar{f}_n(x_n^*)\right)\right]^2.$$

4. Let $\epsilon_g = t_{n-1,\alpha} s_n(x_n^*)/\sqrt{n}$, and output the one-sided CI on $\mu_{\hat{x}}$,

$$[0, G_n(\hat{x}) + \epsilon_g].$$

The SRP and the MRP differ in how the sample variance is calculated. In the MRP, n_g i.i.d. observations of $G_n(\hat{x})$ are calculated and the sample variance of these gap estimates is used to form the CI. In contrast, only one value of $G_n(\hat{x})$ is calculated in the SRP and the individual observations, $f(\hat{x},\xi^j) - f(x_n^*,\xi^j)$ for $j = 1,\ldots,n$, are used to calculate the sample variance. In fact, $G_n(\hat{x})$ is the sample mean of these individual observations, and $s_n^2(x_n^*)$ is the corresponding sample variance. The following theorem provides asymptotic validity of the SRP.

Theorem 1. *Assume $X \neq \varnothing$ and is compact, $\mathbb{E}f(\cdot,\xi)$ is lower semicontinuous on X, and $\mathbb{E}\sup_{x\in X} f^2(x,\xi) < \infty$. Let $\hat{x} \in X$ and $\xi^1,\xi^2,\ldots,\xi^n$ be i.i.d. as ξ. Let $X^* = \operatorname{argmin}_{x\in X} \mathbb{E}f(x,\xi)$ and assume*

$$\inf_{x\in X^*} \sigma_{\hat{x}}^2(x) \leq \liminf_{n\to\infty} s_n^2(x_n^*) \quad \text{and} \quad \limsup_{n\to\infty} s_n^2(x_n^*) \leq \sup_{x\in X^*} \sigma_{\hat{x}}^2(x), \text{ w.p.1.} \tag{4}$$

Then, given $0 < \alpha < 1$ in the SRP,

$$\liminf_{n\to\infty} \mathbb{P}\left(\mu_{\hat{x}} \leq G_n(\hat{x}) + \frac{z_\alpha s_n(x_n^*)}{\sqrt{n}}\right) \geq 1 - \alpha.$$

Theorem 1 justifies construction of the approximate $(1-\alpha)$-level one-sided CI for $\mu_{\hat{x}} = \mathbb{E}f(\hat{x},\xi) - z^*$, given in the SRP without requiring $G_n(\hat{x}) = \bar{f}_n(\hat{x}) - z_n^*$ to be asymptotically normal. Sufficient conditions to ensure (4) and Theorem 1's proof are provided in Bayraksan and Morton [3].

The MRP is a procedure in which we use $n_g \geq 20$–30 replications and the SRP is a procedure with just one replication, $n_g = 1$. As we show in Bayraksan and Morton [3], empirical coverage results for the SRP do not always inherit the relatively conservative nature of those of the MRP. For this reason, we provide the following two-replication procedure (2RP) variant of the SRP.

2RP:

Recall the MRP and fix $n_g = 2$. Replace Steps 1.3, 2, and 3 by the following:

1.3′. Calculate $G_n^k(\hat{x})$ and $s_n^2(x_n^{k*})$.

2′. Calculate the estimates by taking the averages,

$$G_n'(\hat{x}) = \tfrac{1}{2}\big(G_n^1(\hat{x}) + G_n^2(\hat{x})\big) \quad \text{and} \quad s_n^{2\prime} = \tfrac{1}{2}\big(s_n^2(x_n^{1*}) + s_n^2(x_n^{2*})\big).$$

3′. Output the one-sided CI on $\mu_{\hat{x}}$,

$$\left[0, G_n'(\hat{x}) + \frac{t_{2n-1,\alpha} s_n'}{\sqrt{2n}}\right].$$

The sample variance, $s_n^2(x_n^{k*})$, for each sample $k = 1, 2$, in the 2RP is calculated as in the single-replication procedure, and these are averaged to obtain the variance estimator of the 2RP (in Step 2′). We provide in Bayraksan and Morton [3] an asymptotic justification for the 2RP that is analogous to Theorem 1. When concern for conservative coverage results is paramount but the MRP is too computationally expensive to employ, we recommend the 2RP. For some problems, the cost of performing multiple replications, i.e., Issue (a), is the primary computational bottleneck in efficiently assessing solution quality. In such cases, the 2RP can yield considerable computational savings.

5. Bias Reduction

When lower bounds are weak, we may fail to recognize an optimal, or near-optimal, candidate solution. In other words, the bias of z_n^* can significantly degrade our ability to assess the quality of a candidate solution. In this section, we present techniques to reduce this bias, and our approach is rooted in a jackknife estimator. First used by Quenouille [53, 54], jackknife methods have become an important tool in simulation and data analysis. The standard jackknife estimator eliminates $O(n^{-1})$ bias. The generalized jackknife estimator, as developed in Gray and Schucany [17], instead eliminates $O(n^{-p})$ bias for p that may differ from unity. We have already seen in Example 4 that we can have $b(z_n^*) = \mathbb{E}z_n^* - z^* = O(n^{-1/2})$. The following example shows that we can have $b(z_n^*) = O(n^{-p})$ for any $p \in [1/2, \infty)$.

Example 5. Define (SP) through $X = \mathbb{R}$, $\xi \sim N(0,1)$, and $f(x,\xi) = \xi x + |x|^\beta$, where $\beta > 1$. Clearly, $z^* = 0$ and $x^* = 0$. The approximating problem

$$z_n^* = \min_x \left(\frac{1}{n} \sum_{j=1}^{n} \xi^j \right) x + |x|^\beta$$

has optimal value

$$z_n^* = -\beta^{-\beta/(\beta-1)}(\beta-1)|N(0,1)|n^{-p},$$

where $p = \beta/(2(\beta-1))$. Taking expectations, we obtain $b(z_n^*) = \mathbb{E}z_n^* = -an^{-p}$, where $a > 0$ is a constant independent of n. As $\beta \to \infty$, $p \to 1/2$, and as $\beta \to 1$, $p \to \infty$. □

Let $\hat{\theta}_n$ be an estimator of θ that is based on n underlying observations. Here, $\hat{\theta}_n$ could be z_n^* or the gap estimator $\bar{G}_n(n_g)$. Example 5 compels us to consider the case when $\mathbb{E}\hat{\theta}_n - \theta = O(n^{-p})$ for values of p in the range $p \in [1/2, \infty)$. So, following the ideas of Gray and Schucany [17], assume

$$\mathbb{E}\hat{\theta}_{n/2} = \theta + a(n/2)^{-p}, \tag{5a}$$

$$\mathbb{E}\hat{\theta}_n = \theta + an^{-p}, \tag{5b}$$

where n is even. We view (5) as a system of two equations in two unknowns, θ and a. Eliminating the latter to solve for the former,

$$\theta = \frac{n^p \mathbb{E}\hat{\theta}_n - (n/2)^p \mathbb{E}\hat{\theta}_{n/2}}{n^p - (n/2)^p}. \tag{6}$$

This suggests an estimator in which $\hat{\theta}_n$ and $\hat{\theta}_{n/2}$ respectively replace $\mathbb{E}\hat{\theta}_n$ and $\mathbb{E}\hat{\theta}_{n/2}$ in (6).

