JOURNAL OF OPTIMIZATION THEORY AND APPLICATIONS: Vol. 125, No. 2, pp. 349–366, May 2005 (© 2005)
DOI: 10.1007/s10957-004-1842-z

On the Convergence of Sampling-Based Decomposition Algorithms for Multistage Stochastic Programs[1]

K. Linowsky[2] and A. B. Philpott[3]

Communicated by W. B. Gong

Abstract. The paper presents a convergence proof for a broad class of sampling algorithms for multistage stochastic linear programs in which the uncertain parameters occur only in the constraint right-hand sides. This class includes SDDP, AND, ReSa, and CUPPS. We show that, under some independence assumptions on the sampling procedure, the algorithms converge with probability 1.

Key Words. Multistage stochastic programming, sampling, almost sure convergence.

1. Introduction

Multistage stochastic linear programming models have many applications, but they are notoriously difficult to solve. The most successful approaches in practical applications appear to be the sampling-based methods. The first of these (SDDP) was developed by Pereira and Pinto (Ref. 1) in the context of hydroelectricity planning.[4] This algorithm has been applied successfully (see Ref. 2) to compute solutions to long-term hydrothermal reservoir planning models. To the authors' knowledge, no convergence result for this method has appeared in the literature. Since the Pereira and Pinto paper, a number of related algorithms have emerged [see

[1] The first author acknowledges support by the Swiss National Science Foundation. The second author acknowledges support by NZPGST Grant UOAX0203. The authors are grateful to the anonymous referees for comments improving the exposition of this paper.

[2] PhD Student, University of St. Gallen, St. Gallen, Switzerland.
[3] Professor, University of Auckland, Auckland, New Zealand.

[4] SDDP stands for stochastic dual dynamic programming method.

e.g. CUPPS (Ref. 2), AND (Ref. 3), and ReSa (Ref. 4)] based on similar ideas.[5]

In this paper, we derive a general convergence result for algorithms of this type. A convergence proof specifically aimed at the CUPPS algorithm has appeared already in Ref. 2. The argument that we employ in our proof resembles closely that used in Ref. 2, in that we use the same induction on stages. However, our result is more general, being applicable to SDDP, AND, CUPPS, and ReSa. The main contribution of our work is to identify the crucial conditions that guarantee convergence of sampling-based multistage stochastic Benders decomposition methods. These assumptions are made precise in the sequel, but essentially they amount to the following requirements:

(i) cuts should be computed eventually in every stage;
(ii) samples that are used to create scenarios in the forward pass are also used in cut generation.

In Section 2, we give a general formulation of the multistage stochastic programming problem. Section 3 describes the general algorithmic approach. The convergence proof is then derived in Section 4 using a series of lemmas. In Section 5, we show how the sampling-based algorithms of Refs. 1 and 3–5 satisfy the conditions of the theorem.

2. Multistage Decomposition

Multistage stochastic linear programs with recourse are well known in the stochastic programming community. The general form of these programs is described in Ref. 6. In this paper, we restrict our attention to multistage stochastic programs with the following properties:

(A1) Random quantities appear only on the right-hand side of the linear constraints in each stage.
(A2) The set Ω_t of random outcomes in each stage t is discrete and finite,

$$\Omega_t = \{w_{ti} | i = 1, \ldots, q_t < \infty\} \text{ with probabilities } p_{ti} > 0, \forall i.$$

(A3) Random quantities in different stages are independent.
(A4) The feasible region of the linear program in each stage is nonempty and bounded.

[5] CUPPS stands for convergent cutting plane and particle sampling method. AND stands for abridged nested decomposition method. ReSa stands for reduced sampling method.

Under these assumptions, the multistage stochastic linear program can be written in the following form:

Solve the problem defined by

$$
\text{(LP1)}\quad \begin{aligned} Q_1 = \min\nolimits_{x_1} \quad & c_1^\top x_1 + \mathcal{Q}_2(x_1), \\ \text{s.t.} \quad & A_1 x_1 = b_1, \\ & x_1 \geq 0, \end{aligned}
$$

where for all $t = 2, \ldots, T$,

$$
\mathcal{Q}_t(x_{t-1}) = \sum_{i=1}^{q_t} p_{ti}\, Q_t(x_{t-1}, \omega_{ti}),
$$

$Q_t(x_{t-1}, \omega_{ti})$ is defined by the problem

$$
\text{(LP}t\text{)}\quad \begin{aligned} Q_t(x_{t-1}, \omega_t) = \min\nolimits_{x_t} c_t^\top x_t + {} & \mathcal{Q}_{t+1}(x_t), \\ \text{s.t.} \quad & A_t x_t = \omega_t - B_{t-1} x_{t-1}, \\ & x_t \geq 0, \end{aligned}
$$

and we set $\mathcal{Q}_{T+1} \equiv 0$.

Problem [LP*t*] depends on the choice of *wt* and x_{t-1}, and so we could write [LP$t(x_{t-1}, \omega_t)$], though we choose to suppress this dependence in the notation. By Assumption (A3), [LP*t*] is independent of $\omega_{t-1}, \omega_{t-2}, \ldots$. Observe that $Q_t(x_{t-1}, \omega_t)$ is a polyhedral convex function and so is continuous in x_{t-1} at all points of its domain.

In the algorithms that are considered in this paper, the functions $\mathcal{Q}_t(x_{t-1})$ in each stage are approximated by the maximum of a collection of linear functions, each of which is called a cut. This gives rise to a sequence of approximate problems [APt^k] for each stage. These are defined for iteration k as follows.

