

Learning Algorithms for Separable Approximations of Stochastic Optimization Problems

Warren Powell*, Andrzej Ruszczyński†, Huseyin Topaloglu‡

July 29 2002

Abstract

We propose the use of sequences of separable, piecewise linear approximations for solving classes of nondifferential stochastic optimization problems. The approximations are estimated adaptively using a combination of stochastic gradient information and possibly sample information on the objective function itself. We prove the convergence of several versions of such methods when the objective function is separable and illustrate their behavior on numerical examples. We then demonstrate the performance on nonseparable problems that arise in the context of two-stochastic stochastic programming problems, and demonstrate that these techniques provide near optimal solutions with a very fast rate of convergence compared to other solution techniques.

1 Introduction

We consider the following stochastic programming problem

$$\max_{x \in X} \mathbb{E}f(x, \omega), \quad (1)$$

where $f : \mathbb{R}^n \times \Omega \rightarrow \mathbb{R}$, $(\Omega, \mathcal{F}, \mathbb{P})$ is a probability space, and \mathbb{E} denotes the expected value. We assume that the function f is concave with respect to the first argument, and such that

$$F(x) = \mathbb{E}f(x, \omega)$$

is finite for every $x \in X$. We assume that for almost all ω , all $i = 1, \dots, m$, and for all feasible integer values of x_j for $j \neq i$, the function $f(x_1, \dots, x_{i-1}, \cdot, x_{i+1}, \dots, x_n, \omega)$ is piecewise linear with integer break points. We also assume that the set X is closed, and

$$X \subseteq \{x \in \mathbb{R}^n : 0 \leq x_i \leq M_i, i = 1, \dots, n\}.$$

*Department of Operations Research and Financial Engineering, Princeton University, Princeton, NJ, 08544

†Department of Management Science and Information Systems, Rutgers University, 94 Rockefeller Rd, Piscataway, NJ 08854

‡Department of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY

Problems of this type arise in a variety of resource allocation problems. In the car distribution problem of railroads, for example, planners often face the problem of having to reposition empty cars before a customer order has been realized. Truckload carriers have to assign drivers to loads before knowing the opportunities available to the driver at the destination. The air mobility command needs logic for their simulators that will reposition aircraft back to locations where they *might* be needed, before actual demands are known.

All of these problems can be modeled as multistage linear programs. Experimental research (see, for example, Godfrey & Powell (2001)) demonstrate that the recourse function for these problems can be well approximated by sequences of piecewise linear, separable functions. These scalar functions can be estimated by techniques that use sample gradients, where the challenge is to maintain the natural concavity of the underlying problem (it is very important that we solve sequences of concave approximations). As a result of the need to use Monte Carlo samples, it is necessary to introduce steps which maintain the concavity after *every* update. This paper explores a class of such techniques and establishes conditions under which these approximations converge to the true function.

An important special case of (1) is what is referred to as the two-stage stochastic program:

$$\begin{aligned} \max \quad & c^t x + \mathbb{E} Q(s, \omega) \\ \text{subject to:} \quad & Ax = b \\ & Tx - s = 0 \\ & x, s \geq 0 \end{aligned}$$

where,

$$\begin{aligned} Q(s, \omega) = \max \quad & q^t y \\ \text{subject to:} \quad & Wy = h(\omega) + s \\ & y \geq 0. \end{aligned}$$

Another special case occurs when f in (1) is a separable function of the form

$$f(x, \omega) = \sum_{i=1}^n f_i(x_i, \omega). \tag{2}$$

This is the form most often taken in the context of classical resource allocation problems (Righter (1989)) which involve the allocation of resources to independent activities subject to a common budget constraint.

In order to obtain the optimal solution to (1), one can consider building sequential approximations of F , say F^k . If the sequence $\{F^k\}$ converges to F in some sense then we can claim to have a procedure to solve (1). Alternatively we may solve optimization problems of the form $x^k \in \arg \max_{x \in X} F^k(x)$ such that the sequence $\{x^k\}$ converges to $x^* \in \arg \max_{x \in X} \bar{F}(x)$. In this case the sequence of functions $\{F^k\}$ does not necessarily converge to \bar{F} , but $\{x^k\}$ may converge to optimal or near-optimal points. Of critical

importance in practical applications is also the speed of convergence, a question that we treat on an experimental basis.

For our class of applications, it is relatively easy, for a given x^k , to sample an elementary event ω^k and to calculate $f_i(x_i^k, \omega^k)$. Moreover, it is also easy to obtain information about the slope of $f_i(\cdot, \omega^k)$ at x_i^k :

$$v_i^k = f_i(x_i^k, \omega^k) - f_i(x_i^k - 1, \omega^k). \quad (3)$$

On the other hand, it is difficult to obtain the exact values of $\bar{f}_i(x^k)$, since it involves the calculation of the expected value.

Example 1.1 Let x_i denote the amount of resource allocated to activity i , where $i = 1, \dots, n$. These amounts have to be chosen from the set

$$X = \left\{ x \in \mathbb{R}^n : x_i \in \{0, 1, \dots, M_i\}, i = 1, \dots, n, \sum_{i=1}^n x_i \leq b \right\}.$$

For each activity i there is a nonnegative integer random variable D_i representing the demand. The reward associated with activity i is defined as in the newsvendor problem:

$$f_i(x_i, D_i) = q_i \min(x_i, D_i) - c_i x_i,$$

where $q_i > c_i > 0$. Our objective is to allocate the resources in such a way that the expected reward, $F(x) = \mathbb{E} \sum_{i=1}^n f_i(x_i, D_i)$, is maximized, subject to the constraint $x \in X$. Porteus (1990) provides a thorough review of the newsvendor problem in the context of stochastic inventory models. The optimal solution of a single newsvendor problem can be expressed analytically, but this requires knowing the distribution of demand. An extensive literature has evolved to solve what is known as the censored newsvendor problem (where you only see the amount sold, not the actual demand). This literature (see, for example, Ding, Puterman & Bisi (2002)) requires assuming a parametric form for the demand distribution. In this paper, we provide an algorithm that provides an asymptotically optimal solution to the censored newsvendor problem without requiring any particular form for the demand distribution.

In such a problem we may sample a demand realization $D^k = (D_1^k, \dots, D_n^k)$ and calculate

$$v_i^k = \begin{cases} q_i - c_i & \text{if } x_i^k \leq D_i^k, \\ -c_i & \text{if } x_i^k > D_i^k. \end{cases} \quad i = 1, \dots, n.$$

In this case we can also calculate the right slope estimate

$$v_{i+}^k = \begin{cases} q_i - c_i & \text{if } x_i^k < D_i^k, \\ -c_i & \text{if } x_i^k \geq D_i^k, \end{cases} \quad i = 1, \dots, n.$$

Similar estimates can be generated in a slightly more complicated case, with the reward associated with activity i defined as

$$f_i(x_i, D_i) = q_i(\min(x_i, D_i)) - c_i(x_i),$$

where $q_i(\cdot)$ is a concave piecewise linear function, and $c_i(\cdot)$ is a convex piecewise linear function, both with break points at $0, 1, \dots, M_i$.

There exists a wealth of numerical methods for stochastic programming problems. The first group are scenario methods, in which a sufficiently rich sample $\omega^1, \dots, \omega^N$ is drawn from the space Ω , and the expectation is approximated by the sample average:

$$F^N(x) = \frac{1}{N} \sum_{\nu=1}^N \sum_{i=1}^n f_i(x_i, \omega^\nu).$$

A discussion of these approaches can be found in Shapiro & Homem-De-Mello (2000) and Kleywegt, Shapiro & Homem-De-Mello (2002).

The second group of methods are *stochastic subgradient methods*, which use the fact that the random vector v^k in (3) satisfies the relation

$$\mathbb{E}\{v^k | x^k\} \in \partial F(x^k),$$

where ∂F is the subdifferential of F (understood as the negative of the subdifferential of the convex function $-F$). Stochastic gradient algorithms depend on updates of the form:

$$x^{k+1} = x^k + \alpha_k \partial F(x^k)$$

These methods make very many small updates to the current approximation by using the stochastic subgradients as directions, achieving convergence almost surely to the optimal point. A treatment of these methods can be found in Ermoliev (1988). Constraints can be treated either by projection or by recursive linearization (Ruszczynski (1987)).

Finally, Benders decomposition can be used to solve two-stage stochastic linear programs (a problem class which we specifically address in this paper) by replacing the expectation with a series of cuts (Van Slyke & Wets (1969), Hige & Sen (1991)).

Our problem class is motivated by problems that require integer solutions. The introduction of uncertainty often has the effect of destroying the natural integer structure of many problems. All three of these classes of techniques destroy the natural integrality of these problems, either by how the original problem is approximated (scenario methods and Benders decomposition) or the nature of the algorithm itself (stochastic gradient methods). Laporte & Louveaux (1993) show how integrality can be incorporated into Benders decomposition.

We propose to solve these problems by adaptively estimating, using sample gradient information, sequences of *separable* approximations which are piecewise linear and concave. We use the information gathered in iterations $1, \dots, k$ to construct models $\bar{f}_i^k(\cdot)$ of \bar{f}_i , $i = 1, \dots, n$. The basic form of the algorithm is given by:

$$x^k = \arg \max_{x \in X} \sum_{i=1}^n \bar{f}_i^k(x_i), \quad (4)$$

to generate the next approximate solution, x^{k+1} . An associated learning step provides information employed to update the models \bar{f}_i^k . Such an approach is already known to be optimal if the problem is continuously differentiable (Culioli & Cohen (1990), Cheung & Powell (2000)), but there is no comparable result for nondifferentiable problems. While

the relation of stochastic approximation type methods and learning is well known (see, e.g. Kushner & Yin (1997)), the use of the structure (separability and concavity) allows here for the construction of particularly efficient methods.

Our solution strategy extends a line of research in stochastic resource allocation using separable approximations. Frantzeskakis & Powell (1990) suggest a static, piecewise linear separable approximation for specially structured tree problems, a result that is generalized in Powell & Cheung (1994) and applied to multistage resource allocation problems in Cheung & Powell (1996). These methods, however, were not adaptive which limited the quality of the solution. Powell & A. (1998) provided an adaptive learning algorithm based on linear approximations, which was extended in Godfrey & Powell (2001) to an adaptive, piecewise linear approximation based on the “CAVE” algorithm. The CAVE algorithm provided exceptionally good experimental performance, but offered no provable results. Wallace (1987) introduces a piecewise linear upper bound for networks, a result that is generalized in Birge & Wallace (1988) for stochastic programs.

In this paper, we introduce and study, apparently for the first time, the use of sequences of piecewise linear, separable approximations as a strategy for solving nondifferentiable stochastic optimization problems. As a byproduct, we produce a fast algorithm for problems such as two stage stochastic programs with network recourse, a topic that was first studied in depth by Wallace (1986). We establish several important convergence results for the special case of separable objective functions, and show experimentally that the algorithm provides near-optimal, and often optimal, solutions for problems when the objective function is not separable, as would be the case with two-stage stochastic programs. Furthermore, the speed of convergence is much faster than techniques such as Benders decomposition, especially for higher dimensional problems.

