Abstract

We develop a sequential sampling procedure for a class of stochastic programs. We assume that a sequence of feasible solutions with an optimal limit point is given as input to our procedure. Such a sequence can be generated by solving a series of sampling problems with increasing sample size, or, it can be found by any other viable method. Our procedure estimates the optimality gap of a candidate solution from this sequence. If the point estimate of the optimality gap is sufficiently small according to our termination criterion then we stop. Otherwise, we repeat with the next candidate solution from the sequence under an increased sample size. We provide conditions under which this procedure: (i) terminates with probability one and (ii) terminates with a solution which has a small optimality gap with a prespecified probability.

1 Introduction

Stochastic programming provides a means for decision making under uncertainty by incorporating random variables and probabilistic statements into deterministic optimization models. A major challenge in the analysis of stochastic programs of practical size is having to consider a large, sometimes infinite, number of scenarios. This usually leads to intractable models, even when specially-designed algorithms are used. Monte Carlo sampling-based methods provide an attractive approximation when the number of stochastic parameters is large. These methods replace the probabilistic statements that appear in the model (e.g., expectation) with sampling-based estimators (e.g., sample mean). They are usually justified asymptotically, by providing conditions under which the approximating solutions solve the stochastic program as the sample size grows to infinity. However, practical implementations require a finite sample size. A question of interest is then, what this sample size should be to have a good approximate solution. One way to approach this problem is to sequentially increase the sample size until we reach a good solution. Such a procedure requires a reliable means of stopping and rules to increase the sample size. Moreover, a statement regarding the quality of the resulting solution is needed.

In this paper, we develop such a sequential sampling procedure for a class of stochastic programs. In the procedure we propose, we assume we are given as input a sequence of candidate solutions, which has at least one limit point that solves the original stochastic program. Such a sequence can be generated by solving a series of sampling problems with increasing sample size. However, we allow candidate solutions to be generated by any method satisfying the above condition. Given a candidate solution, we assess its quality with increasing sample size. We measure quality in terms of the candidate solution’s optimality gap. At each iteration when assessing the candidate solution’s quality, we have the option to: (i) reuse previously-generated sample observations and add additional, newly-generated samples, or (ii) generate an entire new set of observations. Our ability to use a warm start to more efficiently solve the next iteration’s stochastic program favors the former choice. On the other hand, the risk of persisting with a “bad” set of samples suggests we should occasionally choose the latter option. We investigate this tradeoff. We terminate the procedure when a stopping criterion is satisfied, and we prove that asymptotically, this procedure yields a high quality solution with a desired probability.

Research on sequential sampling methods began in earnest in the 1940s and such methods have been successfully applied to problems in statistics, reliability, and statistical clinical testing (Ghosh and Sen 1991; Ghosh, Mukhopadhyay, and Sen 1997). In sequential sampling, the sample size is not fixed, but depends on the observations collected so far, and hence is a random variable. As a result, sequential estimation relates closely to the study of random walks hitting prespecified sets (Gut
A classic sequential problem involves forming a fixed-width (Chow and Robbins 1965) or relative-width (Nadas 1969) confidence interval for the mean by sequentially increasing the sample size. When simulating stochastic systems, the simulation run length takes the place of the sample size. For steady-state simulations, Law and Kelton (1982) and Law, Kelton, and Koenig (1981) survey sequential methods for constructing fixed-width and relative-width confidence intervals for the mean performance measure of a stochastic system. Glynn and Whitt (1992) provide conditions under which asymptotic validity of sequential stopping rules for simulations can be achieved. More recent work in the area of simulation involves selecting from a number of alternative system designs using sequential sampling; see Kim and Nelson (2001, 2006).

The stochastic programs we consider (see §2 for a formal definition) represent a large class of problems found in statistics and operations research. In statistics, a generalization of maximum likelihood estimators, called M-estimators (Huber 1981), are an example that are of this form. Sequential sampling procedures have been developed for M-estimators; see Hlávka (2000) and references therein. In these procedures, the objective function is assumed to be differentiable and the focus is on estimation of an optimum solution. In the optimization problems we consider, the objective function is frequently non-differentiable and we are indifferent to how close we are to the set of optimum solutions as long as the candidate solution’s objective function value is close to the optimum value, i.e., the optimality gap is small.

Even though the class of stochastic programs we consider can be found under various names in the literature, we mainly focus on stochastic programs with recourse. In particular, while our main results hold for more general models, our computational results are for two-stage stochastic linear programs with recourse (SLP-2). In SLP-2, the objective function is convex and typically also non-smooth. Stochastic quasigradient algorithms can be applied in such cases; they mimic steepest descent in which gradients or subgradients are replaced by sampling-based estimates. Ermoliev (1988) surveys such methods and Pflug (1988) surveys step-size and stopping rules. One advantage of stochastic quasigradient methods is that they have the potential to handle decision-dependent stochasticity. However, when applied to SLP-2, they make limited use of special structure.

The L-shaped method (Van Slyke and Wets 1969) better exploits SLP-2’s special structure and works well for problems with a modest number of scenarios. Dantzig and Glynn (1990) and Infanger (1992) use importance sampling to reduce estimation variance within the L-shaped method. Higle and Sen (1996b) develop an L-shaped method with sampling-based cuts, using a single stream of observations, where the cuts are updated to ensure desirable asymptotics. For stochastic global optimization, Norkin, Pflug, and Ruszczyński (1998) sample within a branch-and-bound algorithm.
Sen 1991; Higle and Sen 1996a), sequential issues that arise have received little attention.

Instead of embedding sampling within an optimization algorithm, such as an L-shaped or a branch-and-bound method, another approach is to first sample a set of observations and then simply solve the resulting problem. There is a significant literature on large sample size properties of this approach; see, e.g., the survey of Shapiro (2003). The sequence of candidate solutions our procedure requires could be generated either by these “internal” or “external” sampling methods.

For convex, piecewise linear stochastic programs that have a unique, sharp optimum, Shapiro, Homem-de-Mello, and Kim (2002) provide insight as to the sample size needed to find the optimal solution via large deviations theory. When independent samples are drawn at a sequence of iterations, Homem-de-Mello (2003) studies rates at which the sample sizes must grow to ensure consistency of the objective function estimator, and he derives associated error statements in the spirit of the law of the iterated logarithm. For stochastic nonlinear programs, Polak and Royset (2007) propose a procedure that approximately minimizes the computational effort required to reduce an initial optimality gap by a prespecified fraction, in the context of so-called diagonalization schemes.

Morton (1998) develops stopping rules for a class of algorithms that use asymptotically-normal optimality gap estimates, formed as a difference of upper- and lower-bound estimators. For minimization problems, upper-bound estimators can be formed for a fixed, feasible solution, and asymptotic normality is easy to achieve. A natural lower-bound estimator arises from optimizing a sample-mean objective function (Mak, Morton, and Wood 1999; Norkin, Pflug, and Ruszczyński 1998). Unfortunately, this estimator is not, in general, asymptotically normal, and so the approach of Morton (1998) does not apply. In this paper, we overcome this difficulty, and allow use of the non-normal optimized sample-mean estimator. We improve on Morton (1998) in two other important ways: The confidence interval we form on the optimality gap uses the (observable) sample variance estimator instead of the (unknown) population variance, and we develop sampling procedures under weaker moment conditions. Importantly, our main results do not require independent and identically distributed (i.i.d.) sampling, and hence can be applied when using other sampling schemes that are designed, e.g., to reduce variance. Another attractive feature of our sequential approach is its flexibility in how observations can be generated at each iteration. 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. Alternatively, the observations from the previous iteration can be discarded and we can generate an entirely new set of observations of increased size. Intermediate options also exist, and are permitted by the theory we develop.

We begin the next section with the class of stochastic programs we consider and provide assumptions that lay the framework for our sequential procedure. In §3, we give an outline of the
sequential procedure, stating the stopping rule, the rule to increase the sample sizes and a statement regarding the quality of the solution obtained. In §4, we establish theoretical properties of the sequential procedure under a moment generating function assumption (§4.1) and under weaker moment conditions (§4.2). Then, in §5, we discuss how to choose some parameters of the sequential procedure to minimize computational effort. In §6, we apply the procedure to two-stage stochastic linear programs with recourse and present and discuss computational results. We end the paper with a summary and future research directions (§7).

2 Framework

Our aim is to find a high quality (optimal or near-optimal) solution to a stochastic program of the form

$$z^* = \min_{x \in X} Ef(x, \tilde{\xi}),$$

(\text{SP})

where the expectation is taken with respect to the distribution of the random vector $\tilde{\xi}$. We assume that the distribution of $\tilde{\xi}$ is known, does not depend on the decision $x$, and that we can sample from it. We further assume $X \neq \emptyset$, $X$ is compact, $E \sup_{x \in X} |f(x, \tilde{\xi})| < \infty$, and $f(\cdot, \tilde{\xi})$ is lower semicontinuous (lsc) on $X$, with probability one (w.p.1). This ensures $Ef(\cdot, \tilde{\xi})$ is lsc, and hence (SP) has a finite optimal solution, which is achieved on $X$. Later we impose more restrictive moment conditions on $f(\cdot, \tilde{\xi})$. We also assume $f(x, \tilde{\xi})$ can be evaluated given a realization of $\tilde{\xi}$ and $x \in X$.

