The Sample Average Approximation Method for Stochastic Programs with Integer Recourse

Shabbir Ahmed* and Alexander Shapiro†
School of Industrial & Systems Engineering
Georgia Institute of Technology
765 Ferst Drive, Atlanta, GA 30332

February 5, 2002

Abstract

This paper develops a solution strategy for two-stage stochastic programs with integer recourse. The proposed methodology relies on approximating the underlying stochastic program via sampling, and solving the approximate problem via a specialized optimization algorithm. We show that the proposed scheme will produce an optimal solution to the true problem with probability approaching one exponentially fast as the sample size is increased. For fixed sample size, we describe statistical and deterministic bounding techniques to validate the quality of a candidate optimal solution. Preliminary computational experience with the method is reported.

Keywords: Stochastic programming, integer recourse, sample average approximation, branch and bound.

1 Introduction

In the two-stage stochastic programming approach for optimization under uncertainty, the decision variables are partitioned into two sets. The *first stage* variables are those that have to be decided before the actual realization of the uncertain parameters becomes available. Subsequently, once the random events have presented themselves, further design or operational policy improvements can be made by selecting, at a certain cost, the values of the *second stage* or *recourse* variables. The objective is to choose the first stage variables in a way that the sum of first stage costs and the expected value of the random second stage or recourse costs is minimized.

*Supported in part by the National Science Foundation under Grant No. DMII-0099726.
†Supported in part by the National Science Foundation under Grant No. DMS-0073770.

A standard formulation of the two-stage stochastic program is [5, 12]:

$$\operatorname*{Min}_{x \in X} \left\{ g(x) := c^T x + \mathbb{E}[Q(x, \xi(\omega))] \right\}, \tag{1}$$

where

$$Q(x, \xi) := \inf_{y \in Y} \left\{ q^T y : Wy \geq h - Tx \right\} \tag{2}$$

is the optimal value and $\xi := (q, T, W, h)$ denotes vector of parameters of the second stage problem. It is assumed that some (or all) of the components of ξ are random, written $\xi(\omega)$, and the expectation in (1) is taken with respect to the probability distribution of $\xi(\omega)$ which is supposed to be known. Problem (1), with variables $x \in \mathbb{R}^{n_1}$, constitute the first stage which needs to be decided prior to a realization of $\xi(\omega)$, and problem (2), with variables $y \in \mathbb{R}^{n_2}$, constitute the recourse for given first stage decision and realization ξ of the random data.

This paper is concerned with two-stage stochastic programs where the recourse is fixed, i.e., the matrix W is constant (not random), and the recourse variables are restricted to be integers, i.e., $Y \subseteq \mathbb{Z}^{n_2}$ in (2). Such problems find applications in production planning [10, 17, 6], scheduling [9, 33, 4], routing [31, 14, 15], location [16], capacity expansion [3], electricity production [32, 8], environmental control [21], and finance [11].

The two key sources of difficulty in solving stochastic programs with integer recourse are:

1. *Exact evaluation of the expected recourse costs.* For a given first-stage decision and a realization of the random data, the recourse costs are computed via solving an integer program. Thus, for continuous distribution of the uncertain parameters, the exact evaluation of the expected recourse costs would require the multi-dimensional integration of integer programming value functions, which is practically impossible. Even for discrete distributions, an exact computation of the expectation would require solving the integer recourse problem for all possible realizations of the uncertain parameters and may be computationally prohibitive.

2. *Optimizing the expected recourse costs.* Even if the expected recourse cost function could be evaluated or approximated easily, the stochastic programming problem (1) involves optimizing this function over the first-stage decisions. It is well known that value functions of integer programs are highly non-convex and discontinuous. Consequently this optimization problem poses severe computational difficulties.

In this paper, we propose a solution strategy for a class of two-stage stochastic programs with integer recourse that addresses the above difficulties. In the proposed approach, the expected recourse cost function in (1) is replaced by a sample average approximation, and the corresponding optimization problem is solved using a specialized algorithm for non-convex optimization. It has been shown in [27], that a solution to this sample average approximation (SAA) problem converges to a solution of the true problem as the sample size tends to infinity. Here, we analyze the rate of convergence of the SAA solutions for stochastic programs with integer recourse. We also describe statistical and deterministic bounding techniques to estimate the near optimality of a candidate solution. Finally, some preliminary computational results are presented.

2 The Sample Average Approximation Method

The main idea of Sample Average Approximation (SAA) approach to solving stochastic programs is as follows. A sample $\xi^1, ..., \xi^N$ of N realizations of the random vector $\xi(\omega)$ is generated, and consequently the expected value function $\mathbb{E}[Q(x, \xi(\omega))]$ is approximated (estimated) by the sample average function $N^{-1} \sum_{n=1}^{N} Q(x, \xi^n)$. The obtained sample average approximation

$$\operatorname*{Min}_{x \in X} \left\{ \widehat{g}_N(x) := c^T x + N^{-1} \sum_{n=1}^{N} Q(x, \xi^n) \right\}, \tag{3}$$

of the stochastic program (1) is then solved by a deterministic optimization algorithm. This approach (and its variants) is also known under various names, such as the stochastic counterpart method [25] and sample path optimization method [24], for example.

Let us denote by $\widehat{v}_N$ and $\widehat{x}_N$ the optimal value and an optimal solution of the SAA problem (3), respectively; and by v^* and x^* the optimal value and an optimal solution of the true problem (1), respectively. The crucial issues to address are: (i) Whether $\widehat{v}_N$ and $\widehat{x}_N$ converges to their true counterparts v^* and x^* as the sample size N is increased? (ii) If so, can we analyze the rate of convergence, and thereby estimate the required sample size to obtain a true optimal solution with certain confidence? (iii) Is there an efficient optimization approach for solving the SAA problem for the required sample size? (iv) Note that for a given N the solution $\widehat{x}_N$ is feasible and is a candidate for an optimal solution to the true problem. Can we provide any information regarding the quality of this candidate solution?

The above questions have been answered quite satisfactorily for two-stage stochastic *linear* programs, i.e., when the first and second stage variables in (1) and (2) are continuous. It has been proved that for stochastic linear programs with discrete distributions, an optimal solution of the SAA problem provides an exact optimal solution of the true problem with probability approaching one exponentially fast as N increases [30]. Statistical tests for validating a candidate solution based on optimality gaps [22, 19] as well as optimality conditions [29] have also been proposed. Furthermore, these sampling techniques have been integrated with decomposition algorithms to successfully solve stochastic linear programs of enormous sizes to great precision [18]. Recently, the convergence of the SAA approach have also been extended to stochastic programs where the set of first-stage decisions is discrete and finite [13]. Encouraging computational results using the SAA method to solve certain classes stochastic programs with finite first-stage decisions have also been reported [34]. In the following section, we extend these results to two-stage stochastic programs with integer recourse and where the space of feasible first-stage decisions is infinite.

3 Convergence Analysis

We discuss in this section some convergence properties of SAA estimators, in particular applied to two-stage programming with integer recourse. In order to apply classical results, such as the Law of Large Numbers, it will be convenient to assume here that the generated sample is iid (independent identically distributed). Note, however, that basic convergence properties can be derived under much broader conditions. This is relevant in connection with various variance reduction techniques.

3.1 Discrete First Stage

We begin by briefly reviewing results in [13] on the convergence of the SAA method when applied to stochastic programs with a finite set of first-stage decisions.

Let us consider instances of two-stage stochastic programs (1) with the following characteristics:

(i) The set of first-stage decisions X is finite (but maybe very large).

(ii) The recourse function $Q(x,\cdot)$ is measurable and $\mathbb{E}\big|Q(x,\xi(\omega))\big|$ is finite for every $x \in X$.

Recall that v^* and $\widehat{v}_N$ denote the optimal values of the true problem and the SAA problem, respectively. Furthermore, for $\varepsilon \geq 0$, let X^ε and $\widehat{X}_N^\varepsilon$ denote the sets of ε-optimal solutions of the true and SAA problems, respectively. In particular, for $\varepsilon = 0$ these sets become the sets X^* and $\widehat{X}_N$ of optimal solutions of the respective problems.