The paper is divided into two parts. Sections 2-7 deal exclusively with problems where the original objective function $F(x, \omega)$ is separable. While this problem class enjoys its own sets of applications (for example, in a variety of budget allocation problems), our interest in this special problem class arises primarily because we are able to prove some important convergence results. Section 2 presents the basic algorithm for learning piecewise linear, concave approximations (while maintaining concavity after every update), and proves convergence to the real function assuming that all intervals are sampled infinitely often. Section 3 provides a variation of the algorithm that combines gradient information with sample information on the function itself. In practical applications, we cannot generally guarantee that we will sample all points infinitely often, and this is not necessary to find the optimal solution. Section 4 provides a key convergence result under a nonuniform learning pattern, where some portions of the function may not be sampled infinitely often. This result is then used in section 5 which proves convergence when we only sample the points x^k generated by equation (4). Section 6 shows how the projection step can be implemented, and section 7 provides the results of a series of experiments that investigate the rate of convergence of variations of the algorithm.

The second part of the paper, given in section 8, focuses on nonseparable problems that arise in the context of two-stage stochastic programs. We cannot guarantee that our algorithm will produce the optimal solution for two-stage problems, but we show that the right separable approximation can produce the optimal solution, and use this to develop

a bound on our result. Numerical comparisons with Benders decomposition, which are optimal for noninteger versions of these problems, demonstrate that our approach provides much faster convergence and optimal or very near-optimal results.

2 Learning Concave Functions of One Variable

We start from the description and analysis of the basic learning algorithm for a concave piecewise-linear function of one variable $\bar{f} : [0, M] \rightarrow \mathbb{R}$. We assume that \bar{f} is linear on the intervals $[s - 1, s]$, $s = 1, \dots, M$. Let

$$\bar{v}_s = \bar{f}(s) - \bar{f}(s - 1), \quad s = 1, \dots, M.$$

Let us note that the knowledge of the vector \bar{v} allows us to reconstruct $\bar{f}(x)$, $x \in [0, M]$, except for the constant term $\bar{f}(0)$:

$$\bar{f}(x) = \bar{f}(0) + \sum_{s=1}^l \bar{v}_s + \bar{v}_{l+1}(x - l),$$

where l is such that $l \leq x < l + 1$. Thus \bar{f} can be used as an estimate of the objective function.

The main idea of the algorithm is to recursively update a vector $v^k \in \mathbb{R}^M$, $k = 1, 2, \dots$, in order to achieve convergence of v^k to \bar{v} (in some stochastic sense). With no confusion, we hope, we still denote by $(\Omega, \mathcal{F}, \mathbb{P})$ the probability space on which this sequence is defined. We use \mathcal{F}_k to denote the σ -subalgebra generated by v^1, \dots, v^k .

Let us note that by the concavity of \bar{f} the vector \bar{v} has nonincreasing components:

$$\bar{v}_{s+1} \leq \bar{v}_s, \quad s = 1, \dots, M - 1. \quad (5)$$

We shall at first assume that there exists a constant B such that

$$\bar{v}_1 \leq B, \quad \bar{v}_M \geq -B. \quad (6)$$

Clearly, the set V of vectors satisfying (5)–(6) is convex and closed. We shall therefore ensure that all our approximate slopes v^k are elements of V as well. To this end we shall employ the operation of orthogonal projection on V

$$\Pi_V(z) = \arg \min\{\|v - z\|^2 : v \in V\}. \quad (7)$$

We show in Section 6 that for the set V defined by (5) such a projection can be calculated in an easy way.

Our learning algorithm has the following form.

Separable, projective approximation routine (SPAR)

Step 0 Set $v^1 \in V$, $k = 1$.

Step 1 Choose $s^k \in \{1, \dots, M\}$.

Step 2 Observe a random variable η^k such that

$$\mathbb{E}\{\eta^k \mid v^1, \dots, v^k, s^k\} = \bar{v}_{s^k}, \text{ a.s.} \quad (8)$$

Step 3 Calculate the vector $z^k \in \mathbb{R}^M$ as follows

$$z_s^k = \begin{cases} (1 - \alpha_k)v_s^k + \alpha_k\eta^k & \text{if } s = s^k, \\ v_s^k & \text{otherwise,} \end{cases} \quad (9)$$

where $\alpha_k \in (0, 1]$.

Step 4 Calculate

$$v^{k+1} = \Pi_V(z^k),$$

increase k by one and go to Step 1.

At this moment we shall not specify the way in which s^k is defined, except that s^k is a random variable. Specific conditions on it will be formulated later.

We shall use the notation

$$p_s^k = \mathbb{P}\{s^k = s \mid \mathcal{F}_k\}, \quad s = 1, \dots, M.$$

The stepsizes α_k employed at Step 3 may be also be random, but must be measurable with respect to the σ -subalgebra generated by v^1, \dots, v^k and s^1, \dots, s^k .

Let us denote by ξ^k the random vector with the components

$$\xi_s^k = \begin{cases} -\eta^k + v_s^k & \text{if } s = s^k, \\ 0 & \text{otherwise.} \end{cases}$$

We can now rewrite the method compactly as

$$v^{k+1} = \Pi_V(v^k - \alpha_k \xi^k), \quad k = 1, 2, \dots$$

It follows from (8) that

$$\mathbb{E}\{\xi_s^k \mid \mathcal{F}_k\} = p_s^k(v_s^k - \bar{v}_s), \quad s = 1, \dots, M.$$

Thus

$$\mathbb{E}\{\xi^k \mid \mathcal{F}_k\} = P^k(v^k - \bar{v}), \quad P^k = \text{diag}(p_s^k)_{s=1}^M, \quad (10)$$

so ξ^k is a (scaled) stochastic gradient of

$$W(v) = \frac{1}{2} \|v - \bar{v}\|^2.$$

By the nonexpansiveness of the projection $\Pi_V(\cdot)$,

$$\begin{aligned}\|v^{k+1} - \bar{v}\|^2 &\leq \|z^k - \bar{v}\|^2 \\ &= \|v^k - \bar{v}\|^2 - 2\alpha_k \langle \xi^k, v^k - \bar{v} \rangle + \alpha_k^2 \|\xi^k\|^2.\end{aligned}\quad (11)$$

We assume that there exists a constant C such that for all k

$$\mathbb{E}\{\|\eta^k\|^2 \mid v^1, \dots, v^k, s^k\} \leq C, \quad \text{a.s.}, \quad k = 1, 2, \dots \quad (12)$$

We also assume that

$$\sum_{k=1}^{\infty} \alpha_k = \infty, \quad \text{a.s.}, \quad (13)$$

$$\sum_{k=1}^{\infty} \mathbb{E}\alpha_k^2 < \infty, \quad (14)$$

$$\liminf_{k \rightarrow \infty} p_s^k > 0, \quad \text{a.s.}, \quad s = 1, \dots, M. \quad (15)$$

Theorem 2.1 *Assume (8) and (12)–(15). Then SPAR generates a sequence $\{v^k\}$ such that $v^k \rightarrow \bar{v}$ a.s..*

Proof. Our proof is standard, but we present it here in order to derive some useful inequalities that will be applied later. We have

$$\begin{aligned}\|v^{k+1} - \bar{v}\|^2 &\leq \|v^k - \bar{v}\|^2 - 2\alpha_k \langle v^k - \bar{v}, P^k(v^k - \bar{v}) \rangle \\ &\quad + 2\alpha_k \langle v^k - \bar{v}, \xi^k - P^k(v^k - \bar{v}) \rangle + \alpha_k^2 \|\xi^k\|^2.\end{aligned}\quad (16)$$

By the boundedness of V , and by (12) and (14),

$$\sum_{k=1}^{\infty} \alpha_k^2 \|\xi^k\|^2 < \infty, \quad \text{a.s.},$$

as a convergent submartingale. By the same three conditions and (10) the series

$$\sum_{k=1}^{\infty} \alpha_k \langle v^k - \bar{v}, \xi^k - P^k(v^k - \bar{v}) \rangle$$

is a martingale, which is convergent a.s. Thus (16) may be rewritten as

$$\|v^{k+1} - \bar{v}\|^2 \leq \|v^k - \bar{v}\|^2 - 2\alpha_k \langle v^k - \bar{v}, P^k(v^k - \bar{v}) \rangle + A_k. \quad (17)$$

where $\sum_{k=1}^{\infty} A_k$ is finite a.s. Therefore the sequence $\{\|v^k - \bar{v}\|^2\}$ is convergent a.s. and

$$\sum_{k=1}^{\infty} \alpha_k \langle v^k - \bar{v}, P^k(v^k - \bar{v}) \rangle < \infty, \quad \text{a.s..}$$

Using (13) and (15) we deduce that a.s. there must exist a subsequence $\{v^k\}$, $k \in \mathcal{K}$, such that $v^k \rightarrow \bar{v}$ for $k \in \mathcal{K}$. Since $\|v^k - \bar{v}\|^2$ converges, the entire sequence $\{v^k\}$ is convergent to \bar{v} . \square

If we remove inequalities (6) from the definition of V , only small technical changes are needed to ensure convergence a.s. Instead of the steps $\alpha_k \xi^k$ we need to use normalized steps $\alpha_k \gamma_k \xi^k$, where the normalizing coefficients have the form:

$$\gamma_k = (\max(\|v^k\|, B))^{-1},$$

for some large constant B . We first prove that both martingales converge, due to the damping by the γ_k 's. Then the corresponding version of (17) yields the boundedness of $\{v^k\}$ a.s. Consequently, the normalizing coefficients are bounded away from 0, a.s., and the remaining part of the analysis goes through, as well.

In many applications, at a given point $s^k \in \{1, \dots, M-1\}$ we can observe *two* random variables: η^k satisfying (8), and η_+^k such that

$$\mathbb{E}\{\eta_+^k \mid v^1, \dots, v^k, s^k\} = \bar{v}_{s^k+1} \quad (18)$$

and

$$\mathbb{E}\{\|\eta_+^k\|^2 \mid v^1, \dots, v^k, s^k\} \leq C, \quad k = 1, 2, \dots \quad (19)$$

This was illustrated in Example 1.1.

Our algorithm can be easily adapted to this case, too. The only difference is Step 3, where we use both random observations, whenever they are available:

$$z_s^k = \begin{cases} (1 - \alpha_k)v_s^k + \alpha_k \eta^k & \text{if } s = s^k, \\ (1 - \alpha_k)v_s^k + \alpha_k \eta_+^k & \text{if } s^k < M \text{ and } s = s^k + 1, \\ v_s^k & \text{otherwise,} \end{cases} \quad (20)$$

The analysis of this version of the method is similar to the basic case. We define

$$\xi_s^k = \begin{cases} -\eta^k + v_s^k & \text{if } s = s^k, \\ -\eta_+^k + v_s^k & \text{if } s^k < M \text{ and } s = s^k + 1, \\ 0 & \text{otherwise.} \end{cases}$$

It follows from (8) and (18) that

$$\mathbb{E}\{\xi_s^k \mid \mathcal{F}_k\} = \begin{cases} p_s^k(v_s^k - \bar{v}_s) & \text{if } s = 1 \\ (p_s^k + p_{s-1}^k)(v_s^k - \bar{v}_s) & \text{if } 1 < s \leq M. \end{cases} \quad (21)$$

Therefore, after replacing the coefficients p_s^k by

$$\bar{p}_s^k = \begin{cases} p_s^k & \text{if } s = 1 \\ p_s^k + p_{s-1}^k & \text{if } 1 < s \leq M, \end{cases}$$

we can reduce this version of the method to the basic case analyzed earlier.