The procedure we propose works as follows: At iteration $k$, we are given a candidate solution, $\hat{x}_k \in X$. We select a sample size, $n_k$, and evaluate this candidate solution. The quality of a candidate solution is defined via its optimality gap: the smaller the optimality gap, the higher the quality. The procedure stops when the optimality gap estimate falls below a certain level. Otherwise, we continue with $\hat{x}_{k+1} \in X$ and $n_{k+1} \geq n_k$. Let $X^*$ denote the set of optimal solutions to (SP). We assume the following with respect to the sequence of candidate solutions.

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

Such a sequence can be found by solving a series of sampling problems

$$z^*_n = \min_{x \in X} \frac{1}{n} \sum_{i=1}^{n} f(x, \tilde{\xi}^i),$$

(\text{SP}_n)

with optimal solutions $x^*_n$ and with sample sizes $n = m_k$, such that $m_k \to \infty$ as $k \to \infty$. Mild
conditions under which such a sequence satisfies A1 (and $z_n^* \to z^*$, w.p.1) can be found, e.g., in Attouch and Wets (1981), Dupačová and Wets (1988), and Rubinstein and Shapiro (1993). However, we note that the sequence of candidate solutions is an input to our sequential procedure and can be generated by any other method, e.g., by the stochastic decomposition algorithm of Higle and Sen (1996b). That said, we assume that the method that generates $\{\hat{x}_k\}$ does not depend on the sampled observations used in our evaluation procedures.

For any $x \in X$, let $\mu_x = Ef(x, \hat{\xi}) - z^*$ and $\sigma^2(x) = \text{var}[f(x, \hat{\xi}) - f(x_{\min}^*, \hat{\xi})]$, where $x_{\min}^* \in \arg\min_{y \in X^*} \text{var}[f(x, \hat{\xi}) - f(y, \hat{\xi})]$. If (SP) has a unique optimum, i.e., $X^* = \{x^*\}$, then $x_{\min}^* = x^*$. When there are multiple optimal solutions to (SP), $x_{\min}^*$ depends on the given $x$ and is the optimal solution with the minimum variance of the gap $f(x, \hat{\xi}) - f(y, \hat{\xi})$ over $y \in X^*$.

Let $\xi^1, \xi^2, \ldots, \xi^n$ be a sample of size $n$. These observations could be i.i.d. as $\hat{\xi}$ or could be drawn in some other way, e.g., to reduce variance. Suppose we have at hand $G_n(x)$, that uses this sample of size $n$ to estimate the optimality gap of $x$, $\mu_x$, and similarly we have an estimator $s_n^2(x) \geq 0$, of the associated variance term, $\sigma^2(x)$. We define

$$D_n(x) = \frac{1}{n} \sum_{i=1}^{n} \left[ f(x, \xi^i) - f(x_{\min}^*, \tilde{\xi}^i) \right],$$

where $x_{\min}^*$ is defined as above for this $x$. The estimators $D_n(x), G_n(x)$ and $s_n^2(x)$ all use the same $n$ observations $\xi^1, \xi^2, \ldots, \xi^n$, and we assume the following:

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, $\liminf_{k \to \infty} P(|G_{n_k}(x_k) - \mu_x| > \delta) = 0$ for any $\delta > 0$.

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

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

A5. $\sqrt{n}(D_n(x) - \mu_x) \implies N(0, \sigma^2(x))$ 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 $\sigma^2(x)$. Here, $\implies$ denotes convergence in distribution.

As a concrete example of these estimators that satisfy the above assumptions, let $\hat{\xi}^1, \hat{\xi}^2, \ldots, \hat{\xi}^n$ be i.i.d. as $\hat{\xi}$, let $x_n^*$ solve $(SP_n)$ for this sample, let $\tilde{f}_n(x) = \frac{1}{n} \sum_{i=1}^{n} f(x, \hat{\xi}^i)$ and form:

$$G_n(x) = \frac{1}{n} \sum_{i=1}^{n} \left( f(x, \hat{\xi}^i) - f(x_n^*, \hat{\xi}^i) \right), \quad (2a)$$

$$s_n^2(x) = \frac{1}{n-1} \sum_{i=1}^{n} \left[ f(x, \hat{\xi}^i) - f(x_n^*, \hat{\xi}^i) \right]^2. \quad (2b)$$
Here, $G_n(x)$ is equivalently expressed as $G_n(x) = \tilde{f}_n(x) - \hat{g}_n(x_n^*)$, and we have $\tilde{f}_n(x_n^*) = z_n^*$. The expression in (2a) motivates the use of the $n$ individual observations of the gap in the sample variance estimator (2b). With this choice of estimators, Assumption A3 is immediate since $\tilde{f}_n(x_n^*) \leq \hat{f}_n(x_n^*)$, w.p.1. Under i.i.d. sampling, $D_n(x)$ is simply a sample mean of i.i.d. observations, and hence if $\sigma^2(x) < \infty$ then A5 holds by the standard central limit theorem (CLT) for i.i.d. random variables. Note that if $\sigma^2(x) = 0$, then $f(x, \tilde{\xi}) - f(x_{\min}, \hat{\xi}) = \mu_x$, for almost all $\tilde{\xi}$ and A5 still holds, albeit in degenerate form. Sufficient conditions for A4 to hold under i.i.d. sampling are given in Bayraksan and Morton (2006). 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 $n \rightarrow \infty$. However, A4 is a form of consistency for $s_n^2(x)$ in the sense that it is bounded below by $\sigma^2(x)$. (Recall, $\sigma^2(x)$ is defined with respect to $x_{\min}$.) In general, a sufficient condition under which A2 holds is that $\tilde{f}_n(x)$ converges uniformly to $E\tilde{f}(x, \tilde{\xi})$ on $X$, w.p.1. This holds under i.i.d. sampling and compact $X$ provided $f(\cdot, \tilde{\xi})$ is continuous on $X$, w.p.1, and $E\sup_{x \in X} |f(x, \tilde{\xi})| < \infty$; see, e.g., Shapiro (2003). We are now ready to present the sequential sampling procedure in more detail.

## 3 Sequential Sampling Procedure

At iteration $k \geq 1$ of the sequential procedure, we select a sample size, $n_k$, and we use $n_k$ total observations to assess the quality of the current solution, $\hat{x}_k$. We can choose to generate $\tilde{\xi}^1, \ldots, \tilde{\xi}^{n_k}$ independently of those generated in previous iterations. Or, we can augment the observations from the previous iteration by generating $n_k - n_{k-1}$ additional observations, $\tilde{\xi}^{n_{k-1}+1}, \ldots, \tilde{\xi}^{n_k}$ (assume $n_0 = 0$). To assess $\hat{x}_k$’s quality, we form an estimate of its optimality gap, $G_{n_k}(\hat{x}_k)$, and its variance, $s_{n_k}^2(\hat{x}_k)$. If the current candidate solution does not satisfy a stopping criterion, we repeat the above steps with sample size, $n_{k+1} \geq n_k$ and the next candidate solution $\hat{x}_{k+1}$. To simplify notation, from now on, we drop the dependence on the candidate solution, $\hat{x}_k$, and the sample size $n_k$, and simply denote $\mu_k = \mu_{\hat{x}_k}$, $\sigma_k = \sigma^2(\hat{x}_k)$, $D_k = D_{n_k}(\hat{x}_k)$, $G_k = G_{n_k}(\hat{x}_k)$ and $s_k = s_{n_k}(\hat{x}_k)$. We terminate the procedure at iteration $T$ when the following stopping criterion is satisfied,

$$T = \inf_{k \geq 1} \{ G_k \leq h's_k + \epsilon' \}.$$  

(3)

The random stopping time $T$ is the first time $G_k$’s width relative to $s_k$ falls below $h' > 0$ plus a small positive number $\epsilon'$, which ensures finite stopping (see next section). By choosing the stopping criterion as in (3), we are willing to accept larger optimality gap estimates for problems with large variability. Note that it is possible to add the condition $\{s_k \leq b\}$ to (3) so that when we stop, $G_k$ is below a certain fixed threshold, $h' b + \epsilon'$.
The stopping criterion (3) is with respect to $h s_T + \epsilon$, and the statement regarding the quality of the candidate solution when we stop is with respect to a larger relative term $h s_T + \epsilon$, where $h > h'$. (Typically, we choose the epsilon terms so they are small compared to $h'$.) Such inflation of the confidence interval statement, relative to the stopping criterion, is fairly standard when using sampling methods with a sequential nature; see for example, Chow and Robbins (1965) or Glynn and Whitt (1992). For the procedure we propose we will show

$$\lim_{h \downarrow h'} \inf P(\mu_T \leq h s_T + \epsilon) \geq 1 - \alpha.$$  \hspace{1cm} (4)

In other words, the optimality gap of $\hat{x}_T$, the candidate solution when we stop, is a fraction of the sample standard deviation plus $\epsilon$, with a desired probability provided $h$ is close enough to $h'$. As before, if we have the additional condition $\{s_T \leq b\}$, then, we asymptotically guarantee that the optimality gap of $\hat{x}_T$ is at most $h b + \epsilon$ with a confidence level of $1 - \alpha$.