For $t = 1$, solve the linear program

$$
\text{(AP}1^k\text{)}\quad \begin{aligned} C_1^k = \min\nolimits_{x_1, \theta_2} \quad & c_1^\top x_1 + \theta_2, \\ \text{s.t.} \quad & A_1 x_1 = b_1, \\ & \theta_2 + (\beta_2^j)^\top x_1 \geq \alpha_{2,j}, \quad j = 0, \ldots, k-1, \\ & x_1 \geq 0. \end{aligned}
$$

For all $t=2,\dots,T-1$, solve

$$\begin{aligned}(\mathrm{AP}t^k)\quad C_t^k(x_{t-1},\omega_t)=\min\nolimits_{x_t,\theta_{t+1}}\quad & c_t^\top x_t+\theta_{t+1},\\ \text{s.t.}\quad & A_t x_t=\omega_t-B_{t-1}x_{t-1},\\ & \theta_{t+1}+(\beta_{t+1}^j)^\top x_t\geq\alpha_{t+1,j},\quad j=0,\dots,k-1,\\ & x_t\geq 0.\end{aligned}$$

For all stages, the first cut ($j=0$) is set as the trivial cut $\theta_{t+1}\geq-\infty$. We shall use the notation (π_t,ρ_t) to denote dual variables of problem [APt^k], where π_t corresponds to the equality constraints and ρ_t corresponds to the cut constraints. We use also the notation $C_t^k(x_{t-1})$ to denote $\Sigma_{i=1}^{q_t} p_{ti} C_t^k(x_{t-1},\omega_t)$.

Observe that, under Assumption (A4),

$$\{x_t \mid A_t x_t=\omega_t-B_{t-1}x_{t-1},\ x_t\geq 0\}$$

is nonempty and bounded so [APt^k] always has a nonempty feasible set (with θ_{t+1} chosen large enough) and hence an optimal solution. Thus, the dual feasible region of [APt^k] is nonempty. Moreover, by Assumption (A1), the dual feasible sets are independent of the outcomes of the random quantities, which allows us to construct a valid cut at each stage based on an assembled collection of dual solutions from different samples.

In the last stage T, the algorithms solve the actual problem [LPT]; therefore,

$$C_T^k(x_{T-1},\omega_T)=Q_T(x_{T-1},\omega_T),\quad \forall k.$$

Since cuts are added from one iteration to the next and since no cuts are taken out, the objective values of the approximated problems form a monotone sequence; i.e.,

$$C_t^{k+1}(x_{t-1},\omega_t)\geq C_t^k(x_{t-1},\omega_t),\quad \forall t,\forall k.$$

3. Class of Sampling-Based Decomposition Algorithms

In this section, we define a general class of sampling algorithms for solving [LP1]. We describe first the cut generation of the algorithms.

Definition 3.1. Sampled Cut. A sampled cut at x_{t-1}^k with sample $\Omega_t^k\subseteq\Omega_t$ is computed as follows.

Step 1. Solve [APt^k] for all $\omega_{ti} \in \Omega_t^k$ and let $(\pi_t^i(x_{t-1}^k), \rho_t^i(x_{t-1}^k))$ be the optimal dual variables attained at an extreme point. Add them to the set $\mathcal{D}_t^k$.

Step 2. For all $\omega_{ti} \notin \Omega_t^k$, set

$$\left(\pi_t^i\left(x_{t-1}^k\right), \rho_t^i\left(x_{t-1}^k\right)\right)$$
$$= \text{arg max } \left\{\pi_t^\top(\omega_{ti} - B_{t-1}x_{t-1}^k + \rho_t^\top(\alpha_{t+1}^k) | (\pi_t, \rho_t) \in \mathcal{D}_t^k\right\},$$

or if $t = T$, set

$$\pi_T^i(x_{T-1}^k) = \text{arg max } \left\{\pi_T^\top(\omega_{Ti} - B_{T-1}x_{T-1}^k) | \pi_T \in \mathcal{D}_T^k\right\}.$$

Step 3. The cut has the formula

$$\theta_t \geq \alpha_{t,k} - (\beta_t^k)^\top x_{t-1},$$

where

$$\beta_t^k = \sum_{i=1}^{qt} p_{ti} B_{t-1}^\top \pi_t^i\left(x_{t-1}^k\right), \qquad 2 \leq t \leq T,$$
$$\alpha_{t,k} = \sum_{i=1}^{qt} p_{ti}\left[\omega_{ti}^\top \pi_t^i(x_{t-1}^k) + (\alpha_{t+1}^{k-1})^\top \rho_t^i(x_{t-1}^k)\right], \qquad 2 \leq t \leq T-1,$$
$$\alpha_{T,k} = \sum_{i=1}^{qt} p_{Ti}\omega_{Ti}^\top \pi_T^i\left(x_{T-1}^k\right).$$

Observe that $\alpha_{t,k}$ is a scalar, whereas α_{t+1}^{k-1} denotes a $(k-1)$-dimensional vector. This means that the dimensions of α_{t+1}^{k-1} and $\rho_t^i(x_{t-1}^k)$ are increasing as the iteration count k increases. Note also that a sampled cut is well defined for $\Omega_t^k = \emptyset$, as long as $\mathcal{D}_t^k \neq \emptyset$. If $\Omega_t^k = \mathcal{D}_t^k = \emptyset$, then we set $\alpha_{t,k} = -\infty$, $\beta_t^k = 0$.

In our convergence proof, we shall make use of the fact that $\pi_t^i(x_{t-1}^k)$ lies in a bounded set. In fact, $\pi_t^i(x_{t-1}^k)$ can take only a finite number of values in the course of the algorithm. This is a consequence of the fact that π_t and ρ_t are chosen to be extreme-point solutions of the dual of [APt^k]. We state this result formally as the following lemma.

Lemma 3.1. For all t, there is some m_t such that $\mathcal{D}_t^k$ has cardinality at most m_t.