It is possible to show that, under the above assumptions (i) and (ii), $\widehat{v}_N$ is a consistent estimator of v^*, i.e., $\widehat{v}_N$ converges with probability one (w.p.1) to v^* as $N \to \infty$. Moreover, by using theory of Large Deviations it is shown in [13] that for any $\varepsilon \geq 0$ and $\delta \in [0,\varepsilon]$ there exists a constant $\gamma(\delta,\varepsilon) \geq 0$ such that

$$1 - \mathbb{P}\left(\widehat{X}_N^\delta \subset X^\varepsilon\right) \leq |X| e^{-N\gamma(\delta,\varepsilon)}. \tag{4}$$

Furthermore, under a mild additional assumption (which always holds in cases where the distribution of $\xi(\omega)$ has a finite support), the constant $\gamma(\delta,\varepsilon)$ is positive, and for small δ and ε can be estimated as

$$\gamma(\delta,\varepsilon) \geq \frac{(\varepsilon^* - \delta)^2}{3\sigma^2} \geq \frac{(\varepsilon - \delta)^2}{3\sigma^2}, \tag{5}$$

where $\varepsilon^* := \min_{x \in X \setminus X^\varepsilon} g(x) - v^*$, and σ^2 is the maximal variance of certain differences between values of the objective function of the SAA problem.

Note that, since here the set X is finite, ε^* is always greater than ε, and hence $\gamma(\delta, \varepsilon)$ is positive even if $\delta = \varepsilon$. In particular for $\delta = \varepsilon = 0$, inequality (4) gives an exponential rate of convergence of the probability of the event $\{\widehat{x}_N \in X^*\}$ to one, for any choice of optimal solution $\widehat{x}_N$ of the SAA problem. Note, however, that for large (although finite) feasible sets X the difference $\varepsilon^* - \varepsilon$ tends to be small. Nevertheless, the right hand side estimate of (5) leads to the following estimate of the sample size $N = N(\varepsilon, \delta, \alpha)$ which is required to solve the "true" (expected value) problem with given probability $1-\alpha$ and given precision $\varepsilon > 0$ by solving the SAA problem with precision $\delta \in [0, \varepsilon)$:

$$N \geq \frac{3\sigma^2}{(\varepsilon - \delta)^2} \log\left(\frac{|X|}{\alpha}\right). \tag{6}$$

Although the above bound usually is too conservative to use for a practical calculation of the required sample size, it shows logarithmic dependence of $N(\varepsilon, \delta, \alpha)$ on the size $|X|$ of the first stage problem.

Note that the above results do not make any assumptions regarding the recourse variables. Thus, the above analysis is directly applicable to two-stage stochastic programs with integer recourse as long as the set of feasible first-stage decisions is finite.

3.2 Continuous First Stage

Convergence results of the previous section can be adjusted to deal with cases where some or all of the first-stage variables are allowed to take continuous values. Recall that any two norms on the finite dimensional space $\mathbb{R}^{n_1}$ are equivalent. For technical reasons it will be convenient to use in what follows the max-norm $\|x\| := \max\{|x_1|, ..., |x_{n_1}|\}$. Suppose that the set X is a bounded (not necessarily finite) subset of $\mathbb{R}^{n_1}$. For a given $\nu > 0$, consider a finite subset X_ν of X such that for any $x \in X$ there is $x' \in X_\nu$ satisfying $\|x - x'\| \leq \nu$. If D is the diameter of the set X, then such set X_ν can be constructed with $|X_\nu| \leq (D/\nu)^{n_1}$. By reducing the feasible set X to its subset X_ν, as a consequence of (6) we obtain the following estimate of the sample size, required to solve the reduced problem with an accuracy $\varepsilon' > \delta$:

$$N \geq \frac{3\sigma^2}{(\varepsilon' - \delta)^2} \left(n_1 \log \frac{D}{\nu} - \log \alpha\right). \tag{7}$$

Suppose, further, that the expectation function $g(x)$ is Lipschitz continuous on X modulus L. Then an ε'-optimal solution of the reduced problem is an ε-optimal solution of problem (1) with $\varepsilon = \varepsilon' + L\nu$. Let us set $\nu := (\varepsilon - \delta)/(2L)$ and $\varepsilon' := \varepsilon - L\nu = \varepsilon - (\varepsilon - \delta)/2$. By employing (7) we obtain the following estimate of the sample size N required to solve the true problem (1):

$$N \geq \frac{12\sigma^2}{(\varepsilon - \delta)^2} \left(n_1 \log \frac{2DL}{\varepsilon - \delta} - \log \alpha\right). \tag{8}$$

The above estimate (8) shows that the sample size which is required to solve the true problem (1) with probability $1-\alpha$ and accuracy $\varepsilon > 0$ by solving the SAA problem with accuracy $\delta < \varepsilon$, grows linearly in dimension n_1 of the first stage problem.

The estimate (8) is directly applicable to two-stage stochastic programs with integer recourse provided that the expected value function $g(x)$ is Lipschitz continuous. It is shown in [26, Proposition 3.6] that, indeed, in the case of two-stage programming with integer recourse the expected value function $g(x)$ is Lipschitz continuous on a compact set if the random data vector $\xi(\omega)$ has a continuous distribution and some mild additional conditions are satisfied. On the other hand, if $\xi(\omega)$ has a discrete distribution, then $g(x)$ is not even continuous. We discuss that case in the following section.

3.3 Continuous First Stage and Discrete Distribution

In this section we discuss two-stage stochastic programs with integer recourse when the random data vector has a discrete distribution with a finite support. We show that in such a setting the true and the SAA problems can be equivalently formulated as problems over a finite feasible region. Throughout the remainder of this paper we make the following assumptions.

(A1) The distribution of the random data vector $\xi(\omega)$ has a finite support $\Xi = \{\xi_1, \dots, \xi_K\}$ with respective (positive) probabilities $p_1, \dots, p_K$. Each realization $\xi_k = (q_k, T_k, W, h_k)$, $k = 1, ..., K$, of $\xi(\omega)$ is called scenario.

Then the expected value $\mathbb{E}[Q(x, \xi(\omega)]$ is equal to $\sum_{k=1}^{K} p_k Q(x, \xi_k)$, and hence we can write the true problem (1) as follows

$$\operatorname*{Min}_{x \in X} \Big\{ g(x) := c^T x + \sum_{k=1}^{K} p_k Q(x, \xi_k) \Big\}. \tag{9}$$

Here

$$Q(x, \xi_k) := \inf_{y \in Y} \Big\{ q_k^T y : Wy \geq h_k - T_k x \Big\}, \tag{10}$$

with $X \subseteq \mathbb{R}^{n_1}$, $c \in \mathbb{R}^{n_1}$, $Y \subseteq \mathbb{R}^{n_2}$, $W \in \mathbb{R}^{m_2 \times n_1}$, and for all k, $q_k \in \mathbb{R}^{n_2}$, $T_k \in \mathbb{R}^{m_2 \times n_1}$, and $h_k \in \mathbb{R}^{m_2}$.

In many applications, the total number of scenarios K is astronomically large, making the exact evaluation of the sum $\sum_{k=1}^{K} p_k Q(x, \xi_k)$ impossible. This motivates the need for a sampling based approach for solving (9). We shall be concerned with instances of (9) where:

(A2) The first stage variables are continuous, and the constraint set X is nonempty, compact, and polyhedral.

(A3) The second stage variables y are purely integer, i.e., $Y \subset \mathbb{Z}^{n_2}$ in (2).

Note that the assumption of purely continuous first-stage variables is without loss of generality. Mixed-integer first stage variables can be handled in the framework to follow without any added conceptual difficulty. In addition to (A1)-(A3), we also assume:

(A4) $Q(x, \xi_k)$ is finite for all $x \in X$ and $k = 1, ..., K$.

(A5) The second stage constraint matrix is integral, i.e., $W \in \mathbb{Z}^{m_2 \times n_2}$.