At iteration $k$, we choose the sample size according to

$$n_k \geq \left(\frac{1}{h - h'}\right)^2 (c_p + 2p \ln^2 k),$$ \hspace{1cm} (5)

where $c_p = \max\{2 \ln \left(\sum_{k=1}^{\infty} k^{-p \ln k} / \sqrt{2\pi \alpha}\right), 1\}$. Here, $p > 0$ is a parameter that affects the number of samples we generate. We discuss how to choose $p$ in more detail below. The sample size growth formula in (5) is proportional to $(h - h')^{-2}$, it has a constant term, $c_p$, which depends on $p$, and it grows as $O(\log^2 k)$ with respect to the iteration number $k$. In the next section, we show if the sample size satisfies (5) then (4) holds under a finite moment generating function assumption. We also show the procedure stops in a finite number of steps. Before proceeding, we summarize our procedure.

**Sequential Sampling Procedure:**

**Input:** Values for $h > h' > 0$, $\epsilon > \epsilon' > 0$, $0 < \alpha < 1$, and $p > 0$. Method that generates candidate solutions $\{\hat{x}_k\}$ with at least one limit point in $X^\ast$. Resampling frequency $k_f$, a positive integer.

**Output:** A candidate solution, $\hat{x}_T$, and a $(1 - \alpha)$-level confidence interval on the optimality gap of the candidate solution, $\mu_T$.

0. (Initialization) Set $k = 1$, calculate $n_k$ as given in (5), and sample observations $\tilde{\xi}_1, \tilde{\xi}_2, \ldots, \tilde{\xi}_{n_k}$.

1. Use $\tilde{\xi}_1, \tilde{\xi}_2, \ldots, \tilde{\xi}_{n_k}$ to form $G_k$ and $s_k^2$.

2. If $\{G_k \leq h' s_k + \epsilon'\}$, then set $T = k$, and go to 4.
3. Set $k = k + 1$ and calculate $n_k$ according to (5). If $k_f$ divides $k$ then sample observations $\tilde{\xi}_1, \tilde{\xi}_2, \ldots, \tilde{\xi}_k$, independently of samples generated in previous iterations. Else, sample $n_k - n_{k-1}$ observations $\tilde{\xi}_{n_k-1+1}, \tilde{\xi}_{n_k-1+2}, \ldots, \tilde{\xi}_k$ from the distribution of $\tilde{\xi}$. Go to 1.

4. Output candidate solution $\hat{x}_T$ and a one-sided confidence interval on $\mu_T$:

$$[0, h s_T + \epsilon].$$ (6)

If the resampling frequency $k_f = 1$ then at every iteration we sample observations $\tilde{\xi}_1, \tilde{\xi}_2, \ldots, \tilde{\xi}_k$ independently of previously-generated observations. At the other extreme, if $k_f$ is sufficiently large then we continually augment the existing set of observations throughout execution of the procedure. Frequent resampling decreases the likelihood the procedure fails to terminate for a number of iterations because it is effectively stuck with a “bad” sample. Infrequent resampling increases the effectiveness of a computationally-efficient warm start for solving the stochastic program required to form $G_k$ and $s^2_k$ in step 1. We investigate the associated tradeoff in Section 6.

4 Asymptotic Validity and Finite Stopping

The stopping time, $T$, and the solution, $\hat{x}_T$, provided by our sequential sampling method are random variables. Therefore, unlike a deterministic optimization algorithm, statements regarding finite stopping and the quality of the solution must be probabilistic. In this section, we first present a result that shows (4) holds under a finite moment generating function assumption. Then, we prove and discuss finite stopping of the algorithm. In the next section we relax the moment generating function assumption, replacing it with a finite $r^{th}$ moment assumption.

4.1 Finite Moment Generating Function

Suppose $\sigma^2(x) > 0$ and assume for some $\gamma_0 > 0$

$$\sup_{n \geq 1} \sup_{x \in X} E \left[ e^{\gamma \left( \frac{D_n(x) - \mu_x}{\sigma(x)} \right)} \right] < \infty, \text{ for } |\gamma| \leq \gamma_0,$$ (7)

where $D_n(x)$ is defined in (1). When $\sigma^2(x) = 0$, $D_n(x)$ is constant almost everywhere for all $n$ and therefore has an MGF. When $\tilde{\xi}_1, \tilde{\xi}_2, \ldots, \tilde{\xi}_n$ are i.i.d. a sufficient condition to ensure (7) is that the moment generating function (MGF) of the scaled random variables $[f(x, \tilde{\xi}) - f(x^*_{\text{min}}, \tilde{\xi})] / \sigma(x)$
exists, i.e.,
\[ \sup_{x \in X} E \left[ e^{\gamma \left( \frac{f(x, \tilde{\xi}) - f(x^*_{\min}, \tilde{\xi}) - \mu_x}{\sigma_x} \right)} \right] < \infty \text{ for } |\gamma| \leq \gamma_0. \] (8)

Condition (8) can be somewhat restrictive, but it is satisfied when \( X \) is compact and the distribution of \( \tilde{\xi} \) has bounded support. More generally, with \( X \) compact, (8) holds when (SP) satisfies the following Lipschitz condition

\[ |f(x_1, \tilde{\xi}) - f(x_2, \tilde{\xi})| \leq K(\tilde{\xi})\|x_1 - x_2\|, \text{ w.p.1,} \] (9)

for all \( x_1, x_2 \in X \), where the Lipschitz constant \( K(\tilde{\xi}) \) satisfies \( E[e^{\gamma K(\tilde{\xi})}] < \infty \) for \( |\gamma| \leq \gamma_0 \). For a special class of (SP), namely, for two-stage stochastic linear programs with fixed recourse,

\[ f(x, \tilde{\xi}) = cx + \min_{y \geq 0} \tilde{q}y \]
\[ \text{s.t. } Wy = \tilde{r} - \tilde{T}x. \]

Suppose this linear program is dual feasible for almost all \( \tilde{q} \) and the stochastic program has relatively complete recourse. Further, suppose \( \tilde{q}, \tilde{r}, \) and \( \tilde{T} \) can each be expressed as a linear combination of the underlying random vector \( \tilde{\xi} \), which has independent components. Note that this allows for first-order dependencies between the components of \( (\tilde{q}, \tilde{r}, \tilde{T}) \). Then, the Lipschitz condition (9) and hence condition (8) will be satisfied when the squared Euclidean norm of the random vector, \( \tilde{\xi} \), has an MGF, i.e., \( E[e^{\gamma \|\tilde{\xi}\|^2}] < \infty \) for \( |\gamma| \leq \gamma_0 \); see Römisch (2003, Proposition 22).

Below we state and prove the validity of the sequential sampling procedure under hypothesis (7). Our result, given in (4), is asymptotic, as \( h \downarrow h' \), i.e., as the sample sizes grow. We note that even in the simple case of constructing confidence intervals for the mean under sequential sampling the validity of the resulting confidence intervals, i.e., that the confidence interval has the desired coverage probability, is proven asymptotically (Chow and Robbins 1965). To prove (4), we make use of: (i) Fatou’s Lemma, which provides inequalities when “lim inf” and an integral (or an infinite sum) are exchanged and (ii) a bound on the tail of a normal random variable. These are given in the next two lemmas.

**Lemma 1 (Fatou’s Lemma)** Suppose \( \{f_n\} \) is a sequence of measurable functions on \( E \).

(i) If \( f_n \geq 0 \) for all \( n \) then
\[ \int_E \liminf_{n \to \infty} f_n \leq \liminf_{n \to \infty} \int_E f_n. \]
(ii) If \( L \leq f_n \leq U \) for all \( n \), such that \( \int_E L < \infty \) and \( \int_E U < \infty \), then,

\[
\int_E \liminf_{n \to \infty} f_n \leq \liminf_{n \to \infty} \int_E f_n \leq \limsup_{n \to \infty} \int_E f_n \leq \int_E \limsup_{n \to \infty} f_n.
\]

**Proof.** For a proof of part (i), see e.g., Rudin (1976). Part (ii) follows from applying part (i) to \((f_n - L)\) and \((U - f_n)\).