Proof. We use induction on t. First, if $(\pi,\rho)\in\mathcal{D}_T^k$, then $\rho=0$ and π is an extreme point of $\{\pi|A_T^\top\pi\le c_T\}$ of which there is only a finite number. So, $|\mathcal{D}_T^k|\le m_T$, for some m_T.

Now, suppose that $|\mathcal{D}_t^k|\le m_t$. Then, the vector

$$\beta_t^j=\sum_{i=1}^{qt} p_{ti}B_{t-1}^\top\pi_t^i(x_{t-1}^j)$$

takes at most $(m_t)^{qt}$ values. This means that $\mathcal{E}_{t-1}^k$, the set of extreme points of

$$\left\{(\pi_{t-1},\rho_{t-1})|A_{t-1}^\top\pi_{t-1}+\sum_{j=1}^{k-1}\beta_t^j\rho_{t-1}^j\le c_{t-1},\sum_{j=1}^{k-1}\rho_{t-1}^j=1\right\},$$

has cardinality no more than m_{t-1}, say, independent of k. But $\mathcal{E}_{t-1}^k\supseteq\mathcal{D}_{t-1}^k$, which establishes the result. □

Now, a general class of sampling-based decomposition algorithms is defined, for which we will show convergence to the optimal solution. The algorithms work in the following way.

Multistage Sampled Benders Decomposition (MSBD).

Step 0. Initialization. Set the iteration counter to $k=1$.

Step 1. Candidate Solutions. In each iteration k, a complete sample path $\{\omega_t^k\}_{t=2,\dots,T}$ of the scenario tree is constructed independently of previous iterations. For this path, the approximate problems are solved up to stage $T-1$, to yield the primal solutions $(x_t^k,\theta_{t+1}^k)$of problem [APt^k].

Step 2. Cut Generation. For each stage $t=2,\dots,T$, sampled cuts are generated at x_{t-1}^k with sample Ω_t^k.

Step 3. Set $k=k+1$ and go to Step 1.

Note that, at each stage t, two samples are used in each iteration. Unless $\Omega_t^k=\{\omega_t^k\}$, these samples may be different. Observe also that they need not be independent; in fact, one might choose $\Omega_t^k=\{\omega_t^k\}$. However, in order to yield a convergence result for MSBD, we will require the following properties of the sampling procedure.

Definition 3.2. Cut-Sampling Property. MSBD is said to fulfill the cut-sampling property (CSP) if, for each stage t, $\{k|\Omega_t^k=\emptyset\}$ is finite.

Definition 3.3. Sample-Intersection Property. MSBD is said to fulfill the sample-intersection property (SIP) if, for each stage t and every outcome $\omega_{ti} \in \Omega_t$, $\Pr[(\omega_{ti} \in \Omega_t^k) \cap (\omega_t^k = \omega_{ti})] > 0$ for every k with $\Omega_t^k \neq \emptyset$.

The cut-sampling property entails that eventually the algorithm will compute $(\pi_t^i(x_{t-1}^k), \rho_t^i(x_{t-1}^k))$ for at least one outcome ω_{ti} at every stage. The sample-intersection property guarantees that the outcomes that are used to compute cuts will (with positive probability) include some information from the outcome in the sample path that is constructed in Step 1. SIP holds for example if ω_t^k and Ω_t^k are sampled independently or alternatively if Ω_t^k is chosen so as to include always ω_t^k as in CUPPS.

Lemma 3.2. Suppose that MSBD satisfies SIP. Then, it fulfills CSP if and only if, for any stage t and every infinite subsequence $\{x_{t-1}^k\}_{k \in J}$ generated by the algorithm, for each $i = 1, \ldots, q_t$ the subsequence $\{x_{t-1}^k\}_{k \in J_i}$ with $J_i = J \cap \{k | \omega_{ti} \in \Omega_t^k \text{ and } \omega_t^k = \omega_{ti}\}$ is infinite with probability one (wp1).

Proof. For some arbitrary stage t, let $\{x_{t-1}^k\}_{k \in J}$ be an infinite subsequence generated by the algorithm and suppose that $\{k | \Omega_t^k = \emptyset\}$ is finite by CSP. Then, the intersection $J \cap K$, where K denotes $\{k | \Omega_t^k \neq \emptyset\}$, is infinite. Furthermore, by SIP, for every $k \in J \cap K$ we have

$$\Pr[(\omega_{ti} \in \Omega_t^k) \cap (\omega_t^k = \omega_{ti})] > 0, \quad \forall i.$$

Due to the Borel-Cantelli lemma (see e.g. Ref. 7), the set $J_i \subseteq J \cap K$ with $\omega_{ti} \in \Omega_t^k$ and $\omega_t^k = \omega_{ti}$ for all $k \in J_i$ is infinite wp1.

Suppose now that CSP does not hold. Then, for some stage t, the set

$$K' = \{k | \Omega_t^k = \emptyset\}$$

is infinite. Then, for any subsequence J' of K',

$$\Omega_t^k = \emptyset, \quad \forall k \in J'. \qquad \square$$

Lemma 3.3. The sampled cuts are valid cuts. Furthermore, the following relations hold:

$$Q_t(x_{t-1}^k) \geq \theta_t^k, \qquad \forall k \in \mathcal{N}, \forall t = 2, \ldots, T,$$
$$Q_{t-1}(x_{t-2}, \omega_{t-1}) \geq C_{t-1}^k(x_{t-2}, \omega_{t-1}) \; \forall x_{t-2}, \omega_{t-1}, \forall k \in \mathcal{N}, \forall t = 2, \ldots, T.$$

Proof. The proof of this lemma can be obtained equivalently to Lemmas 4.1 and 4.3 in Ref. 3. □

Lemma 3.4. Suppose that MSBD satisfies SIP and CSP. Then, for any convergent sequence $\{x_{t-1}^k\}_{k\in J}$ generated by MSBD, there exists wp1 a sequence $\{\Delta^k\}_{k\in J}$ and q_t disjoint subsequences of J indexed by $r_i \in J_i$ with

(i) $\theta_i^k \ge \sum_{i=1}^{qt} p_{ti} C_t^{r_i}(x_{t-1}^{r_i}, \omega_{ti}) + \Delta^k$,
(ii) $J_i \subseteq J \cap \{k|\omega_{ti} \in \Omega_t^k, \omega_t^k = \omega_{ti}\}$,
(iii) $r_i < k$ for all but a finite number of k and $r_i \to \infty$ as $k \to \infty$,
(iv) $\lim_{k\to\infty} |\Delta^k| = 0$.