Assumption (A4) means, of course, that $Q(x, \xi_k) < +\infty$ and $Q(x, \xi_k) > -\infty$ for all $x \in X$ and $k = 1, ..., K$. The first of these two inequalities is known as the *relatively complete recourse* property [35]. Since X is compact, relatively complete recourse can always be accomplished by adding penalty inducing artificial variables to the second stage problem. Assumption (A5) can be satisfied by appropriate scaling whenever the matrix elements are rational.

Under assumption (A3) we have that, for any $\xi \in \Xi$, $Q(\cdot, \xi)$ is the optimal value function of a pure integer program and is well known to be piecewise constant. That is, for any $z \in \mathbb{Z}^{m_2}$ and $\xi \in \Xi$ the function $Q(\cdot, \xi)$ is constant over the set [28]

$$\mathcal{C}(z, \xi) := \{x \in \mathbb{R}^{n_1} : h - z - 1 \le Tx < h - z\}, \tag{11}$$

where the notation " $\le$ " and " $<$ " is understood to be applied componentwise. It follows then that for any $z \in \mathbb{Z}^{m_2}$ the function $\sum_{k=1}^{K} p_k Q(\cdot, \xi_k)$ is constant over the set $\mathcal{C}(z) := \cap_{k=1}^{K} \mathcal{C}(z, \xi_k)$. Note that $\mathcal{C}(z)$ is a neither open nor closed polyhedral region. Now let

$$\mathcal{Z} := \{z \in \mathbb{Z}^{m_2} : \mathcal{C}(z) \cap X \neq \emptyset\}.$$

Since X is bounded by assumption (A2), the set $\mathcal{Z}$ is finite. We obtain that the set X can be represented as the union of a finite number of polyhedral sets $\mathcal{C}(z) \cap X$, $z \in \mathcal{Z}$. On every set $\mathcal{C}(z) \cap X$ the expected value function $g(x)$ is linear, and attains its optimal value at an extreme point (a vertex) of $\mathcal{C}(z) \cap X$. Let us define

$$V := \bigcup_{z \in \mathcal{Z}} \text{vert}\big(\mathcal{C}(z) \cap X\big), \tag{12}$$

where $\text{vert}(S)$ denotes the set of vertices of the polyhedral set S. Note that X is polyhedral by assumption (A2), and $\mathcal{C}(z)$ is polyhedral by definition, thus from the finiteness of $\mathcal{Z}$ it follows that the set V is finite. It has been shown [28], that under assumptions (A1)-(A5), problem (9) possesses an optimal solution x^* such that $x^* \in V$. By virtue of this result, we can then restate (9) as the following two-stage stochastic program with finite first-stage decisions:

$$\operatorname*{Min}_{x \in V} \Big\{ g(x) = c^T x + \sum_{k=1}^{K} p_k Q(x, \xi_k) \Big\}. \tag{13}$$

It was proposed in [28] to solve (13) by enumerating the finite set V.

Let us attempt to estimate $|V|$. From (11), it is clear that the set $\mathcal{Z}$ includes all the integer vectors z, such that for a fixed h^k, the j-th component of z, i.e. z_j takes on the values $\lfloor h_j^k - T_j x \rfloor$ for all $x \in X$. Denote $\mathcal{X} := \{\chi \in \mathbb{R}^{m_2} : \chi = Tx,\ x \in X\}$, and let D be the diameter of $\mathcal{X}$. Note that since X is compact, $\mathcal{X}$ is also compact and hence $D < \infty$. Then, for a fixed h^k, z_j can take on at most $D+1$ possible values. If we now consider all K scenarios, z_j can take on $K(D+1)$ possible values. Consequently, we can bound the cardinality of $\mathcal{Z}$ as $|\mathcal{Z}| \leq [K(D+1)]^{m_2}$. Now, consider, for any $z \in \mathcal{Z}$, the system

$$\begin{aligned} \mathrm{cl}(\mathcal{C}(z)) &= \{x \in \mathbb{R}^{n_1} \ :\ x \geq 0,\ Tx \leq h^k - z, Tx \geq h^k - z - 1,\ \forall\ k\} \\ &= \{x \in \mathbb{R}^{n_1} \ :\ x \geq 0,\ Tx \leq \underline{h} - z,\ Tx \geq \overline{h} - z - 1\}, \end{aligned}$$

where $\underline{h} = \min_k\{h^k\}$ and $\overline{h} = \max_k\{h^k\}$ and the max and min operations are component-wise. Assuming that $X = \{x \in \mathbb{R}^{n_1} : Ax \leq b,\ x \geq 0\}$, where A is an $m_1 \times n_1$ matrix, we have that for any $z \in \mathcal{Z}$,

$$\mathrm{cl}(\mathcal{C}(z)) \cap X = \{x \in \mathbb{R}^{n_1} \ :\ Ax \leq b,\ x \geq 0,\ Tx \leq \underline{h} - z,\ Tx \geq \overline{h} - z - 1\}.$$

The above system is defined by at most $n_1 + m_1 + 2m_2$ linear inequalities (including non-negativity), and thus has at most

$$\left(\begin{array}{c} n_1 + m_1 + 2m_2 \\ n_1 \end{array} \right) < (n_1 + m_1 + 2m_2)^{m_1 + 2m_2}$$

extreme points. We thus have the following upper bound on the cardinality of V,

$$|V| \leq [K(D+1)]^{m_2} \left(n_1 + m_1 + 2m_2\right)^{m_1 + 2m_2}. \tag{14}$$

Thus the cardinality of V, as well the effort required in evaluating a candidate solution $x \in V$, largely depends on K. We can avoid explicit consideration of all possible scenarios $\{\xi_1, \ldots, \xi_K\}$ in (13) by using the SAA approach.

Consider a sample $\{\xi^1, \ldots, \xi^N\}$ of the uncertain problem parameters, with the sample size N much smaller than K. Then the SAA problem corresponding to (13) is:

$$\operatorname*{Min}_{x \in V_N} \Big\{ \widehat{g}_N(x) = c^T x + \frac{1}{N} \sum_{n=1}^{N} Q(x, \xi^n) \Big\}, \tag{15}$$

where V_N is the sample counterpart of the set V. That is,

$$V_N := \cup_{z \in \mathcal{Z}_N} \mathrm{vert}\left(\mathcal{C}_N(z) \cap X\right), \tag{16}$$

with $\mathcal{C}_N(z) := \cap_{n=1}^N \mathcal{C}(z, \xi^n)$ and $\mathcal{Z}_N := \{z \in \mathbb{Z}^{m_2} : \mathcal{C}_N(z) \cap X \neq \emptyset\}$.

Note that the sets V and V_N in problems (13) and (15) are not the same – these sets are induced by the set Ξ and a sampled subset of Ξ, respectively. It is not immediately obvious whether a solution of (15) even belongs to the set of candidate optimal solutions V. Fortunately, since the sampled vectors form a subset of Ξ, it follows that $V_N \subset V$. Thus any solution to (15) is a candidate optimal solution to the true problem. We can now directly apply the exponential convergence results of [13] (see Section 3.1) to stochastic programs with integer recourse and continuous first-stage variables. In particular, inequality (4) becomes

$$1 - \mathbb{P}\left(\widehat{X}_N^\delta \subset X^\varepsilon\right) \leq |V| e^{-N\gamma(\delta,\varepsilon)}, \tag{17}$$

where, as before, $\widehat{X}_N^\delta$ is the set of δ-optimal solutions to the SAA problem, X^ε is the set of ε-optimal solutions to the true problem, and the constant $\gamma(\delta, \varepsilon)$ can be estimated using (5). Using (17) and the estimate of $|V|$ in (14), we can now compute a sample size for the SAA problem to guarantee an ε-optimal solution to the true problem with probability $1 - \alpha$ as follows:

$$N \geq \frac{3\sigma^2}{(\varepsilon - \delta)^2} \left(m_2 \log[K(D+1)] + (m_1 + 2m_2) \log(n_1 + m_1 + 2m_2) - \log \alpha\right). \tag{18}$$

Although the above estimate of the sample size N is too conservative for practical purposes, it shows that N growths at most linearly with respect to the dimensions of the problem and logarithmically with the number of scenarios K. This indicates that one can obtain quite accurate solutions to problem involving a huge number of scenarios using a modest sample size.