**Lemma 2 (Bound on tail of a standard normal)** Let \( Z \) be a standard normal and \( t > 0 \). Then,

\[
P(Z \geq t) \leq \frac{1}{\sqrt{2\pi}} e^{-t^2/2}
\]

**Proof.** See, e.g., Casella and Berger (1990, p.185).

**Theorem 3** Assume A3, A4 and A5 are satisfied and (7) holds for \( \gamma_0 > 0 \). Let \( M^2 = \sup_{x \in \mathcal{X}} \sigma^2(x) \), \( \epsilon > \epsilon' > 0 \), \( p > 0 \) and \( 0 < \alpha < 1 \). Consider the sequential sampling procedure where the sample size is increased according to (5). If the procedure stops at iteration \( T \) according to (3) then,

\[
\liminf_{h \downarrow h'} P(\mu_T \leq h s_T + \epsilon) \geq 1 - \alpha.
\]

**Proof.** Let \( \Delta h = h - h' \) and \( \Delta \epsilon = \epsilon - \epsilon' \). When we stop, \( G_T \leq h' s_T + \epsilon' \) implies

\[
P(\mu_T > h s_T + \epsilon)
\]

\[
\leq P(\mu_T \geq G_T + h s_T + \epsilon)
\]

\[
= \sum_{k=1}^{\infty} P(G_1 > h' s_1 + \epsilon', \ldots, G_{k-1} > h' s_{k-1} + \epsilon', G_k \leq h' s_k + \epsilon', G_k - \mu_k \leq -h s_k - \Delta \epsilon)
\]

\[
\leq \sum_{k=1}^{\infty} P(G_k - \mu_k \leq -h s_k - \Delta \epsilon)
\]

\[
\leq \sum_{k=1}^{\infty} P(D_k - \mu_k \leq -h s_k - \Delta \epsilon),
\]

where inequality (11) follows from A3. So, it suffices to show

\[
\limsup_{\Delta h \downarrow 0} \sum_{k=1}^{\infty} P(D_k - \mu_k \leq -h s_k - \Delta \epsilon) \leq \alpha.
\]

11
To apply part (ii) of Fatou’s lemma, we first show that the right-hand side of (11) is bounded above for all positive \( \Delta h \) that are sufficiently small:

\[
\sum_{k=1}^{\infty} P \left( D_k - \mu_k \leq -\Delta h s_k - \Delta \epsilon \right) \\
\leq \sum_{k=1}^{\infty} P \left( D_k - \mu_k \leq -\Delta \epsilon \right) \\
= \sum_{k=1}^{\infty} \int_{\hat{x}_k} P \left( D_k - \mu_k \leq -\Delta \epsilon \mid \hat{x}_k \right) dP_{\hat{x}_k}
\]

(12)

\[
\leq \int_{\hat{x}_k} E \left[ e^{-\gamma_0 \left( \frac{D_k - \mu_k}{\sigma_k / \sqrt{n_k}} \right)} \mid \hat{x}_k \right] e^{-\gamma_0 (\Delta \epsilon / \sigma_k)} dP_{\hat{x}_k}
\]

(13)

\[
\leq \sup_{k \geq 1} \sup_{x \in X} E \left[ e^{-\gamma_0 \left( \frac{D_k - \mu_k}{\sigma_k / \sqrt{n_k}} \right)} \right] \sum_{k=1}^{\infty} k \left( \frac{\gamma_0 \sqrt{\Delta \epsilon}}{M \Delta h} \right),
\]

(14)

where (13) follows from an application of the Chernoff bound (e.g., Ross (1998)), to the conditional probability in (12). Note that for \( \hat{x}_k \) with \( \sigma_k^2 = 0 \), the probability in (12) is 0. The first term on the right-hand side of (14) is bounded by (7) and the second term is bounded for all sufficiently small \( \Delta h \), i.e., for all \( 0 < \Delta h < \gamma_0 \sqrt{2} \Delta \epsilon / M \). Taking limits we obtain,

\[
\limsup_{\Delta h \downarrow 0} \sum_{k=1}^{\infty} P \left( D_k - \mu_k \leq -\Delta h s_k - \Delta \epsilon \right) \\
\leq \sum_{k=1}^{\infty} \limsup_{\Delta h \downarrow 0} P \left( D_k - \mu_k \leq -\Delta h s_k - \Delta \epsilon \right) \\
\leq \sum_{k=1}^{\infty} \limsup_{\Delta h \downarrow 0} \int_{\hat{x}_k} P \left( \frac{D_k - \mu_k}{\sigma_k / \sqrt{n_k}} \leq -\Delta h \sqrt{n_k} \left( \frac{s_k}{\sigma_k} \right) \mid \hat{x}_k \right) dP_{\hat{x}_k}
\]

\[
\leq \sum_{k=1}^{\infty} \int_{\hat{x}_k} \limsup_{\Delta h \downarrow 0} P \left( \frac{D_k - \mu_k}{\sigma_k / \sqrt{n_k}} \leq -\left( c_p + 2p \ln^2 k \right)^{1/2} \left( \frac{s_k}{\sigma_k} \right) \mid \hat{x}_k \right) dP_{\hat{x}_k}
\]

\[
\leq \alpha,
\]

where the first and the third inequalities follow from an application of Fatou’s lemma. With \( k \) and \( \hat{x}_k \) fixed, \( \frac{D_k - \mu_k}{\sigma_k / \sqrt{n_k}} \) converges to a standard normal by A5 as \( \Delta h \downarrow 0 \) and hence \( n_k \to \infty \). Similarly, \( \liminf_{\Delta h \downarrow 0} (s_k / \sigma_k) \geq 1 \) by A4. The last inequality then follows by applying Lemma 2 and the definition of \( c_p \).

Theorem 3 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 \( 1 - \alpha \). We now turn our attention to finite stopping and show that the
sequential procedure stops with probability one. We state this formally in the proposition below.

**Proposition 4** Assume A1 and A2 are satisfied. Let $\epsilon' > 0$ and $h > h' > 0$ be fixed. Then, for the sequential sampling procedure where the sample size is increased according to (5), and the procedure stops at iteration $T$ according to (3), we have $P(T < \infty) = 1$.

**Proof.** Note that

$$P(T = \infty) \leq \liminf_{k \to \infty} P(G_k > h's_k + \epsilon')$$

$$\leq \liminf_{k \to \infty} P(G_k > \epsilon')$$

$$\leq \liminf_{k \to \infty} P(|G_k - \mu_k| > \epsilon' - \mu_k),$$

and the final term is zero by A1 and A2.  

4.2 **Weaker Moment Conditions**

In this section, we prove a variant of Theorem 3 that assumes finite moments up to order $r$. Specifically, we relax the MGF assumption of the previous section to

$$\sup_{x \in X} E \left[ f(x, \tilde{\xi}) \right]^r < \infty,$$

for some even integer $r \geq 2$. Note that (15) implies $\sup_{x \in X, y \in X} E[f(x, \tilde{\xi}) - f(y, \tilde{\xi})]^r < \infty$. We also simplify the analysis by only considering i.i.d. sampling. Under these assumptions, we select the sample size at each iteration $k$ according to

$$n_k \geq \left( \frac{1}{h - h'} \right)^2 \left( c_{p,q} + 2pk^{2q/r} \right),$$

where $q > 1$, $p > 0$ and where $c_{p,q} = \max\{2 \ln \left( \sum_{k=1}^{\infty} \exp(-pk^{2q/r})/\sqrt{2\pi} \alpha \right), 1\}$. The growth in the sample size is of order $O(k^{2q/r})$ and we must choose $q > 1$. So, if condition (15) holds for $r = 2$ we can choose $q$ just larger than unity so we essentially have the sample size growing at a linear rate, and if condition (15) holds for $r = 4$ we can obtain a rate that essentially grows with $k^{1/2}$. In other words, a less restrictive assumption on the existence of moments implies a faster rate of growth for the sample sizes.

Under the moment generating function hypothesis, the sample size formula (5) contained a parameter, $p > 0$, which was at our disposal. Now, under the moment hypothesis (15), if we elect
the slowest possible rate of growth of $n_k$ by choosing $q$ just larger than unity then we can again view the sample size formula as being parameterized by a scalar term, $p$.

We now state the validity and finite stopping of our procedure under the finite $r^{th}$ moment assumption given in (15). To do this, we need a lemma that establishes a bound on the central moments of a sample mean. We do this in two steps, in Lemmas 5 and 4.2.

**Lemma 5** There are no more than $\left\lfloor \sqrt{rn} \right\rfloor^r$ ways to place $r$ distinguishable balls in $n$ distinguishable bins so that no bin contains exactly one ball.