3 Using Objective Value Observations

In the applications that we have in mind, our observations provide us with more information than just the estimate of the slope of the objective at s^k . We also observe

$$\theta^k = f(s^k, \omega^k)$$

and we know the value $\bar{f}(0)$. With no loss of generality we may assume $\bar{f}(0) = 0$. Then,

$$\mathbb{E}\{\theta^k | v^1, \dots, v^k, s^k\} = \sum_{i=1}^{s^k} \bar{v}_i \quad (22)$$

and we can use this information to facilitate the convergence to \bar{v} .

SPAR with objective function updates (SPAR-Obj)

Step 0 Set $v^1 \in V$, $k = 1$.

Step 1 Choose $s^k \in \{1, \dots, M\}$.

Step 2 Observe random variables η^k and θ^k satisfying (8) and (22).

Step 3 Calculate the vector $z^k \in \mathbb{R}^M$ as follows

$$z_s^k = \begin{cases} v_s^k + \alpha_k \left(\theta^k - \sum_{i=1}^{s^k} v_i^k \right) & \text{for } s = 1, \dots, s^k - 1, \\ (1 - \alpha_k)v_s^k + \alpha_k \eta^k + \alpha_k \left(\theta^k - \sum_{i=1}^{s^k} v_i^k \right) & \text{for } s = s^k, \\ v_s^k & \text{otherwise,} \end{cases} \quad (23)$$

where $\alpha_k \in (0, 1]$.

Step 4 Calculate

$$v^{k+1} = \Pi_V(z^k),$$

increase k by one and go to Step 1.

We additionally assume that there exists a constant C such that

$$\mathbb{E}\{(\theta^k)^2 | v^1, \dots, v^k, s^k\} \leq C, \quad k = 1, 2, \dots \quad (24)$$

We have a result similar to Theorem 2.1.

Theorem 3.1 *Assume (8), (12)–(15) and (22), (24). Then SPAR-Obj generates a sequence $\{v^k\}$ such that $v^k \rightarrow \bar{v}$ a.s..*

Proof. Let us calculate the conditional expectation of the vector $\xi^k = (v^k - z^k)/\alpha_k$. Directly from (8) and (22) we have

$$\left(\mathbb{E}\{\xi^k | v^1, \dots, v^k, s^k\}\right)_s = \begin{cases} \sum_{i=1}^{s^k} (v_i^k - \bar{v}_i^k) & \text{for } s = 1, \dots, s^k - 1, \\ v_{s^k}^k - \bar{v}_{s^k}^k + \sum_{i=1}^{s^k} (v_i^k - \bar{v}_i^k) & \text{for } s = s^k, \\ 0 & \text{for } s = s^k + 1, \dots, M. \end{cases}$$

Therefore

$$\begin{aligned} (\mathbb{E}\{\xi^k | \mathcal{F}_k\})_j &= p_j^k (v_j^k - \bar{v}_j) + \sum_{s=j}^M p_s^k \sum_{i=1}^s (v_i^k - \bar{v}_i^k) \\ &= \sum_{i=1}^{j-1} \left(\sum_{s=j}^M p_s^k\right) (v_i^k - \bar{v}_i^k) + (p_j^k + \sum_{s=j}^M p_s^k) (v_j^k - \bar{v}_j) + \sum_{i=j+1}^M \left(\sum_{s=i}^M p_s^k\right) (v_i^k - \bar{v}_i^k). \end{aligned}$$

Consider the matrix W^k with the entries

$$w_{ij}^k = \begin{cases} \sum_{s=j}^M p_s^k & \text{if } i \leq j, \\ \sum_{s=i}^M p_s^k & \text{if } i > j. \end{cases}$$

Clearly,

$$\mathbb{E}\{\xi^k | \mathcal{F}_k\} = (P^k + W^k)(v^k - \bar{v}^k), \quad (25)$$

where $P^k = \text{diag}(p_j^k)_{j=1}^M$. We have

$$W^k = \sum_{l=1}^M W_l^k,$$

where each of the matrices W_l^k has the entries

$$(W_l^k)_{ij} = \begin{cases} p_l^k & \text{if } i, j \leq l, \\ 0 & \text{otherwise} \end{cases}$$

Each W_l^k is positive semidefinite, then W^k is positive semidefinite, too. Now we can proceed as in the proof of Theorem 2.1. We have an inequality similar to (16):

$$\begin{aligned} \|v^{k+1} - \bar{v}\|^2 &\leq \|v^k - \bar{v}\|^2 - 2\alpha_k \langle v^k - \bar{v}, (P^k + W^k)(v^k - \bar{v}) \rangle \\ &\quad + 2\alpha_k \langle v^k - \bar{v}, \xi^k - (P^k + W^k)(v^k - \bar{v}) \rangle + \alpha_k^2 \|\xi^k\|^2. \end{aligned}$$

Using (25) and (14) we can rewrite it as

$$\|v^{k+1} - \bar{v}\|^2 \leq \|v^k - \bar{v}\|^2 - 2\alpha_k \langle v^k - \bar{v}, (P^k + W^k)(v^k - \bar{v}) \rangle + A_k.$$

where $\sum_{k=1}^{\infty} A_k$ is finite a.s. Since W^k is positive semidefinite, we can just omit it in the above inequality and obtain (17) again. The rest of the proof is the same as before. \square

SPAR-Obj can be modified to assign different weights to the components of the direction associated with the observations η^k and θ^k . In particular, we may set at Step 3

$$z_s^k = \begin{cases} v_s^k + \frac{\alpha_k}{\rho_k} \left(\theta^k - \sum_{i=1}^{s^k} v_i^k \right) & \text{for } s = 1, \dots, s^k - 1, \\ (1 - \alpha_k) v_s^k + \alpha_k \eta^k + \frac{\alpha_k}{\rho_k} \left(\theta^k - \sum_{i=1}^{s^k} v_i^k \right) & \text{for } s = s^k, \\ v_s^k & \text{otherwise.} \end{cases} \quad (26)$$

where $0 < \rho^{\min} \leq \rho_k \rho^{\max}$, and ρ_k is \mathcal{F}_k -measurable. The analysis of this version is identical to the proof of Theorem 3.1. Our numerical experiments reported in Section 7 indicate that the additional scaling in (26) is useful.

4 Nonuniform Learning Patterns

Our analysis so far was dependent on condition (15), which requires that (asymptotically) the probability of sampling each $s \in \{1, \dots, M\}$ as s^k at iteration k is bounded away from zero. In practical problems, this is a major assumption. For example, in two-stage stochastic programs, we might solve the approximate problem in equation (4) which then determines the next point to be sampled. We do not want to directly control the sequence of points that we are sampling.

For these problems, we are not interested in finding good estimates of all the slopes of the function, but rather only those in the vicinity of the optimal solution. Thus, we recognize that there are points that will not be sampled infinitely often. Our interest in this section is establishing convergence for the points which are sampled infinitely often.

Definition 4.1 We shall say that coordinate s satisfies the *probing condition* if for every $\epsilon > 0$ there exist $\delta > 0$ and k_0 such that for all $k \geq k_0$ and all $l \geq k$ we have

$$\left[\sum_{i=k}^l \alpha_i \geq \epsilon \right] \Rightarrow \left[\sum_{i=k}^l \alpha_i p_s^i \geq \delta \right]. \quad (27)$$

We shall analyze the behavior of SPAR under the condition that some (or all) coordinates satisfy the probing condition. The analysis of SPAR-Obj is similar.

Theorem 4.2 *Assume (8) and (12)–(14). Then SPAR generates a sequence $\{v^k\}$ such that, $v_s^k \rightarrow \bar{v}_s$ a.s. for every $s \in \{1, \dots, M\}$ satisfying the probing condition (27).*

Proof. Proceeding exactly as in the proof of Theorem 2.1 we see that inequality (17) is still valid. Therefore the sequence $\{\|v^k - \bar{v}\|^2\}$ is convergent a.s.

Our proof will analyze the properties of the sample paths of the random sequence $\{v^k\}$ for fixed elementary events $\omega \in \Omega \setminus \Omega_0$, where Ω_0 is a null set. It will become clear in the course of the proof what this null set is.

Let us now fix $\omega \in \Omega$ and suppose that there exists a subsequence $\{v^k(\omega)\}_{k \in \mathcal{K}(\omega)}$ for which $v_s^k(\omega) \xrightarrow{k \in \mathcal{K}(\omega)} v_s'(\omega) \neq \bar{v}_s$. We shall derive a contradiction from this assumption. From now on, for brevity, we shall omit the elementary event ω as an argument in various quantities below. The remaining part of our proof has three stages.

Stage 1: For each $\epsilon > 0$ define

$$l(k, \epsilon) = \inf\{i \geq k : \|v^i - v'\| > \epsilon\}.$$

We shall show that if ϵ is small enough, then $l(k, \epsilon) < \infty$ for all sufficiently large $k \in \mathcal{K}$. Inequality (17) implies that

$$\|v^{i+1} - \bar{v}\|^2 \leq \|v^i - \bar{v}\|^2 - 2\alpha_i p_s^i (v_s^i - \bar{v}_s)^2 + A_i, \quad i = 1, 2, \dots$$

Summing up these inequalities from $k \in \mathcal{K}$ to any $m \geq k$ we obtain

$$\|v^{m+1} - \bar{v}\|^2 \leq \|v^k - \bar{v}\|^2 - 2 \sum_{i=k}^m \alpha_i p_s^i (v_s^i - \bar{v}_s)^2 + \sum_{i=k}^m A_i.$$

Let $0 < \epsilon < |v_s' - \bar{v}_s|/3$. By assumption, $\|v^k - v'\| < \epsilon$ for all sufficiently large $k \in \mathcal{K}$. Let m be such that $\|v^i - v'\| < \epsilon$ for all $k \leq i \leq m$. Then $|v_s^i - v_s'| \leq \|v^i - v'\| \leq 2\epsilon$ for these i , so $|v_s^i - \bar{v}_s| \geq \epsilon$. The last displayed inequality yields

$$\|v^{m+1} - \bar{v}\|^2 \leq \|v^k - \bar{v}\|^2 - 2\epsilon^2 \sum_{i=k}^m \alpha_i p_s^i + \sum_{i=k}^m A_i. \quad (28)$$

If $l(k, \epsilon)$ does not exist for large $k \in \mathcal{K}$, we can let $k \rightarrow \infty$, $k \in \mathcal{K}$, $m = m(k) \rightarrow \infty$ in such a way that

$$\sum_{i=k}^{m(k)} \alpha_i \rightarrow \infty. \quad (29)$$

Let us split the set $\{k, k+1, \dots, m(k)\}$ into subsets $\{i_1, \dots, i_2 - 1\}$, $\{i_2, \dots, i_3 - 1\}, \dots$, $\{i_{M(k)-1}, \dots, i_{M(k)} - 1\}$, where $i_1 = k$, $i_{M(k)} - 1 = m(k)$ and

$$\sum_{i=i_j}^{i_{j+1}-1} \alpha_i \geq \epsilon.$$

Since $\alpha_i \rightarrow 0$ by (14), the relation (29) implies that $M(k) \rightarrow \infty$ as $k \rightarrow \infty$, $k \in \mathcal{K}$. Thus, by virtue of assumption (27), $\sum_{i=k}^{m(k)} \alpha_i p_s^i \rightarrow \infty$. Note that $\sum_{i=k}^{m(k)} A_i \rightarrow 0$ if $k, m(k) \rightarrow \infty$, except for a certain null set, and we obtain a contradiction in (28). Therefore, for all sufficiently small $\epsilon > 0$ and all sufficiently large $k \in \mathcal{K}$, the index $l(k, \epsilon)$ is finite and

$$\|v^{l(k, \epsilon)} - \bar{v}\|^2 \leq \|v^k - \bar{v}\|^2 - 2\epsilon^2 \sum_{i=k}^{l(k, \epsilon)-1} \alpha_i p_s^i + \sum_{i=k}^{l(k, \epsilon)-1} A_i. \quad (30)$$