Now consider a situation where the "true" distribution of $\xi(\omega)$ is continuous while a finite number of scenarios is obtained by a discretization of this continuous distribution. If such a discretization is sufficiently fine, then a difference between the corresponding expected values of the objective function is small. That is, let η be a constant bounding the absolute value of the difference between the expected values of $Q(x, \xi(\omega))$, taken with respect to the continuous and discrete distributions of $\xi(\omega)$, for all $x \in X$. It follows that if $\bar{x}$ is an ε-optimal solution of the expected value problem with respect to one of these distributions, then $\bar{x}$ is an $(\varepsilon + 2\eta)$-optimal solution of the expected value problem with respect to the other distribution. Therefore, if the expected value function, taken with respect to the continuous distribution, is Lipschitz continuous on X, then the estimate (8) can be applied to the problem with a discretized distribution adjusted for the corresponding constant η. We discuss this further in Section 6.

4 Solving the SAA problem

The enumeration scheme of Schultz et al. [28] can, in principle, be applied to solve the SAA problem (15). However, in general, it is very hard to a priori characterize the set V_N unless the second-stage problem has very simple structure. Furthermore, typically the cardinality of V_N is so large, that enumeration may be computationally impossible. Alternatively, we can attempt to solve the deterministic equivalent of (15) using a standard branch and bound algorithm such as those implemented in commercial integer programming solvers. However, such a scheme does not attempt to exploit the decomposable structure of the problem and is doomed to failure unless the sample sizes are very small.

We propose to use the decomposition based branch and bound (DBB) algorithm developed in [1, 2] to solve the SAA problem. Instead of a priori characterizing the candidate solution set V_N, this algorithm identifies candidate solutions by successively partitioning the search space. Furthermore, the algorithm makes use of lower bound information to eliminate parts of the search region and avoid complete enumeration. Since the algorithm does not explicitly search V_N, we have to make sure that the final solution produced does belong to this set to achieve the convergence behavior discussed in Section 3.3. In the following, we briefly describe the DBB algorithm and argue that it does return an optimal solution from the set V_N.

In addition to assumptions (A1)-(A5), the DBB algorithm assumes

(A6) The technology matrix T, linking the first and second stage problems, is deterministic, i.e., $T_k = T$ for all k.

Let us denote the linear transformation of the first-stage problem variables x using T by $\chi := Tx$. The variables χ are known as "Tender" variables in the stochastic programming literature. A principle idea behind the DBB algorithm is to consider the SAA problem with respect to the tender variables:

$$\min_{\chi \in \mathcal{X}} \left\{ \widehat{G}_N(\chi) := \Phi(\chi) + \widehat{\Psi}_N(\chi) \right\}, \tag{19}$$

where $\Phi(\chi) := \inf_{x \in X} \left\{ c^T x : Tx = \chi \right\}$, $\widehat{\Psi}_N(\chi) := N^{-1} \sum_{n=1}^{N} \Psi(\chi, \xi^j n)$,

$$\Psi(\chi, \xi) := \inf_{y \in Y} \left\{ q^T y : Wy \geq h - \chi \right\},$$

and $\mathcal{X} := \{\chi \in \mathbb{R}^{m_2} : \chi = Tx,\ x \in X\}$. Typically, $\mathcal{X}$ has smaller dimension than X. More importantly, this transformation induces a special structure to the discontinuous function $\widehat{\Psi}_N(\cdot)$. In particular, it can be shown [1, 2] that $\widehat{\Psi}_N : \mathbb{R}^{m_2} \to \mathbb{R}$ has the following properties:

(i) it is non-increasing along each component χ_j, $j = 1, \ldots, m_2$, of χ,

(ii) for any $z \in \mathbb{Z}^{m_2}$, it is constant over the set

$$C_N(z) := \{\chi : \quad h^n - z - 1 \leq \chi < h^n - z,\ n = 1, ..., N\}.$$

Note that the set $\mathcal{C}_N(z)$, used in the previous section, is related to the set $C_N(z)$ by the transformation $\chi = Tx$. Note also that the set $C_N(z)$ is a hyper-rectangle since it is the Cartesian product of intervals. Thus, the second stage expected value function $\widehat{\Psi}_N(\cdot)$ is piecewise constant over rectangular regions in the space of the tender variables χ. The discontinuities of $\widehat{\Psi}_N(\cdot)$ can then only lie at the boundaries of these regions and, therefore, are all orthogonal to the variable axes. Furthermore, since X is compact, so is $\mathcal{X}$, thus, the number of such regions within the feasible set of the problem is finite.

The DBB algorithm exploits the above structural properties by partitioning the space of χ into regions of the form $\Pi_{j=1}^{m_2}[l_j, u_j)$, where u_j is the j-th component of a point χ at which the second stage value function $\widehat{\Psi}_N(\cdot)$ may be discontinuous. Note that $\widehat{\Psi}_N(\cdot)$ can only be discontinuous at a point χ where at least one of the components of vector $h^n - \chi$ is integral for some $n = 1, ..., N$. Thus, we partition our search space along such values of χ. Branching in this manner, we can isolate regions over which the second stage value function is constant, and hence solve the problem exactly. A formal statement of the DBB algorithm follows [1, 2].

Initialization:

Preprocess the problem by constructing a hyper-rectangle of the form $\mathcal{P}^0 := \Pi_{j=1}^{m_1}[l_j^0, u_j^0)$ such that $\mathcal{X} \subset \mathcal{P}^0$. Add the problem

$$\text{Min}\, \widehat{G}_N(\chi) \ \text{ subject to } \ \chi \in \mathcal{X} \cap \mathcal{P}^0$$

to a list $\mathcal{L}$ of open subproblems. Set $U \leftarrow +\infty$ and the iteration counter $i \leftarrow 0$.

Iteration i:

Step i.1: If $\mathcal{L} = \emptyset$, terminate with solution χ^*, otherwise select a subproblem i, defined as

$$\text{Min}\, \widehat{G}_N(\chi) \ \text{ subject to } \ \chi \in \mathcal{X} \cap \mathcal{P}^i,$$

from the list $\mathcal{L}$ of currently open subproblems. Set $\mathcal{L} \leftarrow \mathcal{L} \setminus \{i\}$.

Step i.2: Obtain a lower bound β^i satisfying

$$\beta^i \leq \inf\{\widehat{G}_N(\chi) : \chi \in \mathcal{X} \cap \mathcal{P}^i\}.$$

If $\mathcal{X} \cap \mathcal{P}^i = \emptyset$, $\beta^i = +\infty$ by convention. Determine a feasible solution $\chi^i \in \mathcal{X}$ and compute an upper bound $\alpha^i \geq \min\{\widehat{G}_N(\chi) : \chi \in \mathcal{X}\}$ by setting $\alpha^i = \widehat{G}_N(\chi^i)$.

Step i.2.a: Set $L \leftarrow \min_{l \in \mathcal{L} \cup \{i\}} \beta^l$.

Step i.2.b: If $\alpha^i < U$, then $\chi^* \leftarrow \chi^i$ and $U \leftarrow \alpha^i$.

Step i.2.c: Fathom the subproblem list, *i.e.*, $\mathcal{L} \leftarrow \mathcal{L} \setminus \{l : \beta^l \geq U\}$. If $\beta^i \geq U$, then go to Step i.1 and select another subproblem.

Step i.3: Partition $\mathcal{P}^i$ into $\mathcal{P}^{i_1}$ and $\mathcal{P}^{i_2}$. Set $\mathcal{L} \leftarrow \mathcal{L} \cup \{i_1, i_2\}$, i.e., append the two subproblems

$$\text{Min}\, \widehat{G}_N(\chi) \ \text{ s.t. } \ \chi \in \mathcal{X} \cap \mathcal{P}^{i_1} \ \text{ and } \ \text{Min}\, \widehat{G}_N(\chi) \ \text{ s.t. } \ \chi \in \mathcal{X} \cap \mathcal{P}^{i_2}$$

to the list of open subproblems. For selection purposes, set $\beta^{i_1}, \beta^{i_2} \leftarrow \beta^i$. Set $i \leftarrow i + 1$ and go to Step i.1.