**Proof.** Let $\gamma^r_n$ denote the quantity in question; we must show $\gamma^r_n \leq \left\lfloor \sqrt{rn} \right\rfloor^r$. The desired result holds when $r \geq n$ and is also immediate for $r \leq 3$. Thus, we restrict attention to the case $r \geq 4$ and $n \geq r$ and proceed by induction. By conditioning on whether the first two balls are placed in the same or different bins we obtain the following recursion:

$$\gamma^r_n = n \sum_{j=0}^{r-2} \binom{r-2}{j} \gamma^{r-2-j}_{n-1} + n(n-1) \sum_{j=2}^{r-2} \binom{r-2}{j} (2j-2) \gamma^{r-2-j}_{n-2}.$$

It is clear, after reindexing the sum in the second term of the right-hand side of the recursion, that to verify the inductive step it suffices to show:

$$r^{r/2}n^{r/2} \geq n \sum_{j=0}^{r-2} \binom{r-2}{j} (r-2-j)^{(r-2-j)/2} (n-1)^{(r-2-j)/2}$$

$$+ n(n-1) \sum_{j=2}^{r-4} \binom{r-2}{j+2} (2j+2-2) (r-4-j)^{(r-4-j)/2} (n-2)^{(r-4-j)/2}.$$

Observe for the range of $r$ and $j$ under consideration

$$\binom{r-2}{j+2} (2j+2-2) (r-4-j)^{(r-4-j)/2} \leq \binom{r-4}{j} 2^j (r-2)(r-3)(r-4)^{(r-4-j)/2}.$$

Applying the binomial theorem we see it suffices to show

$$r^{r/2}n^{r/2} \geq n \left[ 1 + \sqrt{(r-2)(n-1)} \right]^{r-2} + n(n-1)(r-2)(r-3) \left[ 2 + \sqrt{(r-4)(n-2)} \right]^{r-4}. \quad (17)$$
Further observe for our range of \( r \) and \( n \),

\[
1 + \sqrt{(r-2)(n-1)} \leq \sqrt{rn}, \tag{18a}
\]

\[
2 + \sqrt{(r-4)(n-2)} \leq \sqrt{rn}. \tag{18b}
\]

The inductive step may be verified upon substitution of inequalities (18) into (17).

**Lemma 6** Let \( X^1, X^2, \ldots, X^n \) be i.i.d. random variables with mean \( \mu \) and \( \bar{X}_n = \frac{1}{n} \sum_{i=1}^{n} X^i \). If \( E|X^1 - \mu|^r < \infty \) for some integer \( r \geq 1 \) then

\[
E(\bar{X}_n - \mu)^r \leq E|X^1 - \mu|^r \left( \frac{r}{n} \right)^{r/2}. \]

**Proof.** Let \( Y^i = X^i - \mu \). Then, it suffices to verify

\[
E \left( \sum_{i=1}^{n} Y^i \right)^r \leq E|Y^1|^r \left( \sqrt{rn} \right)^r.
\]

Let \( k = (k_1, \ldots, k_n) \), \( \mathcal{R}_n = \{ k : \sum_{i=1}^{n} k_i = r, k \geq 0 \} \), and \( \mathcal{R}_n^- = \{ k : k \in \mathcal{R}_n, k_i \neq 1, i = 1, \ldots, n \} \). Now,

\[
E \left( \sum_{i=1}^{n} Y^i \right)^r = \sum_{k \in \mathcal{R}_n} \frac{r!}{k_1! \cdots k_n!} \prod_{i=1}^{n} E(Y^1)^{k_i} \leq E|Y^1|^r \sum_{k \in \mathcal{R}_n^-} \frac{r!}{k_1! \cdots k_n!}
\]

since

\[
\max_{k \in \mathcal{R}_n} \prod_{i=1}^{n} E(Y^1)^{k_i} \leq E|Y^1|^r. \tag{19}
\]

Inequality (19) follows from the fact that \([E|Y^1|^s]^{1/s}\) is nondecreasing on \([1, r]\); see, e.g., Casella and Berger (1990, p.180). It remains to show

\[
\sum_{k \in \mathcal{R}_n^-} \frac{r!}{k_1! \cdots k_n!} \leq \left( \sqrt{rn} \right)^r,
\]

which follows immediately from Lemma 5.

We are now ready to prove the validity and finite stopping of our procedure under the finite \( r^{th} \) moment assumption given in (15).
Theorem 7 Assume $A1-A4$ are satisfied, (15) holds with $r \geq 2$ even, and that $\tilde{\xi}^1, \tilde{\xi}^2, \ldots$ are i.i.d. as $\xi$. Let $\epsilon > \epsilon' > 0$, $p > 0$, $q > 1$ and $0 < \alpha < 1$. Then, for the sequential sampling procedure where the sample size is increased according to (16), and the procedure stops at iteration $T$ according to (3),

$$P(T < \infty) = 1 \quad \text{and} \quad \liminf_{h \downarrow h'} P(\mu_T \leq hs_T + \epsilon) \geq 1 - \alpha.$$  

Proof. The proof of the finite stopping result is identical to that of Proposition 4. To prove asymptotic validity, we start as in the proof of Theorem 3 and proceed until (12) the same way. Then, instead of using a Chernoff bound, which is Markov’s inequality applied to the exponent of a random variable, we use Markov’s inequality with $r$th moment under assumption (15),

$$\sum_{k=1}^{\infty} \int_{\tilde{\xi}_k} P(D_k - \mu_k \leq -\Delta \epsilon | \tilde{\xi}_k) dP_{\tilde{\xi}_k} 
\leq \sum_{k=1}^{\infty} \int_{\tilde{\xi}_k} P(|D_k - \mu_k| \geq \Delta \epsilon | \tilde{\xi}_k) dP_{\tilde{\xi}_k} 
\leq \sum_{k=1}^{\infty} \int_{\tilde{\xi}_k} E[(D_k - \mu_k)^r | \tilde{\xi}_k] \Delta \epsilon^{-r} dP_{\tilde{\xi}_k} 
\leq \sup_{x \in X} E\left[f(x, \tilde{\xi}) - f(x_{\min}^*, \tilde{\xi}) - \mu_x \right]^{r/2} \Delta \epsilon^{-r} \sum_{k=1}^{\infty} \frac{1}{n_k^{r/2}},$$  

(20)

where (20) follows from Lemma 4.2. From the definition of $n_k$ given in (16), the right-hand side of (20) is bounded. The hypothesis that $\tilde{\xi}_1, \tilde{\xi}_2, \ldots$ are i.i.d. along with assumption (15) holding for $r \geq 2$ implies that assumption A5 holds. As a result, the rest of the proof is analogous to that of Theorem 3. \[ \square \]

Under i.i.d. sampling, Theorem 7 differs from Theorem 3 in that the MGF assumption of (8) is replaced by the moment condition (15). Under this weaker assumption, the sample sizes are chosen according to (16) instead of (5), which require a larger number of sampled observations.

So far, we have verified desirable theoretical properties of our sequential procedure. In the next section, we discuss issues that arise when implementing the procedure. In particular, we discuss how to choose $p$ when using the sample size formula (5) and how to choose $q$ and $p$ when using (16). Then, we apply the procedure to two test problems that are two-stage stochastic linear programs with recourse to examine its performance.
At iteration $k$ of the sequential procedure, we determine the sample size according to (5) under the MGF assumption and according to (16) under the $r$th moment assumption. Then, using this many samples, we solve a sampling problem (SP$_{n_k}$) to estimate the optimality gap of the current candidate solution and its associated variance. Suppose the procedure terminates in $T$ iterations. Then, we solve $T$ sampling problems, (SP$_{n_1}$), (SP$_{n_2}$), ..., (SP$_{n_T}$). Therefore, the computational effort exerted for the evaluation of the candidate solutions is approximately proportional to $\sum_{k=1}^{T} n_k$. (For some decomposition methods, empirical studies suggest the effort to solve a stochastic program grows linearly in the number of scenarios, see e.g., Ruszczyński and Świetanowski (1997) and Verweij et al. (2003)). In other words, the effort is proportional to

$$S_M(p) = T \max \left\{ 2 \ln \left( \frac{\sum_{k=1}^{\infty} k^{-p \ln k}}{\sqrt{2\pi \alpha}} \right), 1 \right\} + 2p \sum_{k=1}^{T} \ln^2 k$$

$$S_W(p, q) = T \max \left\{ 2 \ln \left( \frac{\sum_{k=1}^{\infty} \exp(-pk^{2q/r})}{\sqrt{2\pi \alpha}} \right), 1 \right\} + 2p \sum_{k=1}^{T} k^{2q/r},$$

in these two respective cases. Assume, for the moment, that $T$ is known. The parameter $p > 0$ for $S_M(p)$ and the parameters $p > 0$ and $q > 1$ for $S_W(p, q)$ are at our disposal. To reduce the computational effort, we would like to choose them to minimize $S_M(p)$ and $S_W(p, q)$. The following result helps to do so.

**Proposition 8** Let $S_M(p)$ and $S_W(p, q)$ be defined in (21). $S_M(\cdot)$ is convex on $\{p : p > 0\}$ and $S_W(\cdot, q)$ is convex on $\{p : p > 0\}$ for fixed $q > 1$. Furthermore, $S_M(p)$ and $S_W(p, q)$ are both bounded below by

$$2T \ln \left( \frac{T}{\sqrt{2\pi \alpha}} \right).$$

**Proof.** To prove the convexity result for $S_W(\cdot, q)$ it suffices to show $\ln \left( \sum_{k=1}^{\infty} \exp(-pk^{2q/r}) \right)$ is convex on $\{p : p > 0\}$. With $\lambda \in (0, 1)$ and $p, p' > 0$ this amounts to showing

$$\ln \left( \sum_{k=1}^{\infty} \exp \left[ - (\lambda p + (1 - \lambda)p') k^{2q/r} \right] \right) \leq \ln \left( \sum_{k=1}^{\infty} \exp \left[ -pk^{2q/r} \right] \right)^{\lambda} + \ln \left( \sum_{k=1}^{\infty} \exp \left[ -p'k^{2q/r} \right] \right)^{1-\lambda}.$$
which follows from Hölder’s inequality. The proof for convexity of \( S_M(\cdot) \) is similar; see also Proposition 5 of Morton (1998).