Stage 2: We shall prove that the sum of stepsizes between $k \in \mathcal{K}$ and $l(k, \epsilon)$ is comparable with ϵ , if k is large enough. By the definition of $l(k, \epsilon)$ we have $\|v^{l(k, \epsilon)} - v'\| > \epsilon$. Since $v^k \rightarrow v'$, $k \in \mathcal{K}$, we also have $\|v^{l(k, \epsilon)} - v^k\| > \epsilon/2$ for all sufficiently large $k \in \mathcal{K}$. Thus

$$\sum_{i=k}^{l(k, \epsilon)-1} \alpha_i \|\xi^i\| > \epsilon/2.$$

Let us observe that conditions (12) and (14) imply that the random series

$$\sum_{k=1}^{\infty} \alpha_k (\|\xi^k\| - \mathbb{E}\{\|\xi^k\| \mid \mathcal{F}_k\})$$

is a convergent martingale. Therefore, unless ω is in a certain null set,

$$\sum_{i=k}^{l(k, \epsilon)-1} \alpha_i \|\xi^i\| = \sum_{i=k}^{l(k, \epsilon)-1} \alpha_i \kappa_i + \sigma_k,$$

where $\kappa_i = \mathbb{E}\{\|\xi^i\| \mid \mathcal{F}_i\}$ and $\sigma_k = \sum_{i=k}^{l(k, \epsilon)-1} \alpha_i (\|\xi^i\| - \kappa_i) \rightarrow 0$, as $k \rightarrow \infty$, $k \in \mathcal{K}$. Thus, for all sufficiently large $k \in \mathcal{K}$

$$\sum_{i=k}^{l(k, \epsilon)-1} \alpha_i \kappa_i \geq \epsilon/3.$$

By (12) there exists a constant C such that $\kappa_i \leq C$ for all i . This combined with the last inequality yields

$$\sum_{i=k}^{l(k, \epsilon)-1} \alpha_i \geq \frac{\epsilon}{3C}, \quad (31)$$

for all sufficiently large $k \in \mathcal{K}$.

Stage 3: Combining inequalities (30) and (31) we get

$$\|v^{l(k, \epsilon)} - \bar{v}\|^2 \leq \|v^k - \bar{v}\|^2 - \frac{2\epsilon^3}{3C} + \sum_{i=k}^{l(k, \epsilon)-1} A_i.$$

Since the entire sequence $\{\|v^i - \bar{v}\|^2\}$ is convergent, passing to the limit in the last inequality with $k \rightarrow \infty$, $k \in \mathcal{K}$, yields a contradiction. Consequently, no accumulation point v' may have $v'_s \neq \bar{v}_s$. \square

Clearly, if condition (15) holds, then we have probing for all $s \in \{1, \dots, M\}$ and the above result implies the convergence of v^k to \bar{v} a.s. But the probing condition of Definition 4.1 is weaker than (15) even if we impose it on all coordinates.

A similar result holds for SPAR-Obj.

Theorem 4.3 *Assume (8), (12)–(14) and (22), (24). Then SPAR-Obj generates a sequence $\{v^k\}$ such that $v_s^k \rightarrow \bar{v}_s$ a.s. for every $s \in \{1, \dots, M\}$ satisfying the probing condition (27).*

The proof is basically the same. We first establish (17), proceeding exactly as in the proof of Theorem 3.1. After this we may apply the proof of Theorem 4.2.

5 Learning While Optimizing

Let us now return to problem (1) and let us assume that at each step we generate the next observation point by solving the approximate problem (4), that is

$$s_i^k = x_i^k, \quad i = 1, \dots, n, \quad k = 1, 2, \dots$$

If the solution x^k is not unique, we choose it at random from the set of optimal solutions of (4). For each coordinate function f_i we observe two random variables: η_i^k satisfying (8), and η_{i+}^k satisfying (18). Then we update the left and right slopes for each function f_i according to (20), and the iteration continues. In this way we define the sequence $\{v_i^k\}$, $k = 1, 2, \dots$ of estimates of the slopes of f_i , $i = 1, \dots, n$, and a sequence x^k of the solutions of approximate models (4).

This algorithm is well-defined if the approximate problems (4) have integer solutions for all concave piecewise linear functions f_i having integer break points. This is true, for example, for models having alternative network representations, as those discussed in Powell & Topaloglu (to appear).

It turns out that even without random learning such a method has a good chance of converging to an optimal solution.

Let us note that for a concave, piecewise linear and separable function

$$F(x) = \sum_{i=1}^n f_i(x_i), \quad (32)$$

where each f_i is defined as

$$f_i(x_i) = \sum_{s=1}^l v_s + v_{l+1}(x - l) \quad (33)$$

with an integer l such that $l \leq x < l + 1$, the subdifferential of F at an *integer* point x is given by

$$\partial F(x) = [v_{1,x_1+1}, v_{1,x_1}] \times [v_{2,x_2+1}, v_{2,x_2}] \times \cdots \times [v_{n,x_n+1}, v_{n,x_n}].$$

The necessary and sufficient condition of optimality for problem

$$\max_{x \in X} F(x),$$

where X is a convex closed set, has the form

$$0 \in \partial F(x) - N(x), \quad (34)$$

with $N(x)$ being the normal cone to X at x . A point \hat{x} is called *stable* if it satisfies

$$0 \in \text{int}[\partial F(\hat{x}) - N(\hat{x})]. \quad (35)$$

It can be seen directly from conditions (34) and (35) that a stable point \hat{x} is also a solution to a perturbed problem

$$\max_{x \in X} \tilde{F}(x),$$

provided that $\text{dist}(\partial F(\hat{x}), \partial \tilde{F}(\hat{x})) < \epsilon$ and ϵ is a sufficiently small positive number.

Clearly, the solutions x^k of our approximate problems (4) satisfy condition (34) for the approximate functions \tilde{f}^k constructed by the method. Then, by passing to the limit, we can conclude that each accumulation point (x^*, v^*) of the sequence $\{(x^k, v^k)\}$ satisfies the condition

$$0 \in \partial F^*(x^*) - N(x^*),$$

with F^* constructed from v^* as in (32)–(33). We shall show that if such an accumulation point satisfies the slightly stronger condition of stability, it is optimal for the original problem.

Theorem 5.1 *Assume that for each $i = 1, \dots, n$ the conditions (8), (12)–(14) and (18)–(19) are satisfied. If an accumulation point (x^*, v^*) of the sequence $\{(x^k, v^k)\}$ generated by the algorithm, satisfies the stability condition:*

$$0 \in \text{int} [\partial F^*(x^*) - N(x^*)], \quad (36)$$

then x^* is an optimal solution of (1).

Proof. Let us observe that relation (21) holds for each coordinate i . Therefore inequality (17) is true for each coordinate i :

$$\|v_i^{k+1} - \bar{v}_i\|^2 \leq \|v_i^k - \bar{v}_i\|^2 - 2\alpha_k \langle v_i^k - \bar{v}_i, P_i^k(v_i^k - \bar{v}_i) \rangle + A_{ki}. \quad (37)$$

The matrix P_i^k , which is an \mathcal{F}_k -measurable random variable, is a nonnegative diagonal matrix with positive entries corresponding to the i -th coordinates of possible solutions to (4). The series $\sum_{k=1}^{\infty} A_{ki}$ is convergent a.s. Therefore, the sequence $\{\|v_i^k - \bar{v}_i\|\}$ is convergent for every $i = 1, \dots, n$.

Let us consider a convergent subsequence $\{(v^k, x^k)\}$, $k \in \mathcal{K}$, and let us denote by (v^*, x^*) its limit. If the stability condition holds, then there exists $\epsilon > 0$ such that for all iterations k for which $|v_{i,x_i^*}^k - v_{i,x_i^*}^*| < \epsilon$, $i = 1, \dots, n$, the solution of the approximate problem (4) is x^* . Then the coefficients $p_{i,s}^k$ are 1 for $s = x_i^k$ and $s = x_i^k + 1$ and zero otherwise. Let us focus our attention on the coordinates $s = x_i^k$. Inequality (37) takes on the form

$$\|v_i^{k+1} - \bar{v}_i\|^2 \leq \|v_i^k - \bar{v}_i\|^2 - 2\alpha_k (v_{i,x_i^*}^k - \bar{v}_{i,x_i^*})^2 + A_{ki}. \quad (38)$$

The series $\sum_k A_{ki}$ is convergent a.s. Let $k \in \mathcal{K}$ be large enough so that $|v_{i,x_i^*}^k - v_{i,x_i^*}^*| < \epsilon/2$. Consider $j \geq k$ such that

$$|v_{i,x_i^*}^j - v_{i,x_i^*}^*| < \epsilon \quad \text{for all } i = 1, \dots, n. \quad (39)$$

Let us suppose that

$$v_{i,x_i^*}^* \neq \bar{v}_{i,x_i^*} \quad \text{for at least one } i. \quad (40)$$

We shall prove that it leads to a contradiction.

Clearly, we can always choose a sufficiently small $\epsilon > 0$ such that $|v_{i,x_i^*}^* - \bar{v}_{i,x_i^*}| > 2\epsilon$. Then for the iterations j satisfying (39) we have $|v_{i,x_i^*}^j - \bar{v}_{i,x_i^*}| > \epsilon$, and inequality (38) implies:

$$\|v_i^{j+1} - \bar{v}_i\|^2 \leq \|v_i^j - \bar{v}_i\|^2 - 2\alpha_j \epsilon^2 + A_{ki}. \quad (41)$$

Since the series $\sum \alpha_j$ is divergent, the set of consecutive $j \geq k$ for which condition (39) holds must be finite. Thus, similarly to Stage 1 of the proof of Theorem 4.2, the index

$$l(k, \epsilon) = \inf\{j \geq k : \max_{1 \leq i \leq n} |v_{i,x_i^*}^j - v_{i,x_i^*}^*| > \epsilon\}$$

must be finite, for all sufficiently small $\epsilon > 0$. Summing up, for some coordinate i we have

$$\begin{aligned} |v_{i,x_i^*}^k - v_{i,x_i^*}^*| &\leq \epsilon/2, \\ |v_{i,x_i^*}^{l(k,\epsilon)} - v_{i,x_i^*}^*| &> \epsilon. \end{aligned}$$

Exactly in the same way as in Stage 2 of the proof of Theorem 4.2 we can now prove that there exists $\delta > 0$ such that

$$\sum_{j=k}^{l(k,\epsilon)-1} \alpha_j \geq \delta\epsilon.$$

Combining this inequality with (41) we conclude that for some coordinate i

$$\|v_i^{l(k,\epsilon)} - \bar{v}_i\|^2 \leq \|v_i^k - \bar{v}_i\|^2 - 2\delta\epsilon^3 + \sum_{j=k}^{l(k,\epsilon)-1} A_{kj}.$$

Passing to the limit with $k \rightarrow \infty$, $k \in \mathcal{K}$, we obtain a contradiction with the convergence of the entire sequence $\{\|v_i^j - \bar{v}_i\|^2\}$. Therefore our assumption (40) must be wrong, and we have

$$v_{i,x_i^*}^* = \bar{v}_{i,x_i^*} \quad \text{for all } i = 1, \dots, n. \quad (42)$$

Using the fact that inequality (38) is true also with x_i^k replaced by $x_i^k + 1$ (if $x_i^k < M_i$), we can apply the same argument to prove

$$v_{i,x_i^*+1}^* = \bar{v}_{i,x_i^*+1} \quad \text{for all } i = 1, \dots, n. \quad (43)$$

For $x_i^* = M_i$ we take the convention that $v_{i,x_i^*+1}^* = \bar{v}_{i,x_i^*+1} = -\infty$. Consequently,

$$\partial F(x^*) = \partial F^*(x^*)$$

and the point x^* is optimal for (1). \square

In a similar way (and under identical assumptions) we can prove the convergence of the version that uses function value estimates.