Unfortunately, for a stochastic program, we are unlikely to know the order of the bias, p. One way around this difficulty is to use a two-stage procedure in which we first estimate p, and then use that point estimate in the estimator of θ sketched above. This idea is pursued in Partani et al. [49], but here we instead form

$$\mathbb{E}\hat{\theta}_{n/4} = \theta + a(n/4)^{-p}, \tag{7a}$$

$$\mathbb{E}\hat{\theta}_{n/2} = \theta + a(n/2)^{-p}, \tag{7b}$$

$$\mathbb{E}\hat{\theta}_n = \theta + an^{-p}, \tag{7c}$$

where n is a multiple of 4. We view (7) as a system of three equations in three unknowns: θ, a, and p. We could mimic the above ideas to derive an estimator of θ, but as discussed in Partani [48], the resulting estimator can fail to satisfy even the most basic property like consistency.

We instead form an estimator by first using (7a) and (7b) to obtain $an^{-p} = (\mathbb{E}\hat{\theta}_{n/2} - \theta)^2 / (\mathbb{E}\hat{\theta}_{n/4} - \theta)$. Of course, θ is unknown, but $\theta \approx \mathbb{E}\hat{\theta}_n$ yields

$$an^{-p} \approx \frac{(\mathbb{E}\hat{\theta}_{n/2} - \mathbb{E}\hat{\theta}_n)^2}{\mathbb{E}\hat{\theta}_{n/4} - \mathbb{E}\hat{\theta}_n},$$

which, in view of (7c), approximates $\mathbb{E}\hat{\theta}_n$'s bias. That is,

$$\theta \approx g(\mathbb{E}\hat{\theta}_{n/4}, \mathbb{E}\hat{\theta}_{n/2}, \mathbb{E}\hat{\theta}_n), \tag{8}$$

where

$$g(\theta_1, \theta_2, \theta_3) = \theta_3 - \left(\frac{\theta_2 - \theta_3}{\theta_1 - \theta_3}\right)(\theta_2 - \theta_3).$$

An adaptive jackknife estimator arises by replacing $(\mathbb{E}\hat{\theta}_{n/4}, \mathbb{E}\hat{\theta}_{n/2}, \mathbb{E}\hat{\theta}_n)$ on the right-hand side of (8) with a triple of estimators with corresponding sample sizes. We call this an adaptive estimator because, unlike the generalized jackknife, which requires the order p be prespecified, our approach estimates the rate at which the bias shrinks to zero.

With γ denoting a positive integer, Partani [48] extends this to a family of adaptive jackknife estimators using

$$g_\gamma(\theta_1, \theta_2, \theta_3) = \theta_3 - \sum_{i=1}^{\gamma} \left(\frac{\theta_2 - \theta_3}{\theta_1 - \theta_3}\right)^i (\theta_2 - \theta_3).$$

As γ grows, the family of estimators becomes more aggressive in reducing bias. Here, aggressiveness must be tempered by our aversion to overcorrecting and destroying the validity of the lower-bound, and hence gap, estimators. In this sense, estimators based on $g_\gamma(\cdot)$ do not appear excessively aggressive, even for large γ (Partani [48]). In describing the adaptive multiple replication procedure (MRP^A) below, we let $N = \{1, 2, \ldots, n\}$ index an i.i.d. sample $\xi^1, \xi^2, \ldots, \xi^n$, and we let $\hat{f}(x, N) = |N|^{-1} \sum_{j \in N} f(x, \xi^j)$.

MRP^A:

Input: Value $\alpha \in (0,1)$, sample size n (multiple of 4), replication size n_g, adaptive jackknife parameter γ, and a candidate solution $\hat{x} \in X$.

Output: Approximate $(1-\alpha)$-level confidence interval on $\mu_{\hat{x}}$.

1. For $k = 1, 2, \ldots, n_g$,
   1.1. Sample i.i.d. observations, indexed by N, $\xi^{k1}, \xi^{k2}, \ldots, \xi^{kn}$ from the distribution of ξ.
   1.2. Let N^j, $j = 1, \ldots, 4$, randomly partition N with $|N^j| = n/4$, $j = 1, \ldots, 4$.
   1.3. Let $\hat{\theta}_n^k = \hat{f}(\hat{x}, N) - \min_{x \in X} \hat{f}(x, N)$.
   1.4. Let

$$\hat{\theta}_{n/2}^{k1} = \hat{f}(\hat{x}, N^1 \cup N^2) - \min_{x \in X} \hat{f}(x, N^1 \cup N^2) \quad \text{and}$$
$$\hat{\theta}_{n/2}^{k2} = \hat{f}(\hat{x}, N^3 \cup N^4) - \min_{x \in X} \hat{f}(x, N^3 \cup N^4).$$

   1.5. Let $\hat{\theta}_{n/4}^{kj} = \hat{f}(\hat{x}, N^j) - \min_{x \in X} \hat{f}(x, N^j)$, $j = 1, \ldots, 4$.
   1.6. Let $\hat{\theta}_{n/2}^k = \frac{1}{2} \sum_{j=1}^{2} \hat{\theta}_{n/2}^{kj}$ and $\hat{\theta}_{n/4}^k = \frac{1}{4} \sum_{j=1}^{4} \hat{\theta}_{n/4}^{kj}$.
2. Form

$$\vec{\theta} = \left(\frac{1}{n_g} \sum_{k=1}^{n_g} \hat{\theta}_{n/4}^k, \frac{1}{n_g} \sum_{k=1}^{n_g} \hat{\theta}_{n/2}^k, \frac{1}{n_g} \sum_{k=1}^{n_g} \hat{\theta}_n^k\right).$$

3. Let $\hat{C}$ denote the sample covariance matrix of $(\hat{\theta}_{n/4}^k, \hat{\theta}_{n/2}^k, \hat{\theta}_n^k)$, $k = 1, \ldots, n_g$.