Proof. Consider stage $t \in \{2, \dots, T-1\}$; stage T can be treated in a similar way.

Let $\{x_{t-1}^k\}_{k\in J}$ be a convergent sequence and consider iteration $k \in J$. All cuts generated up to this iteration must be satisfied; so, for all $r < k$,

$$\begin{aligned}\theta_t^k &\ge \alpha_{t,r} - (\beta_t^r)^\top x_{t-1}^k \\ &= \sum_{i=1}^{qt} p_{ti} \left[(\pi_t^i(x_{t-1}^r))^\top(\omega_{ti} - B_{t-1}x_{t-1}^k) + (\alpha_{t+1}^r)^\top \rho_t^i(x_{t-1}^r)\right] \\ &= \sum_{i=1}^{qt} p_{ti} \left[(\pi_t^i(x_{t-1}^r))^\top(\omega_{ti} - B_{t-1}x_{t-1}^r) + (\alpha_{t+1}^r)^\top \rho_t^i(x_{t-1}^r)\right] + \Delta_1^k,\end{aligned}$$

with

$$\Delta_1^k = \sum_{i=1}^{qt} p_{t,i}(\pi_t^i(x_{t-1}^r))^\top B_{t-1}(x_{t-1}^r - x_{t-1}^k).$$

From, Lemma 3.2, for each $i = 1, \dots, q_t$ the set J has an infinite subset J_i wp1, such that $\omega_{ti} \in \Omega_t^{r_i}$ and $\omega_t^{r_i} = \omega_{ti}$ for all $r_i \in J_i$. Choose r_i as the largest member of the set J_i which is smaller than k. Since the sets J_i are infinite, $r_i \to \infty$ as $k \to \infty$.

Since the dual optimal solution of iteration r_i, $(\pi_t^i(x_{t-1}^{r_i}), \rho_t^i(x_{t-1}^{r_i}))$, is dual feasible in iteration $r = \max\{r_i | i = 1, \dots, q_t\}$, we have

$$\theta_t^k \ge \sum_{i=1}^{qt} p_{ti} \left[(\pi_t^i(x_{t-1}^{r_i}))^\top(\omega_{ti} - B_{t-1}x_{t-1}^r) + (\alpha_{t+1}^r)^\top \rho_t^i(x_{t-1}^{r_i})\right] + \Delta_1^k. \quad (1)$$

Here, we adopt the convention that $\rho_t^i(x_{t-1}^{r_i})$ has zero components added to give it dimension r. This means that

$$(\alpha_{t+1}^r)^\top \rho_t^i(x_{t-1}^{r_i}) = (\alpha_{t+1}^{r_i})^\top \rho_t^i(x_{t-1}^{r_i}),$$

so the right-hand side of (1) becomes

$$\sum_{i=1}^{qt} p_{ti}\left[(\pi_t^i(x_{t-1}^{r_i}))^\top(\omega_{ti} - B_{t-1}x_{t-1}^{r_i}) + (\alpha_{t+1}^{r_i})^\top \rho_t^i(x_{t-1}^{r_i})\right] + \Delta_1^k + \Delta_2^k$$
$$= \sum_{i=1}^{qt} p_{ti} C_t^{r_i}(x_{t-1}^{r_i}, \omega_{ti}) + \Delta_1^k + \Delta_2^k,$$

where

$$\Delta_2^k = \sum_{i=1}^{qt} p_{t,i}(\pi_t^i(x_{t-1}^{r_i}))^\top B_{t-1}(x_{t-1}^{r_i} - x_{t-1}^r).$$

Now,

$$|\Delta_1^k| \le \sum_{i=1}^{qt} p_{t,i}||\pi_t^i(x_{t-1}^r)||\ ||B_{t-1}(x_{t-1}^r - x_{t-1}^k||;$$

since the dual extreme points are bounded (because $\mathcal{D}_t^k$ is a bounded set) and since the sequence $\{x_{t-1}^k\}_{k\in J}$ is convergent, we have, with $r \to \infty$ as $k \to \infty$,

$$\lim_{k\to\infty} |\Delta_1^k| = 0.$$

Similarly,

$$\lim_{k\to\infty} |\Delta_2^k| = 0.$$

This completes the proof. □

4. Convergence of the Algorithm

In this section, we prove the convergence of algorithms that satisfy SIP and CSP by induction on the stage t. Following Ref. 3, we prove first two lemmas that establish this induction.

Lemma 4.1. Suppose that MSBD satisfies CSP and SIP. For any given infinite set $K \subseteq \mathcal{N}$, assume that:

(i) $\omega^k_{T-1} = \omega^0_{T-1}$ for some given $\omega^0_{T-1}$for any $k \in K$;
(ii) the sequence $\{x^k_{T-2}\}_{k\in K}$ converges to some given vector x^0_{T-2}.

Then there exists an infinite set $J \subseteq K$ such that:

(a) the sequence $\{x^k_{T-1}\}_{k\in J}$ converges to some vector x^0_{T-1};
(b) the sequence $\{\theta^k_T\}_{k\in J}$ converges to $\mathcal{Q}_T(x^0_{T-1})$wp1;
(c) the sequence $\{C^k_{T-1}(x^k_{T-2}, \omega^0_{T-1})\}_{k\in J}$ converges to $Q_{T-1}(x^0_{T-2}, \omega^0_{T-1})wp$1.