Theorem 5.2 *Assume (8), (12)–(14), (18)–(19), and (22), (24). If an accumulation point (x^*, v^*) of the sequence $\{(x^k, v^k)\}$ generated by SPAR-Obj, satisfies the stability condition (36) then x^* is an optimal solution of (1).*

The proof is almost a verbatim copy of the proof of Theorem 5.1, with the modifications as in Theorem 4.3.

6 Projection

Let us now describe the way the projection $v = \Pi_V(z)$ can be calculated. Clearly, v is the solution to the quadratic programming problem

$$\min \frac{1}{2} \|v - z\|^2 \quad (44)$$

$$\text{subject to: } v_{s+1} - v_s \leq 0, \quad s = 0, \dots, M, \quad (45)$$

where for uniformity we denote $v_0 = B$, $v_{M+1} = -B$. Associating with (45) Lagrange multipliers $\lambda_s \geq 0$, $s = 0, \dots, M$, we obtain the necessary and sufficient optimality conditions:

$$v_s = z_s + \lambda_s - \lambda_{s-1}, \quad s = 1, \dots, M, \quad (46)$$

$$\lambda_s(v_{s+1} - v_s) = 0, \quad s = 0, \dots, M. \quad (47)$$

If i_1, \dots, i_2 is a sequence of coordinates such that

$$v_{i_1-1} > v_{i_1} = v_{i_1+1} = \dots = c = \dots = v_{i_2-1} = v_{i_2} > v_{i_2+1},$$

then adding the equations (46) from i_1 to i_2 yields

$$c = \frac{1}{i_2 - i_1 + 1} \sum_{s=i_1}^{i_2} z_s.$$

If $i_1 = 1$, then c is the minimum of the above average and B , and for $i_2 = M$ the maximum of $-B$ and this average has to be taken.

The second useful observation is that $v^k \in V$ and z^k computed by (9) differs from v^k in just one coordinate. If $z^k \notin V$, one of two cases must occur: either $z_{s^k-1}^k < z_{s^k}^k$, or $z_{s^k+1}^k > z_{s^k}^k$.

If $z_{s^k-1}^k < z_{s^k}^k$, we search for the largest $1 < i \leq s^k$ for which

$$z_{i-1}^k \geq \frac{1}{s^k - i + 1} \sum_{s=i}^{s^k} z_s^k. \quad (48)$$

If such i cannot be found we set $i = 1$. Then we calculate

$$c = \frac{1}{s^k - i + 1} \sum_{s=i}^{s^k} z_s^k \quad (49)$$

and set

$$v_j^{k+1} = \min(B, c), \quad j = i, \dots, s^k. \quad (50)$$

We have $\lambda_0 = \max(0, c - B)$, and

$$\lambda_s = \begin{cases} 0 & s = 1, \dots, i-1, \\ \lambda_{s-1} + z_s - v_s & s = i, \dots, s^k-1, \\ 0 & s = s^k, \dots, M. \end{cases}$$

It is straightforward to verify that the solution found and the above Lagrange multipliers satisfy conditions (46)–(47).

The procedure in the case when $z_{s^k}^k < z_{s^{k+1}}^k$ is symmetrical: it is the same procedure applied to the graph of z rotated by π .

Let us now consider the method which employs *two* random variables at each iteration, with Step 3 as in (20). Then both z_{s^k} and $z_{s^{k+1}}$ may differ from v_{s^k} and $v_{s^{k+1}}$ (although we still have $z_{s^k} > z_{s^{k+1}}$). We shall show that algorithm (48)–(50) can easily be adapted to this case.

Suppose that $z_{s^k} > z_{s^{k-1}}$. We apply (48)–(49) to compute the candidate value for c . Now two cases may occur.

Case 1. If $c \geq z_{s^{k+1}}$, we may apply (50). We can now focus on the points to the right: $z_{s^{k+1}}, z_{s^{k+2}}, \dots$. We apply the symmetric analogue of (48)–(50) to these values, and the projection is accomplished.

Case 2. If $c < z_{s^{k+1}}$, the value of c is not correct. We need to include $z_{s^{k+1}}$ into the averaging procedure. Thus we repeat (48)–(50) but with s^k replaced by $s^k + 1$ in (48), although we still search for $1 < i \leq s^k$. After this we apply (50) for $j = i, \dots, s^k + 1$.

If $z_{s^{k+1}} < z_{s^{k+2}}$ the situation is symmetrical to the one discussed above (after rotating the graph of z by π), and an analogous procedure can be applied.

7 Experiments for separable problems

To illustrate the behavior of the methods discussed in this paper we have at first considered the problem of maximizing a concave function of one variable, exactly as discussed in Sections 2–5. The particular examples that we considered have the following form, where $f(\cdot, \omega)$ is as defined in Example 1.1.

$$\begin{aligned} & \max \mathbb{E}f(x, \omega) \\ & \text{subject to: } 0 \leq x \leq M. \end{aligned}$$

We have compared several methods

- SPAR - This is the basic learning algorithm with projection.
- SPAR-Obj(a) with $\rho_k = s^k$ - This uses objective function estimates to help with the learning, using weights of $\frac{1}{s^k}$ for the objective function estimates.
- SPAR-Obj(b) with $\rho_k = Ms^k$ - Same as above, but with a much smaller weight on the objective function.
- The Leveling Method of Topaloglu & Powell (2001) - This algorithm maintains concavity by forcing slopes that violate an updated estimate to be no larger (to the left) or no smaller (to the right) than the most recently updated cell. This algorithm has been shown to be provably convergent.

- The CAVE Algorithm of Godfrey & Powell (2001) - This was the first algorithm suggested for adaptively estimating piecewise linear functions while maintaining concavity.
- Nonconvex - SPAR without the projection operation.

In the first series of experiments, the random variable s^k is generated from the uniform distribution over $[1, M]$ and the demand random variable D^k is generated from the Poisson distribution for every iteration k . Having sampled s^k and D^k , we compute the left-hand slope of $f(\cdot, D^k)$ at s^k as in Example 1.2. Using the slope information for iteration k and the value of s^k , we can obtain the approximation $\bar{f}^{k+1}(\cdot)$ (we call this a *learning step*).

In order to have an idea about the “quality” of the approximation at iteration k , we find the maximizer of $\bar{f}^k(\cdot)$ (let $\bar{x}^k = \arg \max \bar{f}^k(\cdot)$) and compute $\mathbb{E}f(\bar{x}^k, \omega)$. We note that for a complex problem this evaluation step is hard to carry out since we assume that we normally cannot take the expectation $\mathbb{E}f(\bar{x}^k, \omega)$. $\mathbb{E}f(\bar{x}^k, \omega)$ gives us an idea about the average “actual” performance of the solution given by the approximation $\bar{f}^k(\cdot)$ (we call this an *optimizing step*). Since the problem involved is one dimensional, \bar{x}^k can be found by simple enumeration, allowing us to include the method using nonconvex models (SPAR without the projection operator) into the comparison. This will not be possible in the experiments with multidimensional models that we report later.

We have run each method using 100 demand realizations and each method has been run 5000 times to remove the effect of the particular sample path $D^1, \dots, D^k, \dots, D^{100}$ on the results. Figure 1 presents the averages of $\mathbb{E}f(\bar{x}^k, \omega)$ over 5000 runs.

For each method the sequence of stepsizes was defined as $\alpha_k = 20/(40 + k)$. We take $M = 30$ and D^k as truncated-Poisson distributed with mean 16. We used a truncated form of Poisson distribution to assure that the demand random variables are realized over the relevant region. Finally, we take $q = 2$, $c = 1$.

We see that our basic learning method, SPAR, performs very well. Its quality can be slightly improved by using objective function estimates as in SPAR-Obj, but the weight associated to them must be significantly smaller than the weight associated to the slope observations, as the comparison of versions 2(a) and 2(b) shows. The method without the projection operation (called ‘Nonconvex’ here) performs much worse. The projection transforms the results of an experiment at one value of the argument into adjustments at multiple values, thus facilitating the convergence.

In the second group of experiments the same methods were run with optimizing steps only: that is, s^k is chosen to be the maximizer of $\bar{f}^k(\cdot)$ at every iteration k , as discussed in Section 5. Then $\mathbb{E}f(s^k, \omega)$ gives the average “actual” performance of the solution given by the approximation $\bar{f}^k(\cdot)$. This version concentrates its efforts in the areas of interest, and one might expect that it has a potential of being more efficient. In order to give the method a chance to learn that the current point is not optimal, we have to use the version with the observation of both left and right derivatives at each point, as in (20). This should additionally enhance the convergence. The results are collected in Figure 2.