4. Form $J_\gamma^A = g_\gamma(\vec{\theta})$ and $s^2 = \nabla^\top g_\gamma(\vec{\theta})\ \hat{C}\ \nabla g_\gamma(\vec{\theta})$.
5. Let $\epsilon_g = t_{n_g-1,\alpha} s/\sqrt{n_g}$, and output the one-sided CI on $\mu_{\hat{x}}$,

$$[0, J_\gamma^A + \epsilon_g].$$

The optimization operator "$\min_{x\in X}$," coupled with the fact that Steps 1.3–1.5 use the same samples indexed by N, ensures

$$\hat{\theta}_n^k \le \hat{\theta}_{n/2}^k \le \hat{\theta}_{n/4}^k, \quad \text{w.p.1, } k = 1, \ldots, n_g. \tag{9}$$

The following result establishes a consistency property for J_γ^A. Although this is defined for the adaptive estimator specified in MRP^A, Theorem 2 holds for a more general class of estimators as long as assumption (9) holds; see Partani [48] for the theorem's proof.

Theorem 2. *Assume that $\hat{\theta}_n$ is a strongly consistent estimator of θ. Let γ be a positive integer, and let J_γ^A be defined as in MRP^A. If (9) holds, then*

$$\lim_{n\to\infty} J_\gamma^A = \theta, \text{ w.p.1.}$$

We note that when we form J_γ^A by replacing $(\mathbb{E}\hat{\theta}_{n/4}, \mathbb{E}\hat{\theta}_{n/2}, \mathbb{E}\hat{\theta}_n)$ in $g_\gamma(\cdot)$ by the estimators formed in Steps 1 and 2, we are forming a nonlinear function of a vector-valued sample mean. The delta theorem is therefore used to form the CI in Steps 4 and 5. Moreover, J_γ^A is not an unbiased estimator of $g_\gamma(\mathbb{E}\hat{\theta}_{n/4}, \mathbb{E}\hat{\theta}_{n/2}, \mathbb{E}\hat{\theta}_n)$, but we can employ a second-order Taylor correction (Partani [48]); i.e., we can redefine J_γ^A in Step 4 via

$$J_\gamma^A = g_\gamma(\vec{\theta}) - \frac{1}{2n_g} \sum_{i=1}^{3}\sum_{j=1}^{3} \frac{\partial^2 g_\gamma}{\partial\theta_i \partial\theta_j}(\vec{\theta})\hat{C}_{ij}.$$

We also note that in Step 1.4 we form two problems with $n/2$ samples using $N^1 \cup N^2$ and $N^3 \cup N^4$, but there is benefit to instead forming $\binom{4}{2} = 6$ such problems with $N^1, \ldots, N^4$.

The MRP^A requires additional computational effort over the MRP of §3. Specifically, the MRP requires we solve n_g (30, say) instances of (SP$_n$), whereas MRP^A requires we solve n_g instances of (SP$_n$), $2n_g$ instances of (SP$_{n/2}$), and $4n_g$ instances of (SP$_{n/4}$). Still, when bias limits our ability to construct a sufficiently tight CI, computational work in Partani [48] indicates the additional effort can be worthwhile.

6. Variance Reduction

Uniform random variables (or pseudorandom variates) on the unit interval form the basis for numerical procedures for generating i.i.d. observations of a nonuniform random variable ξ. Using an appropriate transformation ϕ we can express $f(\xi) = f(\phi(U)) \equiv h(U)$, where U is a uniform random vector on the unit cube, $[0,1]^d$, and where, for simplicity, we temporarily suppress x. Here, we would have $d = d_\xi$ if, for example, the components of ξ are independent and each is generated via inversion. So, computing $\mathbb{E}f(\xi)$ may be viewed as numerical integration of h over $[0,1]^d$. In their simplest form, numerical quadrature rules for computing $\mathbb{E}h(U)$ (e.g., d-dimensional variants of Simpson's rule) use a *uniform grid* of points over $[0,1]^d$. However, the computational effort needed to achieve a specific level of error with quadrature grows exponentially in the dimension d, and in the absence of special structures such techniques are not usually viable for d larger than 3 or 4. MC methods instead draw i.i.d. observations of U over $[0,1]^d$. Each observation is generated independently, and so the resulting collection of n points is not as "evenly spread" across $[0,1]^d$ as it could be. Loosely speaking, QMC methods (e.g., Niederreiter [45]) lie somewhere between these two approaches. To avoid the dimensional effect of quadrature, QMC methods do not use a uniform grid but spread their points more evenly than MC methods to improve on the $O(n^{-1/2})$ convergence rate of MC methods, and hence improve their efficiency.

Application of QMC methods in stochastic programming is examined in Diwekar and Kalagnanam [13] and Pennanen and Koivu [50]. Homem-de-Mello [25] provides conditions under which pointwise (i.e., without the "$\min_{x\in X}$" operator) rates of convergence associated with non-i.i.d. sampling schemes like QMC and Latin hypercube sampling (LHS) are inherited by the solution and value estimators from (SP_n). LHS is also applied in Bailey et al. [2], Diwekar and Kalagnanam [13], Freimer et al. [15], and Linderoth et al. [37]. A number of other variance reduction techniques are also applied in stochastic programming. Importance sampling has been employed in Dantzig and Glynn [11] and Infanger [26]. Antithetic variates, control variates, importance sampling, and stratified sampling are explored in Higle [18]. As described in Mak et al. [39], the use of common random numbers in our MRP of §3 reduces variance, and this technique is also used in Higle and Sen [22]. In this section, we discuss the use of randomized QMC (RQMC) methods for reducing variability in MRP estimators.

QMC methods use *low-discrepancy* sequences of points that can achieve an $O(n^{-1}\log^d n)$ rate of convergence. For sufficiently large n, $n^{-1}\log^d n \le n^{-1/2}$. However, when $d = 14$ (which is small for a stochastic program) this requires $n \ge 8.3 \times 10^{59}$. Based on this worst-case bound, we have little hope for QMC methods working well on moderate- to high-dimensional problems. Yet there is considerable computational evidence in the literature that QMC methods can perform quite well relative to this bound and, more importantly, relative to MC methods. Work on the notion of an *effective dimension* of the integrand has been put forward to explain this behavior (Caflisch et al. [7]). Methods for constructing low-discrepancy sequences include (t, m, s)-nets; the sequences of Faure, Halton, Hammersley, Niederreiter, and others (e.g., Niederreiter [45]); and lattices (L'Ecuyer and Lemieux [36], Sloan and Joe [59]). These schemes produce infinite (deterministic) sequences of vectors in $[0,1]^d$ that have partial sequences with low discrepancy.