Details of each of the steps of the above algorithm are discussed in [1]. Here, we briefly describe some of the key features.

Lower Bounding: As mentioned earlier, at iteration i, we shall only consider partitions of the form $\mathcal{P}^i := \Pi_{j=1}^{m_2}[l_j^i, u_j^i)$, where l_j^i is such that $(h_j^n - l_j^i)$ is integral for some n. Consider the problem:

$$\begin{aligned} \beta^i := \quad \min \quad & f(x) + \theta && (20) \\ \text{s.t.} \quad & x \in X, \;\; Tx = \chi, \;\; l^i \leq \chi \leq u^i, \\ & \theta \geq \frac{1}{N}\sum_{n=1}^{N} \Psi^n(u^i - \epsilon), && (21) \end{aligned}$$

where ϵ is sufficiently small such that $\Psi^n(\cdot)$ is constant over $[u^i - \epsilon, u^i)$ for all n. This ϵ guarantees that the second stage value function for values of χ within the interior of the partition $\Pi_{j=1}^{m_2}[l_j^i, u_j^i)$ is approximated. Since we have exactly characterized the regions over which the $\Psi^n(\cdot)$ is constant, we can calculate ϵ *a priori*. The non-increasing property of $\widehat{\Psi}_N(\cdot)$ guarantees that the above problem provides a lower bound over the partition $\mathcal{P}^i$. To solve (20), we first need to solve N second stage subproblems $\Psi^n(\chi)$ to construct the cut (21). The master problem (20) can then be solved with respect to the variables (x, χ, θ). Each of the N subproblems and the master problem can be solved completely independently, so complete stage and scenario decomposition is achieved.

Upper Bounding: Let χ^i be an optimal solution of problem (20). Note that $\chi^i \in \mathcal{X}$, and is therefore a feasible solution. We can then compute an upper bound $\alpha^i := g(\chi^i) \geq \min\{g(\chi) | \chi \in \mathcal{X}\}$.

Fathoming: Once we have isolated a region over which the second stage value function is constant, the lower and upper bounds over this region become equal. Subsequently such a region is fathomed in Step i.2.c of the algorithm. In other words, if, for a partition $\mathcal{P}^i$, the second stage expected value function $\widehat{\Psi}_N(\cdot)$ is constant, then the partition $\mathcal{P}^i$ will be fathomed in the course of the algorithm.

Branching: To isolate the discontinuous pieces of the second stage value function, we are required to partition an axis j' at a point $\chi_{j'}$ such that $\Psi^n(\cdot)$ is possibly discontinuous at $\chi_{j'}$ for some n. We can do this by selecting $\chi_{j'}$ such that $h_{j'}^n - \chi_{j'}$ is integral for some n.

Using the fact that the lower and upper bounds used in the algorithm are always valid, and that there are only a finite number of partitions of the type $\mathcal{P}^i$ to consider, it can be shown that the DBB algorithm produces a global optimal solution to (19) in a finite number of iterations [1, 2]. Note that the global optimal solution vector is obtained from solving the lower bounding problem (20) corresponding to the partition for $\mathcal{P}$ for which the lower and upper bounds have collapsed. Since the bounds on χ for each partition satisfy: $h^n - z - 1 \leq \chi \leq$

$h^n - z$ for some integer z, it is clear that the x-component of the solutions to the lower bounding problems will always satisfy $x \in V_N$. Thus, the DBB algorithm produces an optimal solution to the SAA problem such that $\widehat{x}_N \in V_N$.

5 Solution Validation

In Sections 3 and 4, we have shown that a candidate solution $\widehat{x}_N$, obtained by appropriately solving a sample average approximating problem with sample size N, is a true optimal solution with probability approaching one exponentially fast as N increases. In this section, we describe techniques to estimate the optimality gap of a candidate solution $\widehat{x}_N$ for a finite sample size N.

5.1 Statistical Bounds

Recall that $\widehat{v}_N$ and v^* denote the optimal values of the SAA problem and the true problem, respectively. The following methodology of constructing statistical lower and upper bounds was suggested in [23] and developed further in [19].

It is well known that

$$\mathbb{E}[\widehat{v}_N] \leq v^*. \tag{22}$$

Thus we can obtain a lower bound to the true optimal value by estimating $\mathbb{E}[\widehat{v}_N]$ as follows. By generating M independent samples of the uncertain parameters, each of size N, and solving the corresponding SAA problems, we obtain optimal objective values $\widehat{v}_N^1, \ldots, \widehat{v}_N^M$. Then the quantity

$$\overline{v}_N^M = \frac{1}{M} \sum_{m=1}^{M} \widehat{v}_N^m \tag{23}$$

is an unbiased estimator of $\mathbb{E}[\widehat{v}_N]$, and therefore is a statistical lower bound to v^*. Note that an estimate of variance of the above estimator can be computed as

$$S^2_{\overline{v}_N^M} := \frac{1}{M(M-1)} \sum_{m=1}^{M} \left(\widehat{v}_N^m - \overline{v}_N^M \right)^2. \tag{24}$$

Now consider a feasible solution $\bar{x} \in X$. For example, we can take $\bar{x}$ to be equal to an optimal solution $\widehat{x}_N$ of an SAA problem. We can estimate the true objective value $g(\bar{x})$ at the point $\bar{x}$ by generating an independent sample $\xi^1, \ldots, \xi^{N'}$, of size N', and computing

$$\widehat{g}_{N'}(\bar{x}) = c^T \bar{x} + \frac{1}{N'} \sum_{n=1}^{N'} Q(\bar{x}, \xi^n). \tag{25}$$

We have that $\widehat{g}_{N'}(\bar{x})$ is an unbiased estimator of $c^T\bar{x} + \mathbb{E}[Q(\bar{x}, \xi)]$. Consequently, since $\bar{x}$ is a feasible point of the true problem, $\widehat{g}_{N'}(\bar{x})$ gives a statistical upper bound on the true optimal solution value. An estimate of the variance of the above estimator is given by:

$$S^2_{\widehat{g}_{N'}(\bar{x})} := \frac{1}{N'(N'-1)} \sum_{n=1}^{N'} \left[c^T\bar{x} + Q(\bar{x},\xi^n) - \widehat{g}_{N'}(\bar{x})\right]^2. \tag{26}$$

Using the above expressions, an estimate of the optimality gap of a candidate solution $\bar{x}$ is given by $\widehat{g}_{N'}(\bar{x}) - \overline{v}_N^M$, along with an estimated variance of $S^2_{\overline{v}_N^M} + S^2_{\widehat{g}_{N'}(\bar{x})}$.

5.2 Deterministic Bounds

In this section, we discuss how deterministic bounds on the true optimal value of certain class of stochastic programs with integer recourse can be computed without resorting to sampling. These bounds are based on replacing the uncertain parameters by their mean values.

A consequence of the classical *Jensen's* inequality is that the objective value of the mean-value problem corresponding to a stochastic linear program with only right hand side uncertainty provides a lower bound to the optimal value; and the objective value of the mean-value problem corresponding to a stochastic linear program with only cost uncertainty provides an upper bound on the optimal value. These results are not immediately applicable to stochastic programs with integer recourse, owing to the non-convexities in the value function. Next, we discuss mean value bounds for certain classes of stochastic integer programs.

Right-hand-side Uncertainty

Consider the deterministic equivalent to the canonical two-stage stochastic program with a finite number of scenarios and only right hand uncertainty:

$$\begin{aligned} v^* := \quad \min \quad & c^T x + \sum_{k=1}^{K} p_k q^T y_k & \quad (27)\\ \text{s.t.} \quad & x \in X, & \\ & y^k \in Y, \qquad k = 1,\dots,K, & \\ & T_k x + W y_k \geq h_k, \qquad k = 1,\dots,K. & \quad (28) \end{aligned}$$

The mean-value problem corresponding to (27) is obtained by replacing the stochastic parameters T_k and h_k by their mean values $\overline{T}$ and $\overline{h}$:

$$\begin{aligned} \overline{v} := \quad \min \quad & c^T x + q^T y & \quad (29)\\ \text{s.t.} \quad & x \in X, \quad y \in Y, & \\ & \overline{T}x + Wy \geq \overline{h}. & \quad (30) \end{aligned}$$

When the second stage constraint set Y is continuous polyhedral, i.e., the second stage problem is a linear program, it is well known from *Jensen's* inequality that $\overline{v} \leq v^*$.