Fix \( q > 1 \). Then,

\[
\min_{p > 0} S_W(p, q) \geq \min_{p_1, \ldots, p_T > 0} 2T \ln \left( \sum_{k=1}^{\infty} \frac{\exp(-p_k k^{2q/r})}{\sqrt{2\pi \alpha}} \right) + 2 \sum_{k=1}^{T} p_k k^{2q/r}. \tag{23}
\]

The right-hand side of (23) represents a relaxation in that we can choose a different value of \( p \) for each \( k \). It is clearly optimal to set \( p_k = \infty \) for \( k > T \), and so the right-hand side optimization reduces to

\[
\min_{p_1, \ldots, p_T > 0} 2T \ln \left( \sum_{k=1}^{T} \frac{\exp(-p_k k^{2q/r})}{\sqrt{2\pi \alpha}} \right) + 2 \sum_{k=1}^{T} p_k k^{2q/r}. \tag{24}
\]

The same type of argument used above to establish convexity of \( S_W(\cdot, q) \) shows the objective function of (24) is convex in \((p_1, \ldots, p_T)\). Taking the partial derivatives with respect to \( p_k \) we obtain the following first-order conditions

\[
\frac{T \sqrt{2\pi \alpha}}{\sum_{k=1}^{T} \exp(-p_k k^{2q/r})} \exp \left( -p_k k^{2q/r} \right) = 1, \ k = 1, \ldots, T,
\]

and these hold for \( p_k = 0, \ k = 1, \ldots, T \). Substitution into (24) gives (22), which is independent of \( q \) and hence a lower bound on \( \min_{p > 0, q > 1} S_W(p, q) \). The proof that (24) is a lower bound on \( \min_{p > 0} S_M(p) \) is similar and hence omitted. ■

When employing the sample size formula (5), the associated function \( S_M(p) \) is convex, and associated minimizers \( p^* \) for various values of \( T \) are shown in the second and third columns of Table 1. When we use sample size formula (16) we seek to select \( p \) and \( q \) to solve \( \min_{p > 0, q > 1} S_W(p, q) \). By Proposition 8, \( S_W(\cdot, q) \) is convex and so we can minimize this function for a fixed value of \( q \). The fourth and fifth columns of Table 1 show the results of doing so for \( q = 1.5 \). We know by the lower-bounding values shown in the final column of the table that these are suboptimal by no more than 2.5-5.5%.

The results of Table 1 can guide selection of \( p \) given rough estimates for \( T \). Of course, the assumption that \( T \) is known is unrealistic. While we could view \( T \) as a random variable and attempt to minimize \( S_M \) or \( S_W \) in expectation, we will not do so. When the procedure terminates at a different iteration than that of the assumed \( T \), the differences in sample sizes are quite modest. For instance, taking \( h - h' = 0.5 \) and \( \alpha = 0.10 \), and using \( p^1 = 1.91 \times 10^{-1} \) (for \( T = 50 \)), we have \( n_T \geq 33, 56, \) and 65 samples when \( T = 1, 50 \) and 100 respectively under the MGF assumption of (7). Similarly, we have \( n_T \geq 37, 55, \) and 63 when we instead use \( p^2 = 1.53 \times 10^{-1} \) (for \( T = 100 \)).
For the weaker moment condition with \( r = 2 \) and \( q = 1.5 \), \( n_T \geq 39, 52, \) and 77 when \( T = 1, 50 \) and 100 respectively when we use \( p^1 = 4.67 \times 10^{-3} \) and \( n_T \geq 45, 50, \) and 58 samples when we instead use \( p^2 = 1.66 \times 10^{-3} \). Slightly smaller values of \( S_W(p,q) \) (and \( n_T \)) can be obtained with even larger values of \( q \) but the results are more sensitive to having assumed the “wrong” value of \( T \).

6 Application to Two-Stage Stochastic Linear Programs

In this section, we apply the sequential sampling procedure to two-stage stochastic linear programs with recourse. We present computational results on two test problems from the literature, PGP2 and APL1P. PGP2 is an electric power generation model with 3 stochastic parameters and 576 scenarios (Higle and Sen 1996b). APL1P, another power generation model, has 5 independent stochastic parameters and 1280 scenarios (Infanger 1992). These small problems can be solved exactly, and hence allow us to assess the performance of our sequential sampling procedures. In addition, we use these as test problems since we know they pose challenges for the optimality gap estimators we use. We discuss this issue further below; see also Bayraksan and Morton (2006). We first describe the different gap and variance estimators we use in the sequential sampling procedure. Then, we discuss how we generate the candidate solutions in our computational experiments. Finally, we examine the effect of resampling, and present computational results for these two test problems.

Gap and Variance Estimators

We use four different methods to form the gap and variance estimators to assess the quality of candidate solutions. These are the single replication procedure (SRP), the averaged two-replication procedure (A2RP), and their \( \varepsilon \)-optimal versions (\( \varepsilon \)-optimal SRP, and \( \varepsilon \)-optimal A2RP), which we now detail. For the SRP estimators, step 1 of the sequential procedure presented at the end of §3 computes gap and variance estimators according to equations (2), i.e., step 1 becomes:

<table>
<thead>
<tr>
<th>( T )</th>
<th>( p^* )</th>
<th>( S_M(p^*) )</th>
<th>( p^* )</th>
<th>( S_W(p^*, 1.5) )</th>
<th>( LB )</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>( 4.07 \times 10^{-1} )</td>
<td>82</td>
<td>( 5.05 \times 10^{-2} )</td>
<td>78</td>
<td>74</td>
</tr>
<tr>
<td>50</td>
<td>( 1.91 \times 10^{-1} )</td>
<td>591</td>
<td>( 4.67 \times 10^{-3} )</td>
<td>552</td>
<td>530</td>
</tr>
<tr>
<td>100</td>
<td>( 1.53 \times 10^{-1} )</td>
<td>1,334</td>
<td>( 1.66 \times 10^{-3} )</td>
<td>1,243</td>
<td>1,198</td>
</tr>
<tr>
<td>500</td>
<td>( 1.04 \times 10^{-1} )</td>
<td>8,421</td>
<td>( 1.49 \times 10^{-4} )</td>
<td>7,822</td>
<td>7,598</td>
</tr>
<tr>
<td>1,000</td>
<td>( 9.08 \times 10^{-2} )</td>
<td>18,333</td>
<td>( 5.27 \times 10^{-5} )</td>
<td>17,031</td>
<td>16,583</td>
</tr>
</tbody>
</table>

Table 1: Choices of \( p \) that minimize \( S_M(p) \) and \( S_W(p,q) \) for a given number of iterations, \( T \). In \( S_W(p,q) \), we fix \( q = 1.5 \) and \( r = 2 \). \( LB = 2T \ln(T/(\sqrt{2\pi} \alpha)) \) denotes the lower bound given in Proposition 8. The values for \( S_M \) and \( S_W \) (and \( LB \)) are reported are for \( \alpha = 0.10 \), but the same \( p^* \) values are optimal, e.g., for \( \alpha = 0.05 \).
1.a. Solve (SP_{n_k}) using \( \hat{\xi}^1, \hat{\xi}^2, \ldots, \hat{\xi}^{n_k} \) i.i.d. as \( \hat{\xi} \) to obtain \( x^*_{n_k} \),

1.b. Calculate \( G_k = \frac{1}{n_k} \sum_{i=1}^{n_k} \left( f(\hat{x}_k, \hat{\xi}) - f(x^*_k, \hat{\xi}^i) \right), \) and

\[
s^2_k = \frac{1}{n_k - 1} \sum_{i=1}^{n_k} \left( (f(\hat{x}_k, \hat{\xi}) - f(x^*_k, \hat{\xi}^i)) - (\bar{f}_{n_k}(\hat{x}_k) - \bar{f}_{n_k}(x^*_k)) \right)^2.
\]

The implementation of the sequential procedure with A2RP involves small changes. We select \( n_k \) even and divide the observations into two random partitions and calculate \( G^i_k \) and \( s^2_k \) as above (steps 1.a and 1.b) for each sample containing \( n_k/2 \) observations, \( i = 1, 2 \). We then pool these gap and variance estimators to obtain \( G_k = \frac{1}{2}(G^1_k + G^2_k) \) and \( s_k^2 = \frac{1}{2}(s_k^{11} + s_k^{22}) \). For the sequential procedure that uses A2RP, we use these pooled estimators for the stopping criterion in step 3 of the procedure. Note that the gap and variance estimators formed by SRP and A2RP under i.i.d. sampling satisfy assumptions A2-A5 stated in §2 (Bayraksan and Morton 2006).