The obvious conclusion that we can draw from Figures 1 and 2 is that using optimizing steps is counterproductive at earlier iterations, and the ‘optimizing’ methods are slower at the beginning than the methods with learning. Also, when two-sided derivatives are

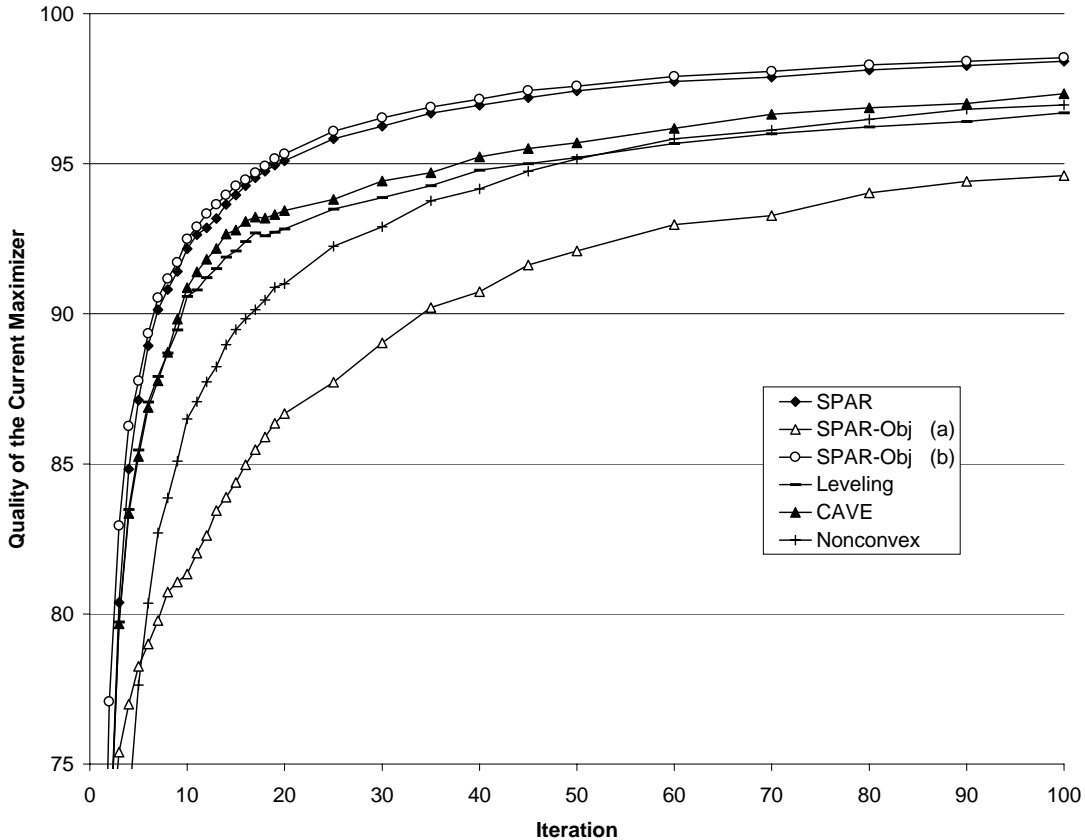


Figure 1: Comparison of methods with alternating learning and optimizing steps in the one-dimensional case. “Quality of the current maximizer” is the average true objective value at the maximizer of the model constructed at the current iteration relative to the maximum value (in %).

observed and only optimizing steps are made, there is virtually no benefit from using objective values.

Our next series of experiments have been done on multi-dimensional problems, as in Example 1.1:

$$\begin{aligned}
 & \max \mathbb{E} \sum_{i=1}^n f_i(x_i, \omega) \\
 & \text{subject to: } \sum_{i=1}^n x_i \leq b, \\
 & \quad 0 \leq x_i \leq M_i, \quad i = 1, \dots, n.
 \end{aligned}$$

Clearly, both SPAR and SPAR-Obj can be applied to these problems componentwise: for a given resource allocation $s^k = (s_1^k, \dots, s_n^k)$ and a sampled demand realization $D^k = (D_1^k, \dots, D_n^k)$, we can apply the updates of SPAR and SPAR-Obj for each component

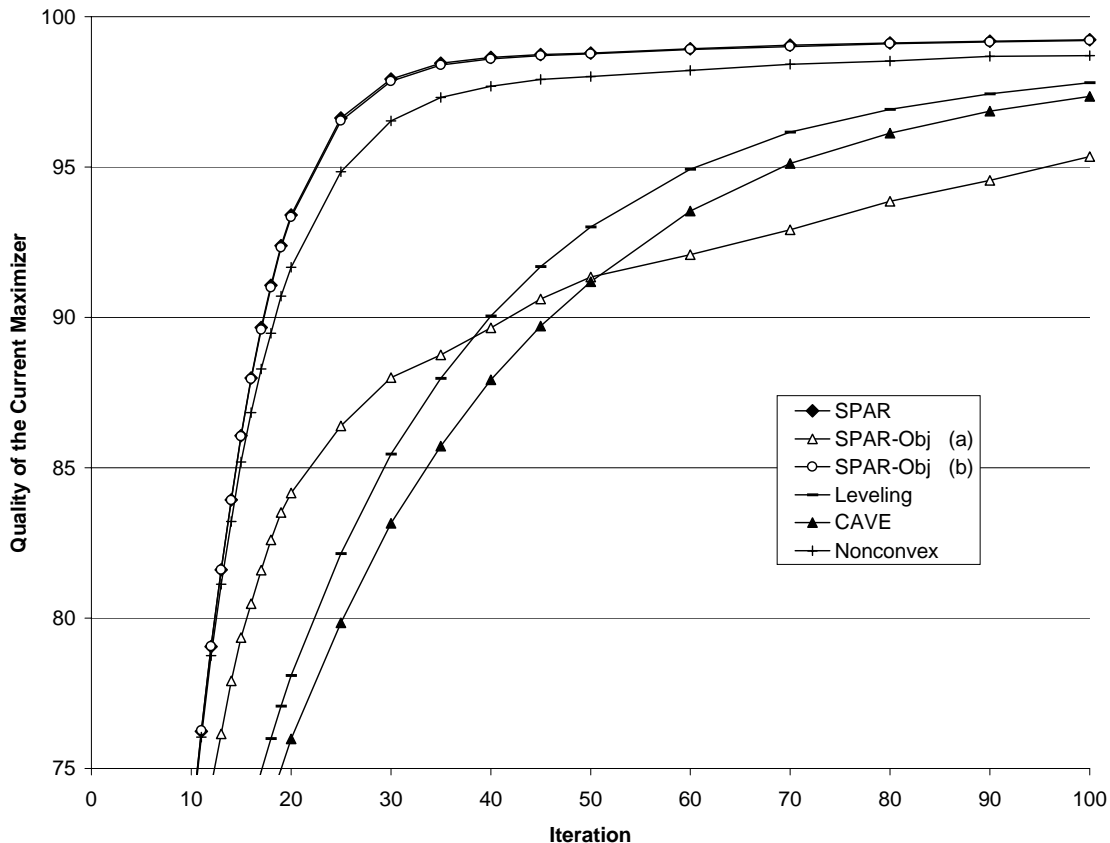


Figure 2: Comparison of methods with only optimizing steps in the one-dimensional case. “Quality of the current maximizer” is the average true objective value at the maximizer of the model constructed at the current iteration relative to the maximum value (in %). SPAR and SPAR-Obj(b) generate virtually identical results, and cannot be distinguished.

$i = 1, \dots, n$. In the description below s_i^k plays the role of s^k for the i -th component, ρ_{ki} the role of ρ_k , etc. We have compared the following methods

- SPAR
- SPAR-Obj(a) with $\rho_{ki} = s_i^k$
- SPAR-Obj(b) with $\rho_{ki} = M_i s_i^k$
- The Leveling Method of Topaloglu & Powell (2001)
- The CAVE Algorithm of Godfrey & Powell (2001)

We do not include the nonconvex case here, because it requires global optimization techniques. In the other methods, we have employed linear programming to maximize $\sum_{i=1}^n f_i^k(\cdot)$.

In the first series of experiments, the random variable s^k is generated from the uniform distribution on the rectangle $[1, M_1] \times \dots \times [1, M_n]$ and each component of the demand

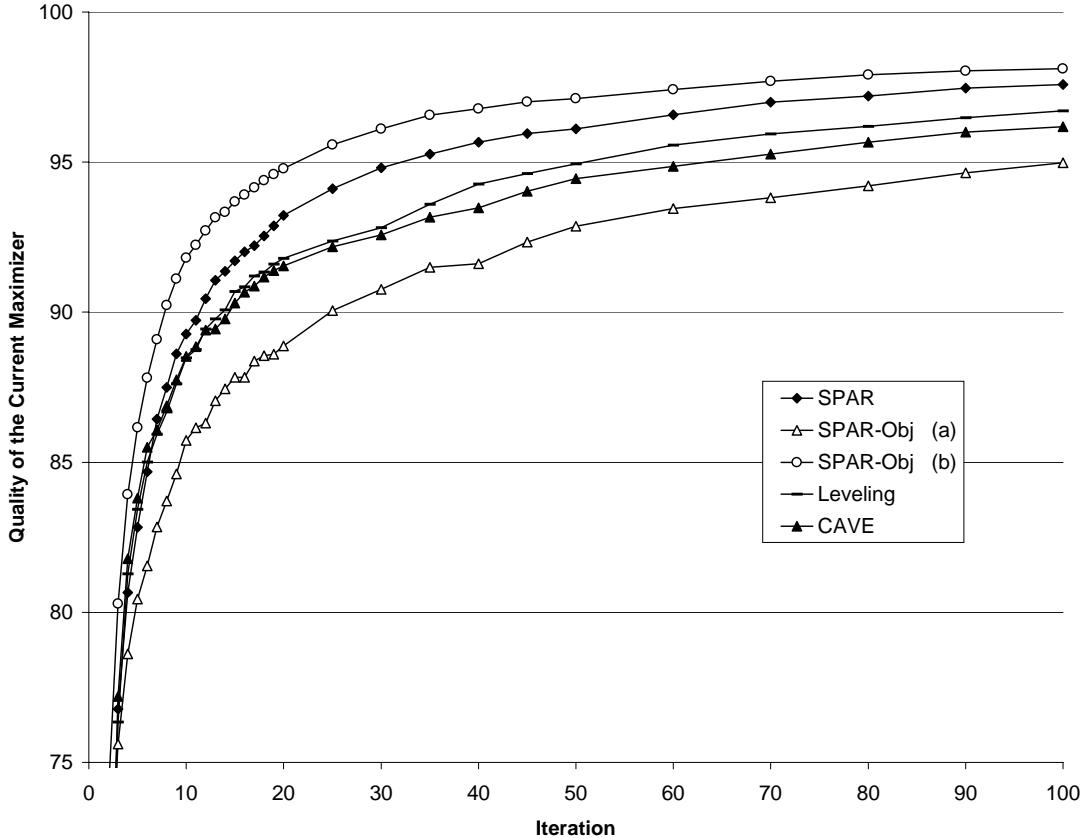


Figure 3: Comparison of methods with alternating learning and optimizing steps in the multi-dimensional case. “Quality of the current maximizer” is the average true objective value at the maximizer of the model constructed at the current iteration relative to the maximum value (in %).

variable D^k is generated independently from the Poisson distribution for every iteration k . Having sampled s_i^k and D_i^k , we compute the left-hand slope of $f_i(\cdot, D_i^k)$ at s_i^k and we obtain the approximation $\bar{f}_i^{k+1}(\cdot)$ (*learning step*). Then we find the maximizer of $\sum_{i=1}^n \bar{f}_i^k(\cdot)$ (let $\bar{x}^k = \arg \max \sum_{i=1}^n \bar{f}_i^k(\cdot)$), and compute $\mathbb{E} \sum_{i=1}^n f_i(\bar{x}_i^k, \omega)$ (*optimizing step*). Each method has been run 50 times using 100 demand realizations at each run. Figure 3 presents the averages of these 50 runs.

For each method and coordinate (activity) the sequence of stepsizes was defined as $\alpha_k = 20/(40+k)$. We take the number of coordinates to be $n = 90$. M_i ranges between 20 and 40, c_i ranges between 0.6 and 1.4 for different 90 activities. D_i^k is truncated-Poisson distributed with mean ranging between 9 and 21 for $i = 1, \dots, 90$. Finally $q_i = 2$ and $b = 950$.