One shortcoming of QMC methods is the difficulty associated with obtaining good error statements. (In contrast, MC methods infer such error estimates from CLTs.) Another difficulty with employing QMC methods, particularly in our setting, is that the point estimates are not unbiased estimators—they are not even random variables. These two issues make it difficult to simply "plug in" a QMC scheme in our MRP in a sensible way.

Randomized QMC methods (e.g., Fox [14]) circumvent both of these difficulties. Let the n points generated via a QMC scheme be denoted $\mathcal{U}_n = \{u^1, u^2, \ldots, u^n\}$. RQMC sampling performs a random shift of these points. In particular, let v be a *random* vector from the uniform distribution on $[0,1]^d$. We redefine each element of $\mathcal{U}_n$ as $\tilde{u}^j = [u^j + v]$, where $[\cdot]$ returns the fractional part of its argument, and then we form $\tilde{h} = n^{-1}\sum_{j=1}^{n} h(\tilde{u}^j)$. Performing n_g i.i.d. shifts of $\mathcal{U}_n$ in this way yields n_g observations $\tilde{h}^k$, $k = 1, \ldots, n_g$, and we define $\bar{h}_{n_g} = n_g^{-1}\sum_{k=1}^{n_g} \tilde{h}^k$. Each $\tilde{u}^i$ is uniform on $[0,1]^d$ and the n_g random shifts are done independently so, $\tilde{h}^k$, $k = 1, \ldots, n_g$, are i.i.d., $\bar{h}(n_g)$ is an unbiased estimator of $\mathbb{E}h(U)$, the standard sample variance is an unbiased and strongly consistent estimator of $\mathrm{var}(\tilde{h}^1)$, and the standard CLT for i.i.d. random variables applies (Fox [14], L'Ecuyer and Lemieux [36], Sloan and Joe [59]). As an immediate consequence, we obtain the following theorem.

Theorem 3. *Consider the variant of the MRP from §3 in which we use RQMC sampling to generate the i.i.d. batches in Step* 1.1. *Then,*

(i) $\mathbb{E}G_n(\hat{x}) \ge \mathbb{E}f(\hat{x}, \xi) - z^*$,
(ii) $\sqrt{n_g}[\bar{G}_n(n_g) - \mathbb{E}G_n(\hat{x})] \Rightarrow N(0, \sigma_g^2)$ *as* $n_g \to \infty$, *where* $\sigma_g^2 = \mathrm{var}\, G_n(\hat{x})$, *and*
(iii) $\mathbb{E}s_G^2(n_g) = \sigma_g^2$ *and* $s_G^2(n_g) \to \sigma_g^2$, *w.p.*1.

From Theorem 3 we can infer that the output of an RQMC variant of the MRP satisfies (asymptotically) the CI (3) on the optimality gap. We observed significant decreases in sampling error, ϵ_g, by using the RQMC variant of the MRP. Moreover, although our initial motivation for exploring RQMC sampling was to reduce variance, we found empirically that

bias is also reduced relative to MC sampling. It does follow from the $O(n^{-1}\log^d n)$ rate of convergence that the bias shrinks more quickly than the worst-case $O(n^{-1/2})$ for MC, but the caveats on the size of n still hold. The intuitive explanation of the nature of the bias from §2 (i.e., having to hedge against only n realizations and not all of ξ's support) suggests that it would be more difficult to "overoptimize" n evenly spread scenarios than n random scenarios. Perhaps for similar reasons, bias reduction results are obtained in Freimer et al. [15] for LHS.

7. Sequential Sampling

The CI procedures presented in §§3–6 are static; that is, the candidate solution, $\hat{x} \in X$, and the sample size, n, are fixed input to those procedures. In this section, we allow both $\hat{x}$ and n to be adaptive; i.e., we use ideas developed so far in an algorithmic fashion. Consider the following basic framework for sequential sampling.

Step 1. Generate a candidate solution.
Step 2. Assess the quality of the candidate solution.
Step 3. Check stopping criterion. If satisfied, stop. Else, go to Step 1.

Note that Steps 1–3 closely resemble a typical optimization algorithm. Deterministic optimization algorithms generate candidate solutions, test whether the current solution passes an optimality test (e.g., whether a first-order necessary condition is satisfied), and if not, the algorithm seeks to improve the solution each time Step 1 is carried out. In the context of stochastic programming with Monte Carlo sampling-based approximations, improving the solution typically involves generating one or more additional samples of ξ. Because of the iterative testing of the statistical stopping criterion in Step 3, this is where stochastic optimization algorithms need additional care. For example, when the above CI procedures are used in an algorithmic setting, the sequence of candidate solutions and the sample size when the procedure stops become random variables. Their analysis differs from their static counterparts; see, for instance, similar sequential analyses in statistics (Chow and Robbins [10], Nadas [44]) and in simulation of stochastic systems (Glynn and Whitt [16], Law and Kelton [34], Law et al. [35]).

In Step 1 of the algorithm, a candidate solution can be generated by a number of methods that solve (approximations of) (SP). Therefore, the sequential procedure can work in conjunction with a range of algorithms. We require that this method generates solutions with at least one limit point that solves (SP), w.p.1. This includes, but is not limited to, solving a sequence of sampling problems (SP_{m_k}) with increasing sample sizes, $m_k \to \infty$ as $k \to \infty$, and setting the solutions $x^*_{m_k}$ as the candidate solution, i.e., $\hat{x}_k = x^*_{m_k}$, at iteration k. Alternatively, a sampling-based algorithm like stochastic decomposition (Higle and Sen [22]) can be used. In Step 2 of sequential sampling, we assess the quality of the current candidate solution. We allow the use of a variety of optimality gap estimators like the aforementioned ones in §§3–6, and ones developed in conjunction with the specific algorithm used in Step 1, provided certain conditions are met. Below, we formalize these assumptions.