Proof. (a) By assumption, the primal feasible sets are bounded. But every infinite bounded sequence has a convergent subsequence. Denote x^0_{T-1} as the corresponding limit and J as the corresponding index set.

(b) From Lemma 3.3 and Lemma 3.4, we have wp1 a sequence $\{\Delta^k\}_{k\in J}$ and subsequences of J indexed by r_i with

$$\begin{aligned}\mathcal{Q}_T(x^k_{T-1}) &\geq \theta^k_T\\ &\geq \sum_{i=1}^{q_T} p_{Ti} C^{r_i}_T(x^{r_i}_{T-1}, \omega_{Ti}) + \Delta^k\\ &= \mathcal{Q}_T(x^k_{T-1}) + \sum_{i=1}^{q_T} p_{Ti}\left[C^{r_i}_T(x^{r_i}_{T-1}, \omega_{Ti}) - Q_T(x^k_{T-1}, \omega_{Ti})\right] + \Delta^k\\ &= \mathcal{Q}_T(x^k_{T-1}) + \sum_{i=1}^{q_T} p_{Ti}\left[Q_T(x^{r_i}_{T-1}, \omega_{Ti}) - Q_T(x^k_{T-1}, \omega_{Ti})\right] + \Delta^k,\end{aligned}$$

which yields

$$|\theta^k_T - \mathcal{Q}_T(x^k_{T-1})| \leq |\Delta^k_1|,$$

with

$$\Delta^k_1 = \sum_{i=1}^{q_T} p_{Ti} \;\left[Q_T(x^{r_i}_{T-1}, \omega_{Ti}) - Q_T(x^k_{T-1}, \omega_{T_i})\right] + \Delta^k.$$

Now,

$$|\Delta^k_1| \leq \sum_{i=1}^{q_T} p_{T_i} |Q_T(x^{r_i}_{T-1}, \omega_{T_i}) - Q_T(x^k_{T-1}, \omega_{Ti})| + |\Delta^k| \to 0, \text{ as } k \to \infty,$$

since the function $Q_T(x_{T-1}, \omega_T)$ is continuous in x_{T-1}, $\{x_{T-1}^k\}_{k\in J}$ is a convergent sequence, and $r_i \to \infty$ as $k \to \infty$. This leads to the following intermediate result:

$$|\theta_T^k - \mathcal{Q}_T(x_{T-1}^k)| \le |\Delta_1^k|, \quad \text{with } \lim_{k\to\infty} |\Delta_1^k| = 0.$$

Furthermore, due to continuity of $\mathcal{Q}_T(x_{T-1})$ in x_{T-1}, we have

$$\lim_{k\to\infty} |\mathcal{Q}_T(x_{T-1}^k) - \mathcal{Q}_T(x_{T-1}^0)| = 0.$$

Therefore,

$$\begin{aligned}|\theta_T^k - \mathcal{Q}_T(x_{T-1}^0)| &\le |\theta_T^k - \mathcal{Q}_T(x_{T-1}^k)| + |\mathcal{Q}_T(x_{T-1}^k) - \mathcal{Q}_T(x_{T-1}^0)| \\ &\le |\Delta_1^k| + |\mathcal{Q}_T(x_{T-1}^k) - \mathcal{Q}_T(x_{T-1}^0)| \\ &\to 0, \quad \text{as } k \to \infty.\end{aligned}$$

Hence, the sequence $\{\theta_T^k\}_{k\in J}$ converges to $\mathcal{Q}_T(x_{T-1}^0)$ with probability 1, which shows part (b).

(c) Considering Lemma 3.3 and Lemma 3.4 again, we have

$$\begin{aligned}Q_{T-1}(x_{T-2}^k, \omega_{T-1}^0) &\ge C_{T-1}^k(x_{T-2}^k, \omega_{T-1}^0) \\ &= c_{T-1}^\top x_{T-1}^k + \theta_T^k \\ &\ge c_{T-1}^\top x_{T-1}^k + \mathcal{Q}_T(x_{T-1}^k) + \Delta_1^k \\ &\ge Q_{T-1}(x_{T-2}^k, \omega_{T-1}^0) + \Delta_1^k,\end{aligned}$$

where the last inequality comes from the fact that x_{T-1}^k is also feasible for problem [LP($T-1$)] with $x_{T-2} = x_{T-2}^k$ and $\omega_{T-1} = \omega_{T-1}^0$. This implies that

$$|C_{T-1}^k(x_{T-2}^k, \omega_{T-1}^0) - Q_{T-1}(x_{T-2}^k, \omega_{T-1}^0)| \le |\Delta_1^k|.$$

Since the function $Q_{T-1}(x_{T-2}, \omega_{T-1})$ is continuous in x_{T-2} and since the sequence $\{x_{T-2}^k\}_{k\in K}$ is convergent in K (hence, also in J),

$$\lim_{k\to\infty} |Q_{T-1}(x_{T-2}^k, \omega_{T-1}^0) - Q_{T-1}(x_{T-2}^0, \omega_{T-1}^0)| = 0.$$

Therefore,

$$\begin{aligned}&|C_{T-1}^k(x_{T-2}^k, \omega_{T-1}^0) - Q_{T-1}(x_{T-2}^0, \omega_{T-1}^0)| \\ &\le |C_{T-1}^k(x_{T-2}^k, \omega_{T-1}^0) - Q_{T-1}(x_{T-2}^k, \omega_{T-1}^0)| \\ &\quad + |Q_{T-1}(x_{T-2}^k, \omega_{T-1}^0) - Q_{T-1}(x_{T-2}^0, \omega_{T-1}^0)| \\ &\le |\Delta_1^k| + |Q_{T-1}(x_{T-2}^k, \omega_{T-1}^0) - Q_{T-1}(x_{T-2}^0, \omega_{T-1}^0)| \to 0, \quad \text{as } k \to \infty.\end{aligned}$$