In our computational tests, we also use \( \varepsilon \)-optimal versions of these methods in which we solve the sampling problem(s) in step 1.a suboptimally. Our motivation for doing so is two-fold. First, it is unnecessary (and computationally wasteful) to numerically optimize a sample mean orders of magnitude more precisely than the confidence interval width output by our procedure. Second, we found in Bayraksan and Morton (2006) that numerically precise optimization of the (SP) used in assessing solution quality can degrade coverage properties, even in the non-sequential setting due to an \( x^*_n \) coinciding with the candidate solution. This property also motivated development of the A2RP estimators described above. An \( \varepsilon \)-optimal version of SRP works as follows: The \( G_k \) calculated in step 1.b above can be expressed \( G_k = \bar{f}_{n_k}(\hat{x}_k) - \bar{z}_{n_k}^* \). When solving (SP_{n_k}) to obtain \( z_{n_k}^* \), we have an algorithm which iteratively refines bounds \( \underline{z}_{n_k}^* \leq z_{n_k}^* \leq \overline{z}_{n_k}^* \). In the regular SRP, the sampling problem in step 1.a is solved with sufficient precision, so that the algorithm effectively terminates with \( \underline{z}_{n_k}^* = z_{n_k}^* = \tau_{n_k}^* \). In contrast, in \( \varepsilon \)-optimal SRP, the sampling problem in 1.a is solved suboptimally, i.e., the algorithm is stopped when \( (\tau_{n_k}^* - \underline{z}_{n_k}^*) / \min\{|\tau_{n_k}^*|, |\underline{z}_{n_k}^*|\} \) drops below \( \varepsilon \). The lower bound, \( \underline{z}_{n_k}^* \), is used for constructing the gap and the sampling variance estimators. When implementing the \( \varepsilon \)-optimal procedures, we set the suboptimality level to \( \varepsilon = 10^{-3} \), which was found to yield good coverage results in the non-sequential setting (Bayraksan and Morton 2006).

**Generating Candidate Solutions**

The method that generates candidate solutions is an input to the sequential procedure. Any method suffices provided its sequence of candidate solutions has at least one limit point in the set of optimal solutions, \( X^* \). In our computational results, we generate the candidate solutions \( \{\hat{x}_k\} \) by solving a separate sampling problem \( (SP_{m_k}) \) with increasing sample sizes \( m_k \) at iteration \( k \). Briefly, we state it below.
Table 2: Parameters and the corresponding initial sample sizes used in the tests. Other parameters are the same for both test problems: $\alpha = 0.10$, $\epsilon = 2 \times 10^{-7}$, $\epsilon' = 1 \times 10^{-7}$, and $p = 1.91 \times 10^{-1}$.

\[
\begin{array}{|c|c|c|}
\hline
 & PGP2 & APL1P \\
\hline
h' & 0.025 & 0.015 \\
h & 0.312 & 0.217 \\
n_1 & 100 & 200 \\
\hline
\end{array}
\]

\[i.\] Set $m_k = m_1$. Sample i.i.d. observations (independent of those used in evaluation procedures) $\tilde{\xi}_1, \tilde{\xi}_2, \ldots, \tilde{\xi}_{m_k}$ from the distribution of $\tilde{\xi}$.

\[ii.\] Solve (SP$_{m_k}$) using observations generated so far to obtain $x^*_{m_k}$.

\[iii.\] Set $\hat{x}_k = x^*_{m_k}$. Calculate $m_{k+1}$, sample $m_{k+1} - m_k$ i.i.d. observations $\tilde{\xi}_{m_k+1}, \tilde{\xi}_{m_k+2}, \ldots, \tilde{\xi}_{m_{k+1}}$ from the distribution of $\tilde{\xi}$. Set $k = k + 1$ and go to $ii$.

Note that in step $i$ above, we use a separate stream of i.i.d. observations from the distribution of $\tilde{\xi}$, independent from the ones generated in steps 0 and 2 of the sequential procedure. For our computational experiments, we set $m_k = 2n_k$. We are willing to spend more computational effort to find high-quality candidate solutions and use fewer sample sizes to evaluate them. All limit points of this sequence of candidate solutions is in the set of optimal solutions, $X^*$, under mild conditions satisfied by our test problems; see, e.g., Rubinstein and Shapiro (1993). So, assumption A1 is satisfied.

**Parameters Used**

Both of the test problems satisfy assumptions stated in §2 ($X \neq \emptyset$ and is compact, $f(\cdot; \tilde{\xi})$ is continuous on $X$, w.p.1., etc.) Moreover, the random vector $\tilde{\xi}$ of these test problems each has discrete distributions with independent components with bounded support. Therefore, the MGF assumption (7) is satisfied for all $\gamma_0$ and hence we use the sample size formula (5). We set $\alpha = 0.10$ and design the procedure for $T = 50$. Minimizing the computational effort, we set $p = 1.91 \times 10^{-1}$ (see §5). Table 2 lists the values of $h'$ and $h$ used for the two test problems PGP2 and APL1P. With the given parameters, the sequential procedure uses $n_1 \geq 100$ for PGP2 and $n_1 \geq 200$ for APL1P.

When implementing the procedure, we set $\epsilon = 2 \times 10^{-7}$ and $\epsilon' = 1 \times 10^{-7}$. Here, $\epsilon'$, in addition to ensuring finite stopping, serves to deal with nonzero numerical tolerances. For instance, suppose we are using a solver with a tolerance of $1 \times 10^{-8}$ and at an iteration $k$, we calculate $G_k = 1 \times 10^{-8}$ and $s_k = 1 \times 10^{-12}$, which we can essentially treat as 0. However, if we have $\epsilon' = 0$ in (3), we would not stop for $h < 10,000$. Note that $\epsilon' = 1 \times 10^{-7}$ used in the stopping criterion (3) is small enough not to interfere with stopping of the sequential procedure when $G_k$ and $s_k^2$ are sufficiently large. We are now ready to review the results of our tests.
Table 3: Effect of resampling frequency, $k_f$. Results are from applying the sequential procedure with SRP to PGP2. We report total solution time (in seconds), total time spent in assessing solution quality (in seconds), total number of sampling problems, $(SP_n)_s$, solved for assessing solution quality, and an average time spent per sampling problem for assessing solution quality. We also report average value of $T$, the iteration the sequential procedure stopped along with a 90% confidence interval half-width for $ET$.

<table>
<thead>
<tr>
<th>$k_f$</th>
<th>Sol Time</th>
<th>Assess Time</th>
<th>$(SP_n)_s$ Solved</th>
<th>Assess Time/$(SP_n)$</th>
<th>$T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>307.97</td>
<td>102.91</td>
<td>2152</td>
<td>0.05</td>
<td>54.14 ± 16.44</td>
</tr>
<tr>
<td>75</td>
<td>279.56</td>
<td>93.51</td>
<td>1950</td>
<td>0.05</td>
<td>41.46 ± 12.78</td>
</tr>
<tr>
<td>50</td>
<td>243.99</td>
<td>82.28</td>
<td>1728</td>
<td>0.05</td>
<td>30.50 ± 8.96</td>
</tr>
<tr>
<td>25</td>
<td>196.42</td>
<td>67.08</td>
<td>1339</td>
<td>0.05</td>
<td>16.27 ± 2.71</td>
</tr>
<tr>
<td>12</td>
<td>162.62</td>
<td>61.27</td>
<td>1007</td>
<td>0.06</td>
<td>11.07 ± 1.72</td>
</tr>
<tr>
<td>6</td>
<td>151.95</td>
<td>66.55</td>
<td>808</td>
<td>0.08</td>
<td>8.65 ± 1.40</td>
</tr>
<tr>
<td>3</td>
<td>133.70</td>
<td>64.40</td>
<td>606</td>
<td>0.11</td>
<td>6.13 ± 0.79</td>
</tr>
<tr>
<td>1</td>
<td>135.10</td>
<td>77.35</td>
<td>417</td>
<td>0.19</td>
<td>4.19 ± 0.59</td>
</tr>
</tbody>
</table>

Computational Results

For our computational results, we use the regularized decomposition algorithm of Ruszczyński (1986). An accelerated implementation of this algorithm, in C++, is due to Ruszczyński and Świetanowski (1997). We modified this code to warm-start the algorithm when an additional sample or a number of samples is added to the current problem, allowing for faster solution times in sequential sampling. This helps us to quickly find a solution to a sampling problem when we augment it with a few additional samples. However, as mentioned before, our method allows for augmenting the samples as well as creating an entirely new set of samples. Augmenting is computationally attractive but we may get trapped, for an extended number of iterations, in a bad sample path. Resampling can help the procedure to move to a different sample path but the computational effort increases. First, we examine this trade-off between augmentation and resampling by changing the resampling frequency, $k_f$. Then, we examine the performance of the four different methods for assessing solution quality.