Let X^* be the set of optimal solutions to (SP). For any $x \in X$, in addition to its optimality gap, μ_x, we define a variance term, $\sigma^2(x) = \text{var}[f(x,\xi) - f(x^*_{\min},\xi)]$, where $x^*_{\min} \in \arg\min_{y\in X^*} \text{var}[f(x,\xi) - f(y,\xi)]$. Let $\xi^1, \xi^2, \ldots, \xi^n$ be a sample of size n. Suppose we have at hand $G_n(x)$ an optimality gap point estimate that uses this sample of size n to estimate μ_x, and similarly suppose we have an estimator $s_n^2(x) \geq 0$ of the associated variance $\sigma^2(x)$. We define

$$D_n(x) = \frac{1}{n}\sum_{i=1}^{n}[f(x,\xi^i) - f(x^*_{\min},\xi^i)]. \tag{10}$$

Note that the expected value of $D_n(x)$ is μ_x, and its variance under independent sampling is $\sigma^2(x)/n$. We assume the estimators $D_n(x)$, $G_n(x)$, and $s_n^2(x)$ all use the same n observations.

Our assumptions on the method to generate candidate solutions at Step 1 and on the estimators follow:

Assumption 1 (A1). *The sequence of feasible candidate solutions* $\{\hat{x}_k\}$ *has at least one limit point in* X^*, *w.p.*1.

Assumption 2 (A2). *Let* $\{x_k\}$ *be a feasible sequence (i.e.,* $x_k \in X$*) with* x *as one if its limit points. Let sample size* n_k *satisfy* $n_k \to \infty$ *as* $k \to \infty$. *Then,* $\liminf_{k\to\infty} \mathbb{P}(|G_{n_k}(x_k) - \mu_x| > \delta) = 0$ *for any* $\delta > 0$.

Assumption 3 (A3). $G_n(x) \geq D_n(x)$, *w.p.*1, *for all* $x \in X$ *and* $n \geq 1$.

Assumption 4 (A4). $\liminf_{n\to\infty} s_n^2(x) \geq \sigma^2(x)$, *w.p.*1, *for all* $x \in X$.

Assumption 5 (A5). $\sqrt{n}(D_n(x) - \mu_x) \Rightarrow N(0, \sigma^2(x))$ *as* $n \to \infty$ *for all* $x \in X$, *where* $N(0, \sigma^2(x))$ *is a normal random variable with mean zero and variance* $\sigma^2(x)$.

Briefly, the assumptions are as follows: (i) the algorithm used in Step 1 eventually generates at least one optimal solution (A1), (ii) the statistical estimators of the optimality gap and its variance have desired properties such as convergence (A2)–(A4), and (iii) the sampling is done in such a way that a form of CLT holds, e.g., i.i.d. sampling, antithetic sampling, bootstrapping, etc. (A5). Assumption (A2) ensures convergence of the optimality gap estimate $G_n(x)$ in probability to the true optimality gap, μ_x, uniformly in X. Similarly, (A4) is a form of convergence property for the sampling variance, $s_n^2(x)$. Note that when (SP) has multiple optimal solutions, we cannot expect $\{x_n^*\}$ to have a single limit point, and hence we cannot expect $s_n^2(x)$ to converge as $n \to \infty$. However, (A4) is a form of convergence for $s_n^2(x)$ in the sense that it is bounded below by a minimum variance of $\sigma^2(x)$. Assumption (A3) ensures the correct direction of the bias in estimation of the optimality gap.

As an example, the SRP and 2RP described in §4 use estimators that satisfy the above assumptions for a class of stochastic programs. We note that other estimators do exist and new ones specifically designed for the algorithm used in Step 1 can be developed that satisfy the above conditions.

At iteration k of sequential sampling, we are given a candidate solution $\hat{x}_k \in X$ (from Step 1). We select a sample size, n_k, and estimate $\hat{x}_k$'s optimality gap through $G_{n_k}(\hat{x}_k)$ and $s_{n_k}(\hat{x}_k)$ (Step 2). To simplify notation, from now on we suppress dependence on $\hat{x}_k$ and n_k, and simply denote $\mu_k = \mu_{\hat{x}_k}$, $G_k = G_{n_k}(\hat{x}_k)$, and $s_k = s_{n_k}(\hat{x}_k)$. Let $h' > 0$ and $\epsilon' > 0$ be two scalars. In Step 3, we check the following stopping criterion and terminate sequential sampling at iteration

$$T = \inf_{k \geq 1} \{k\colon G_k \leq h' s_k + \epsilon'\}; \tag{11}$$

i.e., we stop the first time G_k's width relative to s_k falls below $h' > 0$ plus a small positive number ϵ'. Let $h > h'$. We select the sample size at iteration k according to

$$n_k \geq \left(\frac{1}{h - h'}\right)^2 \left(c_q + 2q \ln^2 k\right), \tag{12}$$

where $c_q = \max\{2\ln(\sum_{k=1}^{\infty} k^{-q \ln k}/\sqrt{2\pi}\alpha), 1\}$. Here, $q > 0$ is a parameter we can choose, which affects the number of samples we generate. Later we will discuss how to choose this parameter to minimize computational effort.

When we stop at iteration T according to (11), the sequential sampling procedure provides an approximate solution, $\hat{x}_T$, and a CI on its optimality gap, μ_T, as

$$[0, hs_T + \epsilon], \tag{13}$$

where $\epsilon > \epsilon'$. Both epsilon terms are small, e.g., around 10^{-6}. It is important to note that the quality statement (13) provides a larger bound ($hs_T + \epsilon$) on μ_T than the bound used as

a stopping criterion ($h's_T + \epsilon'$) for G_T in (11). Such inflation of the CI statement, relative to the stopping criterion, is fairly standard when using sampling methods with a sequential nature (Chow and Robbins [10], Glynn and Whitt [16]). A second remark is that the stopping rule (11) and the corresponding quality statement (13) are written *relative* to the standard deviation. This allows us to stop with a larger optimality gap estimate when the variability of the problem is large compared to a tighter stopping rule when the variability is low. Consequently, the quality statement regarding $\hat{x}_T$ is tighter when variability is low and larger when variability is high.

When the sample sizes are increased according to (12) and the procedure stops at iteration T according to (11), the CI (13) is an asymptotically valid CI provided $D_n(x)$ satisfies a finite moment generating function (MGF) assumption. Next, we state the MGF assumption, and then we summarize this result along with the fact that sequential sampling stops in a finite number of iterations, w.p.1.