This means that the sequence $\{C^k_{T-1}(x^k_{T-2}, \omega^0_{T-1})\}_{k\in J}$ converges to Q_{T-1} $(x^0_{T-2}, \omega^0_{T-1})$ wp1, which shows part (c). □

Lemma 4.2. Suppose that MSBD satisfies CSP and SIP. For any given $t, 1 \le t \le T-1$, and any given set $K \subseteq \mathcal{N}$, suppose that:

(i) $\omega^k_t = \omega^0_t$ for some given ω^0_t for any $k \in K$;
(ii) the sequence $\{x^k_{t-1}\}_{k\in K}$ converges to some given vector x^0_{t-1}.

Then, there exists an infinite set $J \subseteq K$ such that:

(a) the sequence $\{x^k_t\}_{k\in J}$ converges to some vector x^0_t;
(b) the sequence $\{\theta^k_{t+1}\}_{k\in J}$ converges to $\mathcal{Q}_{t+1}(x^0_t)$ wp1;
(c) the sequence $\{C^k_t(x^k_{t-1}, \omega^0_t)\}_{k\in J}$ converges to $Q_t(x^0_{t-1}, \omega^0_t)$ wp1.

Proof. The lemma is proved by induction on t. When $t = T-1$, this lemma is exactly Lemma 4.1 and hence holds. Suppose that the lemma holds for t; then, we need to prove it for $t-1$. Therefore, assume now that, for a given set $K \subseteq \mathcal{N}, \omega^k_{t-1} = \omega^0_{t-1}, \forall k \in K$, and $\{x^k_{t-2}\}_{k\in K} \to x^0_{t-2}$.

(a) In iteration $k \in K$, the algorithm solves problem $[\mathrm{AP}(t-1)^k]$, with $x_{t-2} = x^k_{t-2}$ and $\omega_{t-1} = \omega^0_{t-1}$, and gets the solution (x^k_{t-1}, θ^k_t). Since the feasible set is bounded, the sequence $\{x^k_{t-1}\}_{k\in K}$ has a convergent subsequence. Denote the corresponding limit as x^0_{t-1} and the corresponding index set as L.

Now, the set J is constructed in a way that the induction hypothesis can be applied. As shown in Lemma 3.2 for each $i = 1, \dots, q_t$, the set L has wp1 an infinite subsequence L_i such that, $\omega^k_t = \omega_{ti}$ and $\omega_{ti} \in \Omega^k_t$, for all $k \in L_i$.

For each $i = 1, \dots, q_t$, by the induction assumption that the lemma holds for stage t and by the facts that $\omega^k_t = \omega_{ti}$ for all $k \in L_i$ and that the sequence $\{x^k_{t-1}\}_{k\in L}$ [and hence the sequence $\{x^k_{t-1}\}_{k\in L_i}$] converges to some vector x^0_{t-1}, there must exist an infinite subset J_i of L_i, for each $i = 1, \dots, q_t$, such that the sequence $\{C^k_t(x^k_{t-1}, \omega_{ti})\}_{k\in J_i}$ converges to $Q_t(x^0_{t-1}, \omega_{ti})$ wp1. Therefore, with $k \in J_i$,

$$\lim_{k\to\infty} |C^k_t(x^k_{t-1}, \omega_{ti}) - Q_t(x^0_{t-1}, \omega_{ti})| = 0. \tag{2}$$

Define

$$J = \bigcup_{i=1}^{q_t} J_i.$$

Clearly, $J \subseteq L$; hence, the sequence $\{x^k_{t-1}\}_{k\in J}$ converges to x^0_{t-1}.

(b) From Lemma 3.3 and Lemma 3.4, we have that, for $k \in J$,

$$\mathcal{Q}_t(x_{t-1}^k) \geq \theta_t^k \geq \sum_{i=1}^{qt} p_{it} C_t^{r_i}(x_{t-1}^{r_i}, \omega_{ti}) + \Delta^k, \quad \text{with } \lim_{k\to\infty} |\Delta^k| = 0,$$

where the r_i are elements of

$$J_i = J \bigcap \{k | \omega_{ti} \in \Omega_t^k, \omega_t^k = \omega_{ti}\}.$$

This is equivalent to

$$\mathcal{Q}_t(x_{t-1}^k) \geq \theta_t^k \geq \mathcal{Q}_t(x_{t-1}^k) + \sum_{i=1}^{qt} p_{it}[C_t^{r_i}(x_{t-1}^{r_i}, \omega_{ti}) - Q_t(x_{t-1}^k, \omega_{ti})] + \Delta^k,$$

whence

$$|\theta_t^k - \mathcal{Q}_t(x_{t-1}^k)| \leq |\Delta_1^k|,$$

with

$$\Delta_1^k = \sum_{i=1}^{q_t} p_{it}[C_t^{r_i}(x_{t-1}^{r_i}, \omega_{ti}) - Q_t(x_{t-1}^k, \omega_{ti})] + \Delta^k.$$

Now,

$$\begin{aligned} |\Delta_1^k| &\leq \sum_{i=1}^{q_t} p_{it} \; |C_t^{r_i}(x_{t-1}^{r_i}, \omega_{ti}) - Q_t(x_{t-1}^k, \omega_{ti})| + |\Delta^k| \\ &\leq \sum_{i=1}^{q_t} p_{it} \Big\{ |C_t^{r_i}(x_{t-1}^{r_i}, \omega_{ti}) - Q_t(x_{t-1}^0, \omega_{ti})| \\ &\qquad + |Q_t(x_{t-1}^0, \omega_{ti}) - Q_t(x_{t-1}^k, \omega_{ti})| \Big\} + |\Delta^k|. \end{aligned}$$