Again, our basic learning method, SPAR, performs very well. Using objective function estimates (SPAR-Obj(b)) leads to significant improvements now, provided that the weight assigned to them is small. Apparently, in a large dimensional problem it is likely that for at least one of the functions f_i the early estimates are bad. This has a dramatic effect on

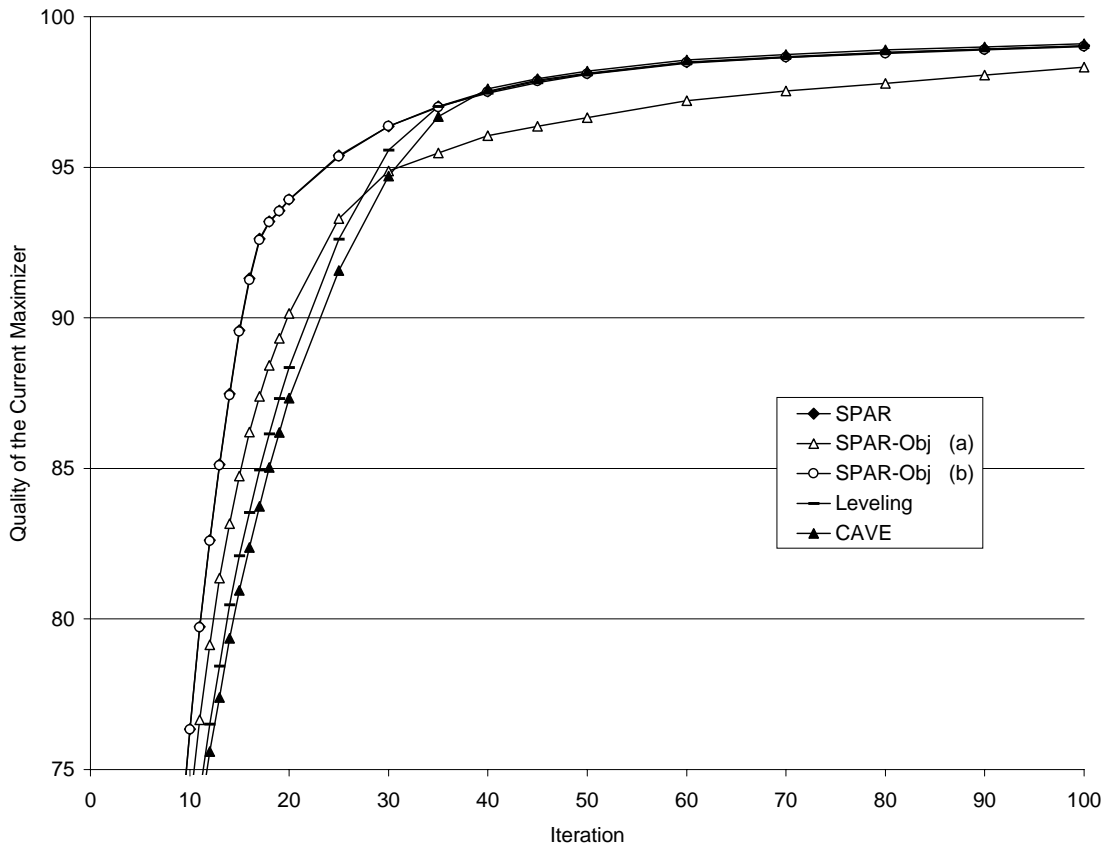


Figure 4: Comparison of methods with only optimizing steps in the multi-dimensional case. “Quality of the current maximizer” is the average true objective value at the maximizer of the model constructed at the current iteration relative to the maximum value (in %). SPAR and SPAR-Obj(b) generate virtually identical results, and cannot be distinguished.

the solution of the entire problem, and even small improvements in the learning speed for this function pay much in the overall optimization.

In the next groups of experiments the same methods were run with optimizing steps only: that is, at every iteration the maximizer of the current approximate problem (4) is chosen as the next point s^k at which the observations are made (see Section 5). We used again two-sided slope observations. The results are collected in Figure 4.

All optimizing methods perform worse than learning methods at early iterations, which are of most interest for us. Only after 40 iterations did the versions with optimizing steps take the lead over their learning counterparts. The disadvantage of the pure optimizing method is larger than in the one dimensional case, because it is sufficient that the learning lags for just one coordinate to significantly distort the optimal solution of (4).

The comparison of methods with uniform learning and with only optimizing steps suggests a strategy of nonuniform learning patterns: many learning steps at the beginning and less later. We have designed such a nonuniform pattern: the number of optimizing

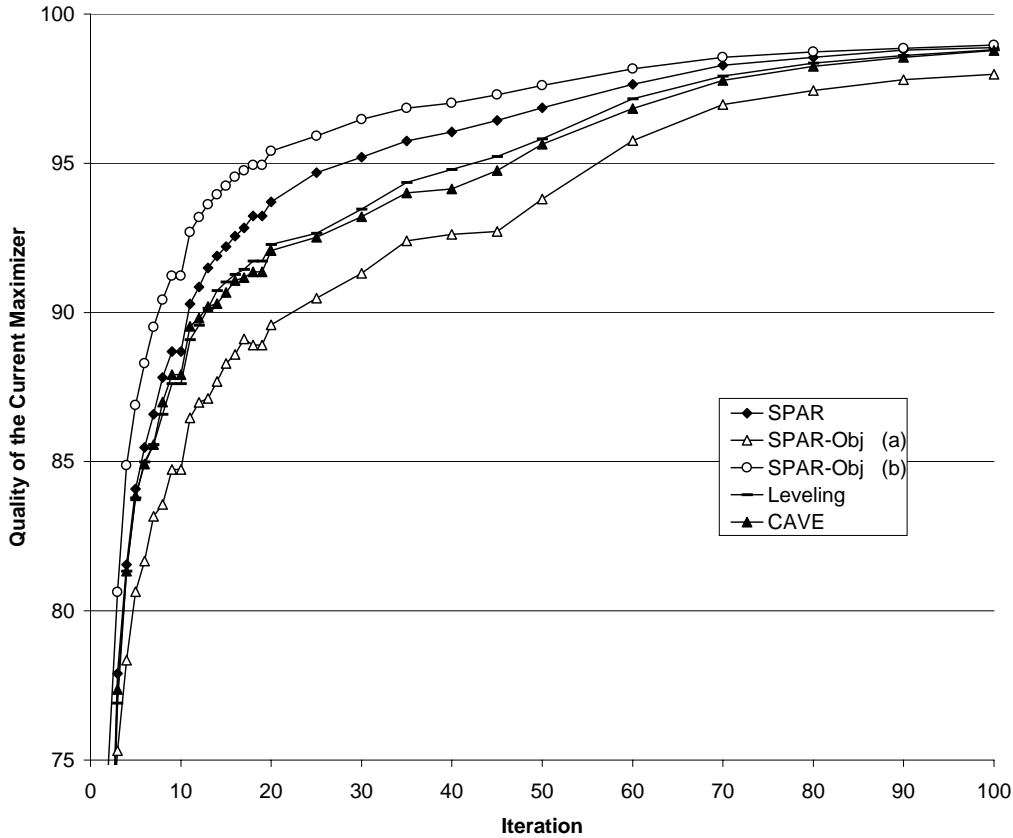


Figure 5: Comparison of methods with nonuniform learning patterns in the multi-dimensional case. “Quality of the current maximizer” is the average true objective value at the maximizer of the model constructed at the current iteration relative to the maximum value (in %).

steps between two learning steps is equal to the number of learning steps made so far. Thus, the frequency of learning fades out slowly. The results are collected in Figure 5.

The relative performance of all methods is the same as before, but they further improved over both “pure” strategies. This can be seen best after superimposing the graphs for the best method, SPAR-Obj(b), in the three versions, as shown in Figure 6.

We see that nonuniform learning has all the advantages of the uniform learning, and additionally it has a better tail performance.

Several conclusions can be drawn from our experiments.

First, the application of the projection operator facilitates the convergence. It provides an update to a range of values on the basis of the observation obtained at one value. Secondly, learning is useful, especially at earlier iterations. Instead of trying to shoot at the optimal point for the current model, it is better to collect information at randomly selected points from time to time. Nonuniform learning patterns, with the frequency of learning steps decreasing as the iterations proceed, appear to be best here. Finally, the indirect use of noisy objective values to correct the subgradients has a positive effect on

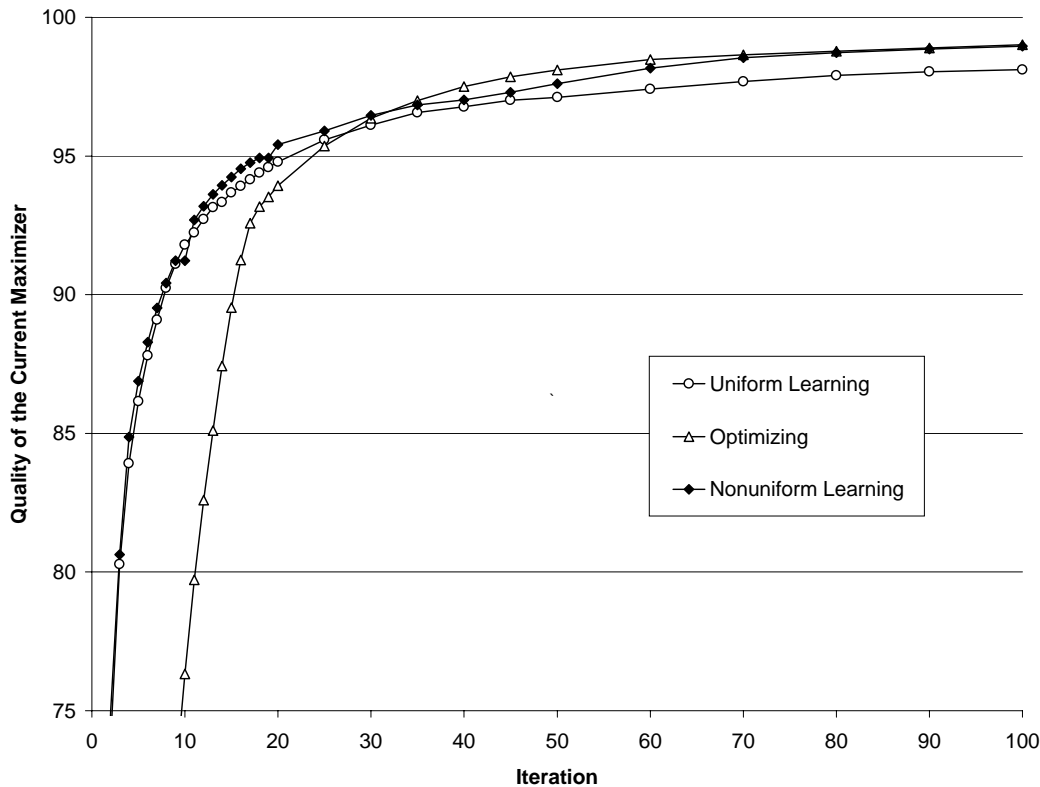


Figure 6: Comparison of three versions of the method with slope and objective observations: uniform learning, pure optimization, and non-uniform learning, in the multi-dimensional case. “Quality of the current maximizer” is the average true objective value at the maximizer of the model constructed at the current iteration relative to the maximum value (in %).

the convergence, provided that the weight of the additional modification is small.

The learning-based methods provide good approximations to the solutions at early iterations, which makes them attractive candidates for problems where the cost of one experiment is high.