Assumption 6 (A6). *For some* $\gamma_0 > 0$,

$$\sup_{n\geq 1} \sup_{x\in X} \mathbb{E} \exp\left[\gamma \frac{D_n(x) - \mu_x}{\sigma(x)/\sqrt{n}}\right] < \infty,$$

for all $|\gamma| \leq \gamma_0$.

Assumption (A6) holds, for instance, when X is compact, the distribution of ξ has bounded support, and $\xi^1, \xi^2, \ldots, \xi^n$ are generated i.i.d. from the distribution of ξ. For more general conditions under which (A6) holds, see Bayraksan and Morton [4]. The theorem below summarizes properties of the sequential procedure.

Theorem 4. *Let* $\epsilon > \epsilon' > 0$ *and* $h > h' > 0$ *and* $0 < \alpha < 1$. *Consider the above sequential sampling procedure where the sample size is increased according to* (12)*, and the procedure stops at iteration* T *according to* (11).

(i) *Assume* (A1) *and* (A2) *are satisfied. Then,* $\mathbb{P}(T < \infty) = 1$.
(ii) *Assume* (A3)–(A6) *are satisfied. Then,*

$$\liminf_{h\downarrow h'} \mathbb{P}(\mu_T \leq h s_T + \epsilon) \geq 1 - \alpha. \tag{14}$$

Part (i) of Theorem 4 implies that if the algorithm used in Step 1 eventually produces an optimal solution (A1) and the optimality gap estimate can consistently estimate solution quality (A2), then our sequential sampling procedure stops in a finite number of iterations, w.p.1. Part (ii) of Theorem 4 shows that for values of h close enough to h', or when the sample sizes n_k are large enough, we have the optimality gap of the solution when we stop within $[0, hs_T + \epsilon]$ with at least the desired probability of $1 - \alpha$.

Theorem 4 can be recovered when the MGF assumption is relaxed, at the price of a faster rate of growth in the sample size. When we assume that the MGF exists through (A6), the sample size growth is sublinear with respect to the iteration number k, of order $O(\log^2 k)$ in (12). When only the first two moments are finite, the growth in sample size is essentially linear, $O(k)$.

In (12), intelligent choices of the parameter q can help minimize the number of required samples and hence minimize the computational burden. This is especially important in applications where obtaining a new sample is costly and/or solving optimization problems with larger sample sizes become computationally prohibitive. Note that if the procedure terminates in T iterations, it uses $n_1 + n_2 + \cdots + n_T$ samples for the stopping rules. It is possible to show that this sum is convex in $q > 0$, and it is possible minimize this function to obtain an optimal q^*. For details on this and selection of other parameters; see Bayraksan and Morton [4].

A number of researchers have studied conditions on sample sizes such that desired (asymptotic) properties are satisfied when a sampling-based approximation is used to solve a

stochastic program. Shapiro et al. [58] provide insight as to the sample size needed to find the optimal solution for a class of convex, piecewise linear stochastic programs with a unique, sharp optimum. Homem-de-Mello [24] studies rates at which the sample size must grow to ensure consistency of the objective function estimator. For stochastic nonlinear programs, Polak and Royset [51] propose a procedure that aims to minimize the computational effort required to reduce an initial optimality gap in the context of so-called diagonalization schemes.

We end this section by noting that an attractive feature of the sequential procedure is its flexibility in how the n_k observations, $\xi^1, \xi^2, \ldots, \xi^{n_k}$, can be generated. One option is to use a single stream of observations in which at each iteration we simply augment the existing set of observations with a few new additional samples (augmentation). Alternatively, the observations from the previous iteration can be discarded and we can generate an entirely new set of observations of increased size (resampling). Intermediate options also exist and are permitted by the above theory. Warm-starting techniques, along with conservative coverage results, favor augmentation, whereas prevention of being trapped in a bad sample path favors resampling. Resampling at a moderate frequency (e.g., every 25 iterations) seems to provide a balance between these considerations, minimizing total solution time while remaining computationally efficient with a relatively high coverage probability (Bayraksan and Morton [4]).

8. Conclusions and Future Work

Stochastic programs can be challenging to solve, and therefore, significant work in the literature is devoted to approximation schemes. One simple approach is to solve (SP_n) in place of (SP). Another approach is to use a sampling-based adaptation of a steepest-descent or cutting-plane algorithm. Conditions under which these approaches are asymptotically valid, i.e., as the number of samples grows large, have received considerable attention. Ways to assess whether a candidate solution obtained when these schemes are terminated finitely have received much less attention. Moreover, sequential issues that arise when repeatedly testing statistical termination criteria have received even less attention. These are the focus of our tutorial.

We have presented in §3 a simple sampling-based procedure called the MRP for forming a confidence interval on the optimality gap, $\mu_{\hat{x}} = \mathbb{E}f(\hat{x}, \xi) - z^*$, of a feasible candidate solution, $\hat{x}$. This approach is similar to the batch-means approach commonly used in simulation (Law [33]). The MRP solves n_g instances of (SP_n) to form n_g optimality gap estimators, which are then averaged to obtain a point estimator of $\mu_{\hat{x}}$. The central limit theorem is invoked to form an approximate confidence interval on the optimality gap. At first, solving (say) $n_g = 30$ instances of (SP_n) to assess the quality of a solution $\hat{x}$ obtained, e.g., by solving a single instance (SP_{n_x}), may seem excessive. However, for many problems we can obtain sufficiently tight confidence intervals using $n \ll n_x$ so that the computational effort required to assess $\hat{x}$'s quality is substantially less than the effort required to compute $\hat{x}$. Sections 4–6 of this tutorial are devoted to remedies that we recommend when this is not the case, i.e., when the computational effort required to form a sufficiently tight confidence interval on $\mu_{\hat{x}}$ is excessive because of bias or variance, or simply to the computational difficulty of solving the underlying approximating problem. Section 7 addresses how to use our methods to assess solution quality in an algorithmic fashion. This allows for adaptively determining the sample size, as well as iteratively improving the candidate solution, to obtain a high-quality solution with a desired probability.

Section 4 forms a confidence interval on $\mu_{\hat{x}}$ using a single replication, that is, by solving just one (SP_n) instead of n_g instances. In some "nonsmooth" cases, our experience is that this procedure can be risky with respect to coverage properties, and so we recommend a more conservative two-replication procedure.

When the bias of the lower-bound estimator z_n^* is large, the confidence interval will be loose. Thus, §5 provides an adaptive jackknife estimator to reduce bias within the context of

a multiple-replications procedure. The procedure differs from previous jackknife estimators with respect to its adaptivity; i.e., it estimates the rate, p, at which $O(n^{-p})$ bias shrinks to zero.