If $k \to \infty$, then $r_i \to \infty$, and from (2) we have that, for $r_i \in J_i$,

$$\lim_{k\to\infty} |C_t^{r_i}(x_{t-1}^{r_i}, \omega_{ti}) - Q_t(x_{t-1}^0, \omega_{ti})| = 0, \quad \text{wp1}.$$

Furthermore, due to continuity of $Q_t(x_{t-1}, \omega_t)$ in x_{t-1} and due to convergence of the sequence $\{x_{t-1}^k\}_{k\in J} \to x_{t-1}^0$,

$$\lim_{k\to\infty} |Q_t(x_{t-1}^0, \omega_{ti}) - Q_t(x_{t-1}^k, \omega_{ti})| = 0.$$

Therefore, $\lim_{k\to\infty} |\Delta_1^k| = 0$, wp1, and so

$$|\theta_t^k - \mathcal{Q}_t(x_{t-1}^k)| \to 0, \text{ as } k \to \infty, \quad \text{wp1}.$$

Continuity of $\mathcal{Q}_t$ gives

$$\lim_{k\to\infty} |\mathcal{Q}_t(x_{t-1}^k) - \mathcal{Q}_t(x_{t-1}^0)| = 0;$$

therefore, wp1,

$$|\theta_t^k - \mathcal{Q}_t(x_{t-1}^0)| \to 0, \quad \text{as } k \to \infty.$$

Hence, the sequence $\{\theta_t^k\}_{k\in J}$ converges to $\mathcal{Q}_t(x_{t-1}^0)$ wp1, which shows part (b).

(c) Using the same argument as in the proof of Lemma 4.1, we obtain

$$|C_{t-1}^k(x_{t-2}^k, \omega_{t-1}^0) - Q_{t-1}(x_{t-2}^k, \omega_{t-1}^0)| \le |\Delta_1^k|,$$

and by continuity,

$$\lim_{k\to\infty} |Q_{t-1}(x_{t-2}^k, \omega_{t-1}^0) - Q_{t-1}(x_{t-2}^0, \omega_{t-1}^0)| = 0.$$

This yields, wp1,

$$|C_{t-1}^k(x_{t-2}^k, \omega_{t-1}^0) - Q_{t-1}(x_{t-2}^0, \omega_{t-1}^0)| \to 0, \quad \text{as } k \to \infty,$$

which means that the sequence $\{C_{t-1}^k(x_{t-2}^k, \omega_{t-1}^0)\}_{k\in J}$ converges to Q_{t-1} $(x_{t-2}^0, \omega_{t-1}^0)$ with probability 1, which shows part (c). □

Theorem 4.1. Suppose that MSBD satisfies CSP and SIP. The sequence of the solution values $\{C_1^k\}_{k\in\mathcal{N}}$ of problem [AP1^k] converges to Q_1 wp1.

Proof. In the approximated first-stage problem [AP1^k], the constraint $A_1x_1 = b_1$ can be formulated as

$$A_1x_1 = \omega_1 - B_0x_0,$$

with $\omega_1 \equiv b_1, x_0 \equiv 0$, and any given B_0. The value C_1^k can be seen as a trivial function $C_1^k(x_0, \omega_1)$. The result then follows from Lemma 4.2 equivalently to Ref. 3, Theorem 5.1. □

Theorem 4.2. Suppose that MSBD satisfies CSP and SIP. Then, any accumulation point of the sequence $\{x_1^k\}_{k\in\mathcal{N}}$ is an optimal solution of problem [LP1] wp1.

Proof. See Ref. 3, Theorem 5.2. □

5. Convergent Sampling Algorithms

The class of algorithms described in the previous sections is quite general and includes a wide range of approaches. One subclass (which includes CUPPS) follows a forward pass for both getting candidate solutions and generating cuts for the next iteration. An alternative (such as SDDP) generates cuts in a backward pass. This means that, while generating the cuts for stage t in iteration k, the cuts that were generated for stage $t+1$ in iteration k are already taken into account, which may lead to an improvement in the speed of convergence. Under CSP and SIP, the convergence result above remains the same for the backward-pass algorithms, if one observes that the sampled cuts should be modified so that one cut constraint additional to [APt^k] is considered when obtaining the dual variables $(\pi_t^i(x_{t-1}^k), \rho_t^i(x_{t-1}^k))$.

The analysis of the previous section considers algorithms which use only one path of the tree per iteration. The class of algorithms can be extended to the multipath case of n_k sample paths, whereby in iteration k there are n_k paths sampled. If, say, the last of the paths is sampled independently from stage to stage, then this can be thought of as a single iteration of the algorithm with (possibly) extra cuts added on the backward pass.

In fact, the following general multipath scheme is possible: in iteration k, a candidate solution for the first stage is determined and a cut for stage 1 is generated. In the second stage, n_2^k scenarios are sampled. Then, of the n_2^k scenarios, s_2^k are chosen and candidate solutions x_2^k are determined. Of these s_2^k candidate solutions, c_2^k are chosen at which the algorithm will generate cuts for stage 2. For each of the s_2^k candidate solutions, n_3^k samples are considered at stage 3. Then, of the $s_2^k \cdot n_3^k$ samples, s_3^k samples are chosen and of these c_3^k cuts are generated for the stage 3, etc.

Therefore, n_t^k is the number of samples in stage t for each sample s_{t-1}^k of stage $t-1$; s_t^k is the number of samples of which to proceed to the next stage; and c_t^k is the number of cuts generated for stage t. The following relations hold:

$$\begin{aligned} &s^k_{t-1} n^k_t \geq s^k_t \geq c^k_t \geq 0, && \forall t \geq 2, \forall k, \\ &n^k_t \geq 1, && \forall t, \forall k, \\ &s^k_t = 1, && \forall k. \end{aligned}$$

Observe that not all the possible choices of these parameters satisfy the condition CSP. For example, if for some $t \geq 2$,

$$c^k_t = 0, \quad \forall k,$$

then there is no guarantee of convergence.