In Table 3 we report the results of our tests where we varied the resampling frequency, $k_f$, from 100 down to 1 for the sequential sampling procedure using the SRP estimators on PGP2. Each row of the table reports averages of the following, over 100 independent runs of the procedure: the total solution time, which includes both the time for generating the sequence of candidate solutions and the time for assessing their quality; the time spent assessing solution quality; the number of $(SP_n)_s$ solved in assessing solution quality; the time spent for assessing solution quality per sampling problem $(SP_n)$; and, the average number of iterations of the sequential procedure (along with a 90% confidence interval halfwidth). We only examine the resampling frequency issue in the context of
assessing solution quality, i.e., we always augment the sampling problem (SP$_{m_k}$) used for generating the candidate solutions. However, as described in step 3 of the sequential procedure (§3), when the iteration number, $k$, is an integer multiple of $k_f$, we discard the set of observations used for computing gap and variance estimators and we generate a new set of samples, $\tilde{\xi}_1, \tilde{\xi}_2, \ldots, \tilde{\xi}_{n_k}$. Even though we report the total solution times, we focus on the computational effort for assessing solution quality since our approach allows for generating the sequence of candidate solutions in other ways. In Table 3, as $k_f$ decreases, $T$, the iteration on which the sequential procedure stops and the total solution time both tend to decrease. However, the time spent in assessing solution quality first decreases, then, starts to increase. This is because as we resample with a higher frequency (i.e., as $k_f$ decreases), we lose the computational advantage of warm starts. For instance, with $k_f = 25$, the total time spent in assessing solution quality during the 100 runs is 67.08 seconds and a total of 1339 sampling problems are solved for this purpose. The average time spent per sampling problem is 0.05 seconds in this case. However, when $k_f = 1$, i.e., when we resample at every iteration, we solve a total of only 417 sampling problems but with a larger amount of time, 77.35 seconds. The average time spent per sampling problem grows to 0.19 seconds. When we look at the time spent per sampling problem, the computational advantage of augmentation is in effect until $k_f = 25$, and as $k_f$ further decreases, resampling increases the solution time per sampling problem. We did the same analysis for APL1P and found similar results for $k_f = 12$. For the rest of our computational experiments, we set $k_f = 25$ for PGP2 and $k_f = 12$ for APL1P.

We now turn to coverage properties of our sequential procedure, and examine these using four different estimators, namely, SRP, A2RP and their $\varepsilon$-optimal versions. In Bayraksan and Morton (2006), we recommended the $\varepsilon$-optimal version of A2RP as a computationally-attractive alternative to the multiple-replication procedure of Mak, Morton, and Wood (1999). Both of those papers focused on non-sequential procedures involving a single candidate solution, $\hat{x}$. Below, we examine their performance within a sequential sampling procedure. Tables 4 and 5 provide a summary of results for PGP2 and APL1P, respectively. As above, the results are based on 100 independent runs of the sequential procedures. To reduce the effect of sampling when comparing the results, we use the same stream of random numbers to feed each of the four estimation procedures. Also, since A2RP and its $\varepsilon$-optimal version use sample sizes that are even, we round up the sample sizes for SRP and its $\varepsilon$-optimal version so they are even as well. The tables report the number of iterations ($T$), the confidence interval width (CI), and the fraction of the 100 confidence intervals that contained the true optimality gap ($\hat{p}$). For each one of these values, we report the average of the 100 runs, along with their 90% confidence interval half-widths. For instance, the empirical coverage probabilities, $\hat{p}$, estimate the probability that the CI produced at the end of the sequential procedure, $[0, \ h_{sT} + \epsilon]$, contains the optimality gap of the candidate solution at the stopping
<table>
<thead>
<tr>
<th>Method</th>
<th>$T$</th>
<th>CI</th>
<th>$\hat{p}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRP</td>
<td>16.27 ± 2.71</td>
<td>10.49 ± 2.73</td>
<td>0.79 ± 0.07</td>
</tr>
<tr>
<td>$\varepsilon$-Opt SRP</td>
<td>14.97 ± 2.20</td>
<td>39.50 ± 17.37</td>
<td>0.96 ± 0.03</td>
</tr>
<tr>
<td>A2RP</td>
<td>107.28 ± 14.48</td>
<td>5.88 ± 2.13</td>
<td>0.76 ± 0.07</td>
</tr>
<tr>
<td>$\varepsilon$-Opt A2RP</td>
<td>103.29 ± 13.79</td>
<td>6.90 ± 2.02</td>
<td>0.98 ± 0.02</td>
</tr>
</tbody>
</table>

Table 4: Summary of results for PGP2 ($k_f = 25$; $z^* = 447.324$ for PGP2).

<table>
<thead>
<tr>
<th>Method</th>
<th>$T$</th>
<th>CI</th>
<th>$\hat{p}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRP</td>
<td>15.40 ± 2.89</td>
<td>52.77 ± 8.54</td>
<td>0.88 ± 0.05</td>
</tr>
<tr>
<td>$\varepsilon$-Opt SRP</td>
<td>15.72 ± 2.90</td>
<td>54.98 ± 9.54</td>
<td>0.89 ± 0.05</td>
</tr>
<tr>
<td>A2RP</td>
<td>56.18 ± 13.05</td>
<td>66.10 ± 5.82</td>
<td>0.99 ± 0.02</td>
</tr>
<tr>
<td>$\varepsilon$-Opt A2RP</td>
<td>59.89 ± 13.30</td>
<td>73.24 ± 9.68</td>
<td>0.99 ± 0.02</td>
</tr>
</tbody>
</table>

Table 5: Summary of results for APL1P ($k_f = 12$; $z^* = 24,642.32$ for APL1P).

iteration, $\mu_T$. In other words, $\hat{p}$ estimates the probability in (4). A 90% confidence interval on the actual coverage probability, $P(\mu_T \leq h s_T + \epsilon)$, is given by $\hat{p} \pm 1.645 \sqrt{\hat{p}(1-\hat{p})/100}$, which is reported in the tables.

The results of Tables 4 and 5 indicate that the A2RP method, on average, takes more iterations to terminate ($T$ is larger), and the $\varepsilon$-optimal versions yield larger CIs and have more conservative coverage probabilities ($\hat{p}$ is not below 0.90). The sequential procedure appears to work well for APL1P. The CI widths are within 0.5% of optimality for all methods and coverage probabilities are around or higher than 0.90. The results for PGP2 are more variable among the four methods. The variance of some of the frequently-obtained solutions is quite large for PGP2 and this results in larger CIs. A2RP and its $\varepsilon$-optimal version decrease the variance but the sequential procedure runs for a longer amount of time. We note that one might try to control the sampling variance by adding the condition $\{s_k \leq b\}$ to the stopping criterion. However, in our computational results, we have seen that this sometimes results in very long runs, with $T \geq 1,000$. For PGP2, the CIs obtained with $\varepsilon$-optimal A2RP, on average, are within 1.6% of optimality with high coverage probabilities. Overall, we view the results as consistent with our findings in the non-sequential setting, i.e., we view $\varepsilon$-optimal version of A2RP as preferable for these two test problems: Its coverage results are conservative, i.e., the procedure does not exhibit the risk of undercoverage that can arise from estimators rooted in the SRP and the numerically-precise A2RP.

7 Conclusions

In this paper, we develop a sequential sampling procedure to solve stochastic programs. We assume that a sequence of candidate solutions with at least one limit point that solve (SP) is given
as an input to the procedure. Then, the sequential sampling procedure assesses the quality of these candidate solutions with increasing sample size and terminates according to a stopping criterion. The stopping criterion depends on the optimality gap estimate of the current solution and its associated variance. If the stopping criterion is satisfied, then the procedure outputs a confidence interval on the optimality gap of the current candidate solution. If not, the sample size is increased. The sequential procedure we develop allows for augmentation of additional observations to the ones that were previously generated, or, generation of an entirely new set of observations. Warm-starting techniques favor the first method and prevention of being trapped in a bad sample path favors the second method.

We provide rules to stop and to increase the sample sizes and a statement regarding the quality of the solution obtained when the procedure terminates. We show that this procedure asymptotically finds a high-quality solution with a desired probability, under certain assumptions. We also show that the procedure terminates in a finite number of steps with probability 1. Then, we discuss several issues that arise in its implementation, such as determining what parameters to use. Our aim in choosing the parameters is to minimize the computational effort exerted while running the procedure. Finally, we apply the sequential sampling procedure to two two-stage stochastic linear programs with recourse from the literature. We test four different methods to assess a candidate solution’s quality that were previously developed. These are SRP, A2RP, and their $\varepsilon$-optimal versions. Our preliminary computational results indicate that the sequential sampling procedure with $\varepsilon$-optimal A2RP yields good coverage results with reasonable CI widths.

An area of future research is to develop more efficient sequential sampling procedures while maintaining the desired asymptotic properties. For instance, adaptive sequential methods can be designed where the sampling method takes into account the information obtained about the problem so far. Carefully designed adaptive methods can be more efficient while maintaining the desired asymptotic properties.

**Acknowledgements**

The authors thank Andrzej Ruszczyński and Artur Świetanowski for access to their regularized decomposition code. This research is partially supported by the National Science Foundation under Grant DMI-0217927.
References