Section 6 presents a simple variant of our MRP that uses randomized quasi-Monte Carlo samples, as opposed to i.i.d. samples, to reduce variance. This method provides an asymptotically valid confidence interval just like the MRP, but the sampling error can be significantly reduced. Experiments with this method show that bias is also reduced.

Suppose we have a candidate solution, along with a fixed computational budget. Then, we can carry out

(i) the MRP of §3 with n_g instances of (SP_n) with sample size n formed using i.i.d. sampling;
(ii) the SRP (or 2RP) of §4 with a sample size $n' > n$;
(iii) the MRP^A procedure of §5 with n_g replications of $(\mathrm{SP}_{n''})$ with $n'' < n$; or
(iv) the MRP of §6 with n_g instances of (SP_n) with sample size n formed using randomized quasi-Monte Carlo sampling.

For any particular problem, either (i), (ii), (iii), or (iv) could be best. Often the ease of implementing (i) in addition to its applicability to a wide range of stochastic programs (e.g., linear, integer, or nonlinear stochastic programs) with minimal requirements makes it most attractive, at least if acceptable results can be obtained with modest computational effort. When this is not the case, the nature of the computational bottleneck often clearly points to which remedy to try. What we have not attempted is an adaptive scheme that would automatically identify and employ the "right" remedy.

That said, we have taken steps toward an adaptive approach. The SRP and 2RP approaches take as input the sample size n, and the MRP and MRP^A also require a replication size n_g. The user does not have direct control over the width of the resulting confidence interval, ϵ. The sequential procedure of §7 takes the desired confidence interval width as input and adaptively determines the required sample size.

The jackknife approach is just one way to reduce bias, and quasi-Monte Carlo sampling is just one way to reduce variance. Ongoing work uses a probability metric approach to reduce bias in 2RP and MRP. Increasing the quality of these point and interval estimators by reducing bias and variance with minimal computational effort is critical to their success, especially when used within sequential sampling procedures. This is an important area for future research. Fully adaptive sequential methods that analyze the information obtained about the problem so far to identify and correct the aforementioned potential problems can yield powerful and efficient solution methods. The MRP of §3 has been applied with success for a decade now, whereas the remedies we have described here are relatively new. Work on constructing and assessing the quality of candidate policies for multistage stochastic programs is ongoing (Chiralaksanakul and Morton [9], Tanrisever et al. [60]). We believe that as these mature, they will also lend themselves to wide application and to being embedded in optimization software to provide reliable estimates on the quality of solutions for stochastic programs.

Acknowledgments

Section 5 is based on the doctoral thesis of Partani [48]. The authors thank Georg Pflug for valuable discussions with respect to Example 5 and are grateful to Jeff Linderoth for his helpful comments that improved an earlier draft. This research was supported by the National Science Foundation under Grants CMMI-0653916 and EFRI-0835930.

References

[1] S. Ahmed and A. Shapiro. The sample average approximation method for stochastic programs with integer recourse. *Optimization Online*, http://www.optimization-online.org, 2002.

[2] T. G. Bailey, P. A. Jensen, and D. P. Morton. Response surface analysis of two-stage stochastic linear programming with recourse. *Naval Research Logistics* 46:753–778, 1999.

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

[4] G. Bayraksan and D. P. Morton. A sequential sampling procedure for stochastic programming. *Operations Research*. Forthcoming.

[5] 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.

[6] A. Brooke, D. Kendrick, A. Meeraus, and R. Raman. GAMS, A User's Guide. GAMS Development Corporation, Washington, DC, http://www.gams.com/, 2006.

[7] R. E. Caflisch, W. J. Morokoff, and A. B. Owen. Valuation of mortgage backed securities using Brownian bridges to reduce effective dimension. *Journal of Computational Finance* 1:27–46, 1997.

[8] G. Casella and R. L. Berger. *Statistical Inference*. Duxbury Press, Belmont, CA, 1990.

[9] A. Chiralaksanakul and D. P. Morton. Assessing policy quality in multi-stage stochastic programming. The Stochastic Programming E-Print Series. http://www.speps.org, 2004.

[10] Y. S. Chow and H. Robbins. On the asymptotic theory of fixed-width sequential confidence intervals for the mean. *Annals of Mathematical Statistics* 36:457–462, 1965.

[11] G. B. Dantzig and P. W. Glynn. Parallel processors for planning under uncertainty. *Annals of Operations Research* 22:1–21, 1990.

[12] G. B. Dantzig and G. Infanger. A probabilistic lower bound for two-stage stochastic programs. Technical Report SOL 95-6, Department of Operations Research, Stanford University, Stanford, CA, 1995.

[13] U. M. Diwekar and J. R. Kalagnanam. An efficient sampling technique for optimization under uncertainty. *American Institute of Chemical Engineers Journal* 43:440–447, 1997.

[14] B. L. Fox. *Strategies for Quasi-Monte Carlo*. Kluwer Academic Publishers, Boston, 1999.

[15] M. Freimer, D. Thomas, and J. T. Linderoth. The impact of sampling methods on bias and variance in stochastic linear programs. Technical Report 05T-002, Industrial and Systems Engineering, Lehigh University, Bethlehem, PA, 2005.

[16] P. W. Glynn and W. Whitt. The asymptotic validity of sequential stopping rules for stochastic simulations. *The Annals of Applied Probability* 2:180–198, 1992.

[17] H. L. Gray and W. R. Schucany. *The Generalized Jackknife Statistic*. Marcel Dekker, New York, 1972.

[18] J. L. Higle. Variance reduction and objective function evaluation in stochastic linear programs. *INFORMS Journal on Computing* 10:236–247, 1998.

[19] J. L. Higle and S. Sen. Statistical verification of optimality conditions for stochastic programs with recourse. *Annals of Operations Research* 30:215–240, 1991.

[20] J. L. Higle and S. Sen. Stochastic decomposition: An algorithm for two-stage linear programs with recourse. *Mathematics of Operations Research* 16:650–669, 1991.

[21] J. L. Higle and S. Sen. Duality and statistical tests of optimality for two stage stochastic programs. *Mathematical Programming* 75:257–275, 1996.

[22] J. L. Higle and S. Sen. *Stochastic Decomposition: A Statistical Method for Large Scale Stochastic Linear Programming*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996.