We conclude this section by showing how MSBD algorithm from the literature fit into this framework. The results are summarized in Table 1.

Stochastic Dual Dynamic Programming (SDDP). This algorithm was introduced in Ref. 1. In SDDP, n scenario paths are sampled in each iteration. In a forward pass, for each stage in each scenario, a candidate solution is calculated by solving [APt^k]. Then, in a backward pass, in each stage t the entire single-period subtree ($\Omega^k_t = \Omega_t$) is solved and a cut is generated for stage $t-1$. Thus, SDDP is a multipath scheme with $n_k = n, \forall k$.

Convergent Cutting-Plane and Partial Sampling (CUPPS). This algorithm is given in Ref. 3. In each iteration, it samples only one scenario, ($\Omega^k_t = \{\omega^k_t\}$). Both calculating candidate solutions and generating cuts are performed in the forward pass.

Abridged Nested Decomposition (AND). This algorithm is described in Ref. 4. As in SDDP, $\Omega^k_t = \Omega_t$, and, like SDDP, it involves sampling in the forward pass, but the main difference is that AND does not proceed forward from all solutions of the realizations sampled in each stage. Instead, in each stage, n^k_t successors are sampled as in the general multipath scheme. Of these nodes, $s^k_t \leq n^k_t$ nodes are sampled from which to proceed and $c^k_t = s^k_t, \forall k, \forall t$.

Table 1. Examples of convergent algorithms.

Algorithm	Forward/Backward	Sampled cut	Multipath scheme
SDDP	B	$\Omega^k_t = \Omega_t$	$n_k = n \ \forall k$
CUPPS	F	$\Omega^k_t = \{\omega^k_t\}$	–
AND	B	$\Omega^k_t = \Omega_t$	$s^k_t = c^k_t \ \forall k \ \forall t$
ReSa	B	$\Omega^k_t = \Omega_t$	$n^k_t = s^k_t \ \forall k \ \forall t$

Reduced Sampling Method (ReSa). This algorithm was developed in Ref. 5. The basic structure is the same as in SDDP. First, some scenarios of the tree are sampled. The difference from SDDP lies in the backward pass. In ReSa, in each stage the subtrees are solved to generate a cut for only some randomly chosen scenarios. Therefore, fewer subproblems have to be solved than for SDDP, but one gets also fewer cuts per iteration. ReSa is a general multipath scheme, with $n_t^k = s_t^k \geq c_t^k$. The cuts are generated by solving entire single-period subtrees ($\Omega_t^k = \Omega_t$). If $c_t^k > 0$, for all t and all k sufficiently large, then ReSa satisfies the cut-sampling property and so converges wp1.

6. Conclusions

This paper presents a general convergence result for multistage stochastic Benders decomposition codes that use sampling. Although we make no assertions in this paper about the rate of convergence of these algorithms, this theory provides some guidance for researchers who select parameters to tune these algorithms. To ensure convergence wp1 in ReSa, for example, one should ensure in accordance with the cut-sampling property that the algorithm eventually computes at least one cut for each stage in the backward pass.

A key restriction on the algorithms that we study is the sample-intersection property, which guarantees that some proportion of random outcomes obtained by sampling moving forward in time are chosen for cut calculation. It is not hard to see why such a condition might be needed. Certainly, one can conceive of (perverse) algorithms that compute only cuts when the x_t^k values lie in certain subsets of the feasible region [AP1^k]. Since the optimal solution might have x_t lying outside this subset, there is no reason to suppose that the algorithm would converge wp1 even if the subsets in which cuts are computed are visited infinitely often.

The convergence proof above uses a bound on the optimal dual variables for [APt^k], that comes from their construction as extreme-point solutions. A possible extension is to allow the calculation of cuts for $\omega_{ti} \in \Omega_t^k$ to be inexact, in the sense that the dual variables (π_t^k, ρ_t^k) are computed to be within ϵ_t^k of optimally (see Ref. 8). Under the assumption of a bounded dual feasible region for each problem [APt^k], it is easy to extend our results to show convergence under CSP and SIP wp1 if $\epsilon_t^k \to 0$ for each t.

References

1. Pereira, M. V. F., and Pinto, L. M. V. G., *Multistage Stochastic Optimization Applied to Energy Planning*, Mathematical Programming, Vol. 52, pp. 359–375, 1991.
2. Power Systems Research, Homepage. See http://www.psr-inc.com.br/sddp.asp.
3. Chen, Z. L., and Powell, W. B., *Convergent Cutting Plane and Partial-Sampling Algorithm for Multistage Stochastic Linear Programs with Recourse*, Journal of Optimization Theory and Applications, Vol. 102, pp. 497–524, 1999.
4. Donohue, C. J., *Stochastic Network Programming and the Dynamic Vehicle Allocation Problem*, PhD Dissertation, University of Michigan, Ann Arbor, Michigan, 1996.
5. Hindsberger, M., and Philpott, A. B., *ReSa: A Method for Solving Multistage Stochastic Linear Programs*, European Journal of Operations Research (submitted).
6. Birge, J. R., and Louveaux, F., *Introduction to Stochastic Programming*, Springer Verlag, New York, NY, 1997.
7. Grimmett, G. R., and Stirzaker, D. R., *Probability and Random Processes*, Oxford University Press, Oxford, UK, 1992.
8. Zakeri, G., Philpott, A. B., and Ryan, D. M., *Inexact Cuts in Stochastic Benders Decomposition*, SIAM Journal on Optimization, Vol. 10, pp. 643–657, 2000.