Let us now consider instances of (27) where the set Y has integrality restrictions. Consider the surrogate relaxation of (27) obtained by taking the probability weighted sum of the constraints (28):

$$\begin{aligned} v^S := \quad \min \quad & c^T x + \sum_{k=1}^{K} p_k q^T y_k && (31) \\ \text{s.t.} \quad & x \in X, \\ & y_k \in Y, \quad k = 1, \dots, K, \\ & \overline{T} x + \sum_{k=1}^{K} p_k W y_k \geq \overline{h}. && (32) \end{aligned}$$

It is well known that $v^S \leq v^*$. The Lagrangian relaxation obtained by dualizing constraint (32) is

$$L(\lambda) = \quad \min_{x \in X} \left(c^T - \lambda^T \overline{T} \right) x + \sum_{k=1}^{K} p_k \left[\min_{y_k \in Y} (q^T - \lambda^T W) y_k \right] + \lambda^T \overline{h}.$$

Since the inner problem with respect to the second stage variables y_k is independent of the scenarios, the above expression simplifies to

$$L(\lambda) = \quad \min_{x \in X} (c^T - \lambda^T \overline{T}) x + \min_{y \in Y} (q^T - \lambda^T W) y + \lambda^T \overline{h}. \qquad (33)$$

¿From Lagrangian duality it is known that $L(\lambda) \leq v^S$, and hence $L(\lambda) \leq v^*$. Recall that the Lagrangian dual problem is $\max_{\lambda \geq 0} L(\lambda)$, hence $L(\lambda)$ is the value of any feasible solution. Observe that (33) is the Lagrangian relaxation of the mean-value problem (29) obtained by dualizing constraint (30).

We have thus shown that the objective value of any feasible solution to the Lagrangian dual problem obtained by dualizing the constraint (30) in the mean value problem provides a lower bound to the optimal objective value of the stochastic program with integer recourse.

Cost parameter Uncertainty

Consider now the stochastic program with uncertainties in the recourse cost parameters only:

$$\begin{aligned} v^* := \quad \min \quad & c^T x + \sum_{k=1}^{K} p_k q_k^T y_k \\ \text{s.t.} \quad & x \in X, \\ & y_k \in Y, && k = 1, \dots, K, \\ & T x + W y_k \geq h, && k = 1, \dots, K. \end{aligned}$$

The mean value problem corresponding to the above problem is obtained by replacing the stochastic parameters q_k by its mean values $\overline{q}$:

$$\begin{aligned} \overline{v} := \quad \min \quad & c^T x + \overline{q}^T y \\ \text{s.t.} \quad & x \in X, \\ & y \in Y, \\ & Tx + Wy \geq h. \end{aligned}$$

Let (x^*, y^*) be an optimal solution to the mean-value problem with a corresponding optimal objective value of $\overline{v}$. Clearly such a solution is feasible to the stochastic program and has an objective value of $\overline{v}$, thus $v^* \leq \overline{v}$. Thus the mean value problem provides a valid upper bound to the optimal value of the stochastic program with integer recourse.

Although the computation of mean-value bounds are cheaper, these are significantly weaker than the statistical bounds discussed in Section 5.1. However, it should be noted that these bounds are deterministic quantities and have no variability. The deterministic bounds can be used to discard poor candidate solutions without additional sampling. More usefully, the bounds can be precomputed and used to expedite the branch and bound optimization algorithm for solving the SAA problems.

6 Numerical Results

In this section, we describe some preliminary computational results using the sample average approximation method on a small test problem from the literature.

Summary of the proposed method

We begin by summarizing the proposed method. Let M be the number of replications, N be the number of scenarios in the sampled problem, and N' be the sample size used to estimate $c^T\bar{x} + \mathbb{E}[Q(\bar{x}, \xi)]$ for a given feasible solution $\bar{x}$. The SAA method can then be summarized is as follows:

1. For $m = 1, \ldots, M$, repeat the following steps:
   (a) Generate random sample $\xi^1, \ldots, \xi^N$.
   (b) Solve the SAA problem
   $$\operatorname*{Min}_{x \in X} \left\{ c^T x + \frac{1}{N} \sum_{n=1}^{N} Q(x, \xi^n) \right\},$$
   and let $\widehat{x}_N^m$ be the solution vector, and $\widehat{v}_N^m$ be the optimal objective value.

(c) Generate independent random sample $\xi^1, \ldots, \xi^{N'}$. Evaluate $\widehat{g}_{N'}(\widehat{x}_N^m)$ and $S^2_{\widehat{g}_{N'}(\widehat{x}_N^m)}$ using (25) and (26) respectively.

2. Evaluate $\overline{v}_N^M$ and $S^2_{\overline{v}_N^M}$ using (23) and (24) respectively.

3. For each solution $\widehat{x}_N^m$, $m = 1, \ldots, M$, estimate the optimality gap by $\widehat{g}_{N'}(\widehat{x}_N^m) - \overline{v}_N^M$, along with an estimated variance of $S^2_{\overline{v}_N^M} + S^2_{\widehat{g}_{N'}(\widehat{x}_N^m)}$. Choose one of the M candidate solutions.

Note that the optimization in step 1(b) requires us to use the decomposition based branch and bound algorithm of Section 4. The method outlined above returns several candidate solutions along with estimates of their optimality gap. The choice of a particular solution from this list can be based on some post-processing rules such as the smallest estimated gap, or the least estimated objective value. The parameters M, N, and N' may need to be adjusted to trade-off computational effort with the desired confidence level.

The Test Problem

We demonstrate the proposed methodology by solving the following instance of a two-stage stochastic integer program from the literature [28, 7]:

$$\begin{array}{ll} \text{Min} & -1.5x_1 - 4x_2 + \mathbb{E}[Q(x_1, x_2, \xi_1(\omega), \xi_2(\omega))] \\ \text{s.t.} & 0 \leq x_1, x_2 \leq 5, \end{array}$$

where

$$\begin{array}{lll} Q(x_1, x_2, \xi_1, \xi_2) := & \min & -16y_1 - 19y_2 - 23y_3 - 28y_4 \\ & \text{s.t.} & 2y_1 + 3y_2 + 4y_3 + 5y_4 \leq \xi_1 - \frac{2}{3}x_1 - \frac{1}{3}x_2 \\ & & 6y_1 + y_2 + 3y_3 + 2y_4 \leq \xi_2 - \frac{1}{3}x_1 - \frac{2}{3}x_2 \\ & & y_1, y_2, y_3, y_4 \in \{0, 1\}, \end{array}$$

and $\xi_1(\omega)$ and $\xi_2(\omega)$ are independent, and both have a uniform discrete distribution with 10000 equidistant equiprobable mass points in the interval $[5, 15]$. Thus the total number of scenarios in the problem is 10^8. Also note that the problem has uncertain parameters only on the right hand side.

Let us try to estimate the cardinality of the candidate set V as defined by (12) for our test problem. Note that for feasible values of x, the vector Tx takes on values inside the box $[0, 5] \times [0, 5]$. Then for any fixed value of the right-hand-side vector h^k, the vector $h^k - Tx$ is integer valued at the intersection of the 6 grid lines for which $h_1^k - T_1 x$ is integer with the 6 grid lines for which $h_2^k - T_2 x$ is integer. Thus for all K scenarios, both components of $h - Tx$ can be integer valued at a maximum of $(6K)^2$ grid points. The 4 bounds on x intersecting with the above mentioned $12K$ grid lines will introduce additional at most $48K$ vertices. We can thus bound the cardinality of the candidate set by $|V| \leq (36K^2 + 48K)$, where K is the number of scenarios.