[23] J. L. Higle and S. Sen. Statistical approximations for stochastic linear programming problems. *Annals of Operations Research* 85:173–192, 1999.

[24] T. Homem-de-Mello. Variable-sample methods for stochastic optimization. *ACM Transactions on Modeling and Computer Simulation* 13:108–133, 2003.

[25] T. Homem-de-Mello. On rates of convergence for stochastic optimization problems under non-independent and identically distributed sampling. *SIAM Journal on Optimization* 19:524–551, 2008.

[26] G. Infanger. Monte Carlo (importance) sampling within a Benders decomposition algorithm for stochastic linear programs. *Annals of Operations Research* 39:69–95, 1992.

[27] G. Infanger. *Planning Under Uncertainty: Solving Large-Scale Stochastic Linear Programs*. The Scientific Press Series, Boyd & Fraser, Danvers, MA, 1994.

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

[29] B. D. Keller and G. Bayraksan. Scheduling jobs sharing multiple resources under uncertainty: A stochastic programming approach. *IIE Transactions on Operations Engineering*. Forthcoming.

[30] A. S. Kenyon and D. P. Morton. Stochastic vehicle routing with random travel times. *Transportation Science* 37:69–82, 2003.

[31] A. J. Kleywegt, A. Shapiro, and T. Homem-de-Mello. The sample average approximation method for stochastic discrete optimization. *SIAM Journal of Optimization* 12:479–502, 2001.

[32] G. Lan, A. Nemirovski, and A. Shapiro. Validation analysis of robust stochastic approximation method. *Optimization Online*, http://www.optimization-online.org, 2008.

[33] A. M. Law. *Simulation Modeling and Analysis*, 4th ed. McGraw-Hill, Boston, 2007.

[34] A. M. Law and W. D. Kelton. Confidence intervals for steady-state simulations II: A survey of sequential procedures. *Management Science* 28:550–562, 1982.

[35] A. M. Law, W. D. Kelton, and L. W. Koenig. Relative width sequential confidence intervals for the mean. *Communications in Statistics* B10:29–39, 1981.

[36] P. L'Ecuyer and C. Lemieux. Variance reduction via lattice rules. *Management Science* 46:1214–1235, 2000.

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

[38] J. Luedtke and S. Ahmed. A sample approximation approach for optimization with probabilistic constraints. *SIAM Journal on Optimization* 19:674–699, 2008.

[39] W. K. Mak, D. P. Morton, and R. K. Wood. Monte Carlo bounding techniques for determining solution quality in stochastic programs. *Operations Research Letters* 24:47–56, 1999.

[40] D. P. Morton and E. Popova. A Bayesian stochastic programming approach to an employee scheduling problem. *IIE Transactions on Operations Engineering* 36:155–167, 2003.

[41] D. P. Morton and R. K. Wood. On a stochastic knapsack problem and generalizations. D. L. Woodruff, ed. *Advances in Computational and Stochastic Optimization, Logic Programming, and Heuristic Search: Interfaces in Computer Science and Operations Research*. Kluwer Academic Publishers, Boston, 149–168, 1998.

[42] D. P. Morton, F. Pan, and K. J. Saeger. Models for nuclear smuggling interdiction. *IIE Transactions on Operations Engineering* 38:3–14, 2007.

[43] D. P. Morton, E. Popova, and I. Popova. Efficient fund of hedge funds construction under downside risk measures. *Journal of Banking and Finance* 30:503–518, 2006.

[44] A. Nadas. An extension of a theorem of Chow and Robbins on sequential confidence intervals for the mean. *Annals of Mathematical Statistics* 40:667–671, 1969.

[45] H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods*. CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM, Philadelphia, 1992.

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

[47] F. Pan and D. P. Morton. Minimizing a stochastic maximum-reliability path. *Networks* 52:111–119, 2008.

[48] A. Partani. Adaptive jackknife estimators for stochastic programming. Ph.D. thesis, The University of Texas at Austin, Austin, 2007.

[49] A. Partani, D. P. Morton, and I. Popova. Jackknife estimators for reducing bias in asset allocation. *Proceedings of the Winter Simulation Conference*, 783–791, 2006.

[50] T. Pennanen and M. Koivu. Epi-convergent discretizations of stochastic programs via integration quadratures. *Numerische Mathematik* 100:141–163, 2005.

[51] E. Polak and J. O. Royset. Efficient sample sizes in stochastic nonlinear programming. *Journal of Computational and Applied Mathematics* 217:301–310, 2008.

[52] A. Prékopa. *Stochastic Programming*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1995.

[53] M. H. Quenouille. Approximate tests of correlation in time-series. *Journal of the Royal Statistical Society, Series B* 11:68–84, 1949.

[54] M. H. Quenouille. Problems in plane sampling. *Annals of Mathematical Statistics* 20:355–375, 1949.

[55] 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.

[56] A. Shapiro. Monte Carlo sampling methods. A. Ruszczyński and A. Shapiro, eds. *Stochastic Programming, Handbooks in Operations Research and Management Science*, Vol. 10. Elsevier, Amsterdam, 353–425, 2003.

[57] A. Shapiro and T. Homem-de-Mello. A simulation-based approach to two-stage stochastic programming with recourse. *Mathematical Programming* 81:301–325, 1998.

[58] A. Shapiro, T. Homem de Mello, and J. Kim. Conditioning of convex piecewise linear stochastic programs. *Mathematical Programming* 94:1–19, 2002.

[59] I. H. Sloan and S. Joe. *Lattice Methods for Multiple Integration*. Clarendon Press, Oxford, UK, 1994.

[60] F. Tanrisever, D. Morrice, and D. P. Morton. Managing capacity flexibility in make-to-order production environments. Working paper, Information Risk and Operations Management Department, The University of Texas at Austin, Austin, 2009.

[61] B. Verweij, S. Ahmed, A. Kleywegt, G. Nemhauser, and A. Shapiro. The sample average approximation method applied to stochastic vehicle routing problems: A computational study. *Computational and Applied Optimization* 24:289–333, 2003.

[62] S. W. Wallace and W. T. Ziemba, eds. *Applications of Stochastic Programming*. MPS-SIAM Series in Optimization, 2005.

[63] D. W. Watkins, Jr., D. C. McKinney, and D. P. Morton. Groundwater pollution control. S. W. Wallace and W. T. Ziemba, eds. *Applications of Stochastic Programming*. MPS-SIAM Series on Optimization, 409–424, 2005.