Note that the considered discrete distribution of $\xi = (\xi_1, \xi_2)$ is obtained by a (very fine) discretization of the corresponding uniform distribution. Let us denote by P_1 the uniform probability distribution on the square region $S := [5, 15] \times [5, 15] \subset \mathbb{R}^2$, and by P_2 its equidistant discretization with r equidistant equiprobable mass points in each interval $[5, 15]$. We have then that the constant

$$\eta := \sup_{x \in X} \left| \mathbb{E}_{P_1}[Q(x, \xi(\omega)] - \mathbb{E}_{P_2}[Q(x, \xi(\omega)] \right| \tag{34}$$

decreases to zero as r increases at a rate of $O(r^{-1})$. Indeed, for a given x, the function $Q(x, \cdot)$ is constant except possibly on a finite number of orthogonal lines with integer distances between these lines. There are no more than 20 such lines which intersects the square region S. The discontinuity jump that $Q(x, \cdot)$ can have at a point of such a line is an integer and is bounded by the constant 86. Therefore, $\eta \leq cr^{-1}$ with $c = 86(20)$, for example. In fact, this estimate of the constant c is very crude and it appears that the convergence of η to zero is much faster. In the following computational experiments, the optimal solutions to the SAA problems were different from each other for all generated samples. This indicates that for the considered discretization with $r = 10000$ the error of the sample average approximation was considerably bigger than the constant η.

Implementation

The proposed sampling strategy and the optimization routine was implemented in C. To reduce the variance within the SAA scheme, we employed the Latin Hypercube Sampling scheme [20] to generate the uncertain problem parameters. In the DBB algorithm, we employed enumeration to solve the second stage integer programs, and the CPLEX 7.0 LP solver to solve the master linear program. A mean value lower bound for the test instance was pre-computed by solving the Lagrangian dual problem described in Section 5.2. The mean value solution and the mean value lower bound was used to initialize the branch and bound algorithm. The termination tolerance for the DBB scheme was set at 0.0001. All computations were performed on a 450 MHz Sun Ultra60 workstation.

Numerical results

We first demonstrate the need for special purpose solution strategies to solve the sample average approximating problem. In Table 1, we compare the branch and bound nodes and cpu seconds required by the DBB algorithm against that required by the CPLEX 7.0 integer programming solver for solving the SAA problem for sample sizes $N = 10$ to 50. Problems with sample sizes of 200 or higher, could not be solved by CPLEX 7.0 within 1000 CPU seconds. Consequently, specialized decomposition algorithms are crucial for solving the SAA problem efficiently – the DBB can solve 200 sample size instances in less than 5 seconds.

	CPLEX 7.0		DBB	
N	Nodes	CPUs	Nodes	CPUs
10	44	0.06	25	0.00
20	315	0.46	31	0.01
30	1661	2.15	49	0.01
40	3805	6.32	81	0.03
50	58332	96.94	81	0.04

Table 1: Comparison of CPLEX 7.0 and DBB

Tables 2 and 3 presents the computational results for using the SAA method combined with the DBB algorithm for sample sizes of $N = 20$ and $N = 200$. The number of replications was $M = 10$, and the sample size to estimate the objective value was $N' = 10000$. The computations correspond to simple Monte Carlo sampling, i.e. without the use of Latin Hypercube Sampling scheme. In these tables, the first column is the replication number m, the second and third columns are components of the corresponding first stage solution $\widehat{x}^m_N$, column four is the estimated objective function value with column five displaying the estimated variance, column six is the estimated lower bound, column seven is an estimate of the optimality gap of the solution $\widehat{x}^m$, and column eight is an estimate of variance of the optimality gap estimate. It is clear from these table, that even with modest sample size, very good quality solutions can be obtained by the proposed methodology. Increasing the sample size to $N = 200$, we were able to reduce the variance of the estimates from about 1.96 to about 0.107, a 95 % reduction.

m	x_1	x_2	$\widehat{g}_{N'}(x_1, x_2)$	$S^2_{\widehat{g}_{N'}(x_1,x_2)}$	$\widehat{v}^m_N$	Gap (est.)	Var
1	0.027	4.987	-60.80547	0.02302	-63.73667	0.19937	1.95858
2	0.000	4.245	-59.78910	0.02201	-55.38000	1.21573	1.95757
3	0.000	4.530	-60.50430	0.02228	-62.27000	0.50053	1.95784
4	0.027	4.987	-60.69477	0.02297	-63.73667	0.31007	1.95854
5	0.000	4.500	-60.44130	0.02223	-56.80000	0.56353	1.95779
6	0.000	3.900	-59.88580	0.02267	-61.45000	1.11903	1.95824
7	0.350	4.820	-59.76140	0.02266	-63.65500	1.24343	1.95822
8	0.000	5.000	-60.62720	0.02294	-69.10000	0.37763	1.95850
9	0.000	4.890	-60.77030	0.02292	-58.16000	0.23453	1.95848
10	0.000	2.865	-58.52530	0.02070	-55.76000	2.47953	1.95626
				$\overline{v}^M_N$	$= -61.00483$		
				$S^2_{\overline{v}^M_N}$	$= 1.93556$		

Table 2: Simple Monte Carlo $N = 20, N' = 10000, M = 10$

m	x_1	x_2	$\widehat{g}_{N'}(x_1, x_2)$	$S^2_{\widehat{g}_{N'}(x_1,x_2)}$	$\widehat{v}^m_N$	Gap (est.)	Var
1	0.000	4.905	-60.71670	0.02284	-61.33000	0.90596	0.10745
2	0.000	4.980	-60.44850	0.02248	-62.36500	1.17417	0.10709
3	0.000	4.890	-60.75090	0.02268	-62.05000	0.87177	0.10730
4	0.007	4.997	-60.69747	0.02304	-60.68667	0.92520	0.10765
5	0.000	4.650	-60.49100	0.02249	-59.65500	1.13167	0.10711
6	0.000	4.110	-60.14670	0.02302	-61.38000	1.47597	0.10764
7	0.000	4.890	-60.46580	0.02263	-62.86500	1.15687	0.10725
8	0.020	4.760	-60.35070	0.02255	-62.18000	1.27197	0.10717
9	0.070	4.945	-60.59350	0.02279	-61.97000	1.02917	0.10741
10	0.000	4.995	-60.58240	0.02274	-61.74500	1.04027	0.10736
					$\overline{v}^M_N = -61.62267$		
					$S^2_{\overline{v}^M_N} = 0.08462$		

Table 3: Simple Monte Carlo $N = 200, N' = 100, M = 10$

Tables 4 and 5 demonstrates the effect of the variance reduction by using Latin Hypercube Sampling (LHS). For sample sizes, $N = 20$ the variance estimate was reduced from 1.957 to 0.119, a 94 % reduction; and for $N = 200$, the variance estimate was reduced from 0.107 to 0.036, a 66 % reduction. The robustness of the candidate optimal solutions is also evident from these tables.

m	x_1	x_2	$\widehat{g}_{N'}(x_1, x_2)$	$S^2_{\widehat{g}_{N'}(x_1,x_2)}$	$\widehat{v}^m_N$	Gap (est.)	Var
1	0.010	4.990	-60.61650	0.02281	-62.77500	1.02600	0.11972
2	0.000	5.000	-60.59210	0.02290	-61.40000	1.05040	0.11981
3	0.000	4.920	-60.59980	0.02273	-61.58000	1.04270	0.11964
4	0.000	4.170	-59.84780	0.02212	-62.78000	1.79470	0.11903
5	0.000	3.780	-59.48660	0.02203	-59.52000	2.15590	0.11895
6	0.000	4.230	-59.94800	0.02206	-62.22000	1.69450	0.11897
7	0.000	5.000	-60.67800	0.02271	-61.65000	0.96450	0.11962
8	0.000	4.935	-60.57800	0.02275	-62.09000	1.06450	0.11967
9	0.000	4.770	-60.39320	0.02257	-61.78000	1.24930	0.11949
10	0.000	4.695	-60.39980	0.02262	-60.63000	1.24270	0.11953
					$\overline{v}^M_N = -61.64250$		
					$S^2_{\overline{v}^M_N} = 0.09691$		

Table 4: With LHS $N = 20, N' = 10000, M = 10$

7 Concluding Remarks

In this paper, we have extended the sample average approximation method to two-stage stochastic programs with integer recourse, continuous first-stage

m	x_1	x_2	$\widehat{g}_{N'}(x_1,x_2)$	$S^2_{\widehat{g}_{N'}(x_1,x_2)}$	$\widehat{v}^m_N$	Gap (est.)	Var
1	0.000	4.995	-60.65330	0.02302	-60.86500	0.18987	0.03613
2	0.000	4.740	-60.42840	0.02254	-60.48500	0.41477	0.03565
3	0.007	4.997	-60.74907	0.02282	-61.39167	0.09410	0.03593
4	0.000	4.890	-60.57400	0.02270	-60.99500	0.26916	0.03581
5	0.000	4.965	-60.64100	0.02274	-60.80500	0.20217	0.03585
6	0.000	4.980	-60.53170	0.02269	-61.26000	0.31147	0.03580
7	0.000	4.980	-60.71420	0.02311	-61.08500	0.12897	0.03623
8	0.000	4.995	-60.68270	0.02279	-60.62000	0.16047	0.03590
9	0.000	4.980	-60.57940	0.02285	-60.18500	0.26377	0.03596
10	0.000	4.980	-60.69060	0.02296	-60.74000	0.15257	0.03607
					$\overline{v}^M_N = -60.84317$		
					$S^2_{\overline{v}^M_N} = 0.01311$		

Table 5: With LHS $N = 200, N' = 10000, M = 10$

variables and a huge number of scenarios. The proposed methodology relies on constructing approximate problems via sampling, and solving these problems using a novel optimization algorithm. We have argued that the proposed scheme will produce an optimal solution to the true problem with probability approaching one exponentially fast as the sample size is increased. For fixed sample size, we have described statistical and deterministic bounds to validate the quality of a candidate optimal solution. Our preliminary computational experiments have demonstrated the efficacy of the proposed method.

References

[1] S. Ahmed. *Strategic Planning under Uncertainty: Stochastic Integer Programming Approaches.* Ph.D. Thesis, University of Illinois at Urbana-Champaign, 2000.

[2] S. Ahmed, M. Tawarmalani, and N. V. Sahinidis. A finite branch and bound algorithm for two-stage stochastic integer programs. Submitted for publication. E-print available in the Stochastic Programming E-Print Series: `http://dochost.rz.hu-berlin.de/speps/`, 2000.

[3] D. Bienstock and J. F. Shapiro. Optimizing resource acquisition decisions by stochastic programming. *Management Science*, 34:215–229, 1988.

[4] J. R. Birge and M. A. H. Dempster. Stochastic programming approaches to stochastic scheduling. *Journal of Global Optimization*, 9(3-4):417–451, 1996.

[5] J. R. Birge and F. Louveaux. *Introduction to Stochastic Programming.* Springer, New York, NY, 1997.

[6] G. R. Bitran, E. A. Haas, and H. Matsuo. Production planning of style goods with high setup costs and forecast revisions. *Operations Research*, 34:226–236, 1986.

[7] C. C. Caroe. *Decomposition in stochastic integer programming.* PhD thesis, University of Copenhagen, 1998.

[8] C. C. Caroe, A. Ruszczynski, and R. Schultz. Unit commitment under uncertainty via two-stage stochastic programming. In *Proceedings of NOAS 97,* Caroe *et al.* (eds.), Department of Computer Science, University of Copenhagen, pages 21–30, 1997.

[9] M. A. H. Dempster. A stochastic approach to hierarchical planning and scheduling. In *Deterministic and Stochastic Scheduling,* Dempster *et al.*(eds.), D. Riedel Publishing Co., Dordrecht, pages 271–296, 1982.

[10] M. A. H. Dempster, M. L. Fisher, L. Jansen, B. J. Lageweg, J. K. Lenstra, and A. H. G. Rinnooy Kan. Analytical evaluation of hierarchical planning systems. *Operations Research*, 29:707–716, 1981.

[11] C. L. Dert. *Asset Liability Management for Pension Funds, A Multistage Chance Constrained Programming Approach*. PhD thesis, Erasmus University, Rotterdam, The Netherlands, 1995.

[12] P. Kall and S. W. Wallace. *Stochastic Programming*. John Wiley and Sons, Chichester, England, 1994.

[13] 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.

[14] G. Laporte, F. V. Louveaux, and H. Mercure. Models and exact solutions for a class of stochastic location-routing problems. *European Journal of Operational Research*, 39:71–78, 1989.

[15] G. Laporte, F. V. Louveaux, and H. Mercure. The vehicle routing problem with stochastic travel times. *Transportation Science*, 26:161–170, 1992.

[16] G. Laporte, F. V. Louveaux, and L. van Hamme. Exact solution of a stochastic location problem by an integer L-shaped algorithm. *Transportation Science*, 28(2):95–103, 1994.

[17] J. K. Lenstra, A. H. G. Rinooy Kan, and L. Stougie. A framework for the probabilistic analysis of hierarchical planning systems. Technical report, Mathematisch Centrum, University of Amsterdam, 1983.

[18] J. Linderoth, A. Shapiro, and S. Wright. The empirical behavior of sampling methods for stochastic programming. Optimization Technical Report 02-01, Computer Sciences Department, University of Wisconsin-Madison, 2002.

[19] 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.

[20] M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21:239–245, 1979.

[21] V. I. Norkin, Y. M. Ermoliev, and A. Ruszczynski. On optimal allocation of indivisibles under uncertainty. *Operations Research*, 46:381–395, 1998.

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

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

[24] E. L. Plambeck, B. R. Fu, S. M. Robinson, and R. Suri. Sample-path optimization of convex stochastic performance functions. *Mathematical Programming, Series B*, 75:137–176, 1996.

[25] R. Y. Rubinstein and A. Shapiro. Optimization of static simulation models by the score function method. *Mathematics and Computers in Simulation*, 32:373–392, 1990.

[26] R. Schultz. On structure and stability in stochastic programs with random technology matrix and complete integer recourse. *Mathematical Programming*, 70(1):73–89, 1995.

[27] R. Schultz. Rates of convergence in stochastic programs with complete integer recourse. *SIAM J. Optimization*, 6(4):1138–1152, 1996.

[28] R. Schultz, L. Stougie, and M. H. van der Vlerk. Solving stochastic programs with integer recourse by enumeration: A framework using Gröbner basis reductions. *Mathematical Programming*, 83:229–252, 1998.

[29] A. Shapiro and T. Homem de Mello. A simulation-based approach to two-stage stochastic programming with recourse. *Mathematical Programming*, 81(3, Ser. A):301–325, 1998.

[30] A. Shapiro and T. Homem-de-Mello. On rate of convergence of Monte Carlo approximations of stochastic programs. *SIAM Journal on Optimization*, 11:70–86, 2001.

[31] A. M. Spaccamela, A. H. G. Rinnooy Kan, and L. Stougie. Hierarchical vehicle routing problems. *Networks*, 14:571–586, 1984.

[32] S. Takriti, J. R. Birge, and E. Long. A stochastic model of the unit commitment problem. *IEEE Transactions on Power Systems*, 11:1497–1508, 1996.

[33] S. R. Tayur, R. R. Thomas, and N. R. Natraj. An algebraic geometry algorithm for scheduling in the presence of setups and correlated demands. *Mathematical Programming*, 69(3):369–401, 1995.

[34] B. Verweij, S. Ahmed, A. J. Kleywegt, G. Nemhauser, and A. Shapiro. The sample average approximation method applied to stochastic routing problems: A computational study. Submitted for publication. E-print available at `http://www.optimization-online.org`, 2001.

[35] R. J-B. Wets. Programming under uncertainty: The solution set. *SIAM Journal on Applied Mathematics*, 14:1143 – 1151, 1966.