8 Application to Non-Separable Resource Allocation Problems

We now turn our attention to nonseparable problems that arise in the context of two-stage stochastic resource allocation problems. Section 8.1 shows that there exists a separable approximation that will produce an optimal first-stage solution, although there is no guarantee that our algorithm will find this approximation. We also provide a bound on the solution provided by our algorithm. Then, section 8.2 compares our algorithm, which

produces integer solutions, to the solutions produced by several variations of Benders decomposition. We consider only the optimizing variation of the algorithm presented in section 5

8.1 Outline of the method

Let us start from the following observation. Consider the problem

$$\max_{x \in X} F(x), \quad (51)$$

where $X \subset \mathbb{R}^n$ is a closed convex set, and $F : \mathbb{R}^n \rightarrow \mathbb{R}$ is a concave function. Suppose that \hat{x} is the optimal solution of the problem, and that F is subdifferentiable at \hat{x} . Let us construct a concave separable approximation of F at \hat{x} in the form

$$\bar{F}(x) = F(\hat{x}) + \sum_{i=1}^n \bar{f}_i(x_i),$$

where

$$\bar{f}_i(x_i) = \begin{cases} F'(\hat{x}, e_i)(x_i - \hat{x}_i) & \text{if } x_i \geq \hat{x}_i, \\ F'(\hat{x}, -e_i)(\hat{x}_i - x_i) & \text{if } x_i \leq \hat{x}_i. \end{cases}$$

In the formula above $F'(\hat{x}_i, d)$ denotes the directional derivative of F at \hat{x} in direction d , and e_i is the i -th unit vector in \mathbb{R}^n . Then \hat{x} is also the solution of the deterministic approximate problem

$$\max_{x \in X} \bar{F}(x).$$

Indeed, by the optimality of \hat{x} , the directional derivative $F'(\hat{x}, d)$ of F at \hat{x} in any direction

$$d \in \text{cone}(X - \hat{x}),$$

is non-positive. Since the directional derivative is a concave positively homogeneous function, and

$$d = \sum_{i=1}^n d_i e_i = \sum_{i=1}^n (d_i)_+ e_i + \sum_{i=1}^n (d_i)_- (-e_i),$$

we have

$$F'(\hat{x}, d) \geq \sum_{i=1}^n (d_i)_+ F'(\hat{x}, e_i) + \sum_{i=1}^n (d_i)_- F'(\hat{x}, -e_i) = \bar{F}'(\hat{x}, d). \quad (52)$$

Therefore $\bar{F}'(\hat{x}, d) \leq 0$ for every feasible direction d , as required.

The converse statement is not true, in general, and one can easily construct counterexamples, in which the separable approximation $\bar{F}(\cdot)$ constructed at some point y achieves its maximum at y , but $F(\cdot)$ does not.

Then, however, we can derive an upper bound on the optimal value of $F(\cdot)$ in X as follows. If \hat{x} is an optimal solution of problem (51), then

$$F(\hat{x}) - F(y) \leq F'(y, \hat{x} - y) \leq -F'(y, y - \hat{x}) \leq -\bar{F}'(y, y - \hat{x}).$$

In the second inequality above we have used the concavity of $F'(y, \cdot)$, and in the last (52). Therefore

$$\max_{x \in X} F(x) - F(y) \leq -\min_{x \in X} \bar{F}'(y, y - x),$$

and the quantity at the right hand side can be easily calculated or estimated, given the current piecewise linear separable approximation \bar{F} .

In a stochastic situation, of course, our methods use *estimates* of the directional derivatives in the directions e_i , rather than exact values, so the error bound will be an estimate, as well.

Summing up, the method proceeds as follows. At iteration k it has n separable approximations $\bar{f}_i^k(\cdot)$, defined by slope vectors $v_i^k = (v_{i1}^k, \dots, v_{iM_i}^k)$, where M_i is an upper bound on the variable x_i . The current point x^k is defined as the solution of the approximate problem (4). Then we observe a random vector η^k such that

$$\mathbb{E}\{\eta^k \mid v^1, \dots, v^k, x^k\} \in \partial F(x^k), \text{ a.s. .}$$

It is used to update the n slope vectors v_i^k according to SPAR, with η_i^k playing the role of the slope observation for \bar{f}_i at x_i^k . The concavity of the updated separable approximation is preserved owing to the projection, and the next point x^{k+1} can be calculated from (4).

There are two sources of errors in this approximate procedure. The main one is the use of separable approximations, as discussed above. The second one is the use of an arbitrary stochastic subgradient η^k rather than estimates of the forward and backward directional derivatives, as required in Section 5. Nevertheless, the method performs remarkably well on a class of stochastic optimization problems that we discuss below.

8.2 Numerical Illustration

We illustrate our algorithm using the following two-stage stochastic program:

$$\begin{aligned} & \max \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{I} \cup \mathcal{C}} c_{ij} x_{ij} + \mathbb{E}Q(s, D) \\ \text{subject to: } & \sum_{j \in \mathcal{I} \cup \mathcal{C}} x_{ij} \leq p_i, \quad i \in \mathcal{I}, \\ & \sum_{i \in \mathcal{I}} x_{ij} - s_j = 0, \quad j \in \mathcal{I} \cup \mathcal{C}, \\ & x_{ij}, s_j \geq 0, \end{aligned}$$

where $Q(s, D)$ is the optimal value of the second stage problem:

$$\begin{aligned}
& \max \sum_{i \in \mathcal{I} \cup \mathcal{C}} \sum_{j \in \mathcal{C}} d_{ij} y_{ij} + \sum_{i \in \mathcal{C}} \sum_{l \in \mathcal{L}} r_i^l z_i^l \\
& \text{subject to: } \sum_{j \in \mathcal{C}} y_{ij} \leq s_i, \quad i \in \mathcal{I} \cup \mathcal{C}, \\
& \sum_{i \in \mathcal{I} \cup \mathcal{C}} y_{ij} - \sum_{l \in \mathcal{L}} z_j^l \geq 0, \quad j \in \mathcal{C}, \\
& z_j^l \leq D_j^l, \quad l \in \mathcal{L}, \quad j \in \mathcal{C}, \\
& y_{ij}, z_j^l \geq 0.
\end{aligned} \tag{53}$$

The problem above can be interpreted as follows. There is a set \mathcal{I} of production facilities (with warehouses) and a set of customers \mathcal{C} . At the first stage, an amount x_{ij} is transported from production facility i to a warehouse or customer location j , before the demand realizations at customer locations become known. After the realization of the demands at customer locations are observed, we move an amount y_{ij} from location i to customer location j . At each customer location we face different types of demands, indexed by $l \in \mathcal{L}$: D_j^l is the demand of type l at location j . We serve z_j^l units of demand of type l at location j ; the excess demand, if any, is lost. The production capacity of facility i is denoted p_i .

The problem above is a convex two stage stochastic programming problem. Let $\lambda(s, D)$ denote the vector of Lagrange multipliers associated with the constraints (53). Then

$$\lambda(s, D) \in \partial Q(s, D).$$

It follows that the vector η^k with coordinates

$$\eta_{ij}^k = c_{ij} + \lambda_j(s^k, D^k), \quad i \in \mathcal{I}, \quad j \in \mathcal{I} \cup \mathcal{C},$$

is a stochastic subgradient of the overall objective, provided that the demand realization D^k is sampled independently of the current approximate solution x^k and of all the past estimates used by the method.

For the first stage costs, we set $c_{ij} = c_0 + c_1 \delta_{ij}$, where δ_{ij} is the Euclidean distance between locations i and j , and c_0 can be interpreted as the unit production cost (if any) and c_1 is the transportation cost applied on ‘‘per mile’’ basis. For the second stage costs, we have:

$$d_{ij} = \begin{cases} d_1 \delta_{ij} & \text{if } i \in \mathcal{I} \\ d_0 + d_1 \delta_{ij} & \text{if } i \in \mathcal{C}. \end{cases} \tag{54}$$

d_0 represents the fixed charge for shipping a unit of the product from one customer location to another customer location, and d_1 are the costs of transportation in the second stage. For every demand type l occurring in location i , we associate a revenue r_i^l . Table 8.2 summarizes the characteristics of the problems that we use in our numerical illustration. The problems differ in the cost parameters, which determine the cost of surplus and shortfall at each location, the number of locations, which determines the dimensionality

Problem	$ \mathcal{I} \cup \mathcal{C} $	c_0	c_1	d_0	d_1	$ \Omega $
PA	50	0.0	1.0	300.0	3.0	100
PB	50	300.0	1.0	300.0	3.0	100
PC	50	600.0	1.0	300.0	3.0	100
PD	50	900.0	1.0	300.0	3.0	100
PE	50	1200.0	1.0	300.0	3.0	100
PF	50	1500.0	1.0	300.0	3.0	100
PG	50	2000.0	1.0	300.0	3.0	100
PAA	50	0.0	1.0	300.0	3.0	1000
PAB	50	300.0	1.0	300.0	3.0	1000
PAC	50	600.0	1.0	300.0	3.0	1000
PAD	50	900.0	1.0	300.0	3.0	1000
PAE	50	1200.0	1.0	300.0	3.0	1000
PAF	50	1500.0	1.0	300.0	3.0	1000
PAG	50	2000.0	1.0	300.0	3.0	1000
PBA	50	600.0	1.0	0.0	1.0	100
PBB	50	600.0	1.0	300.0	3.0	100
PBC	50	600.0	1.0	600.0	3.0	100
PBD	50	600.0	1.0	900.0	6.0	100
PBE	50	600.0	1.0	1200.0	6.0	100
PBF	50	600.0	1.0	1500.0	9.0	100
PBG	50	600.0	1.0	2000.0	9.0	100
PCA	10	600.0	1.0	300.0	3.0	100
PCB	25	600.0	1.0	300.0	3.0	100
PCC	50	600.0	1.0	300.0	3.0	100
PCD	100	600.0	1.0	300.0	3.0	100

Table 1: Problem characteristics

of the decision variables, and the number of scenarios. The relative size of the second stage costs (d_0, d_1) to the first stage costs (c_0, c_1) determines the value of making the “right” decision in the first stage. Since the costs are relative, we fixed $c_1 = 1$ and instead varied only d_1 .

We have fixed the number of scenarios to eliminate the need for statistical tests. Three sets of experiments use only 100 scenarios since these allowed us to find the optimal solution using L-shaped decomposition. The experiments with 1000 scenarios were chosen to represent problems with large numbers of scenarios. We avoided even larger problems only because the datasets became too large, and we wanted to use datasets which could be easily shared.

In a large problem instance, it is not obvious how to generate points in the feasible region of problem (8.2). Thus we only use SPAR with learning while optimizing as described in section 5. As a benchmark, we use three well-know stochastic programming algorithms: L-shaped decomposition Van Slyke & Wets (1969), stochastic decomposition Hige & Sen (1991), and cutting plane partial sampling Chen & Powell (1999).

Our focus is on the rate of convergence, measured by the improvement in the objective function as the number of iterations increases. However, we also report CPU times, since algorithms such as CUPPS (which requires scanning all scenarios at every iteration) require more effort per iteration than an algorithm such as stochastic decomposition, which only requires scanning the scenarios that have been generated up to that iteration. This difference is most notable for the large scenario runs. Our separable learning algorithm is not sensitive to the number of scenarios, and the solution times do not grow in any

Problem	Method	Number of independent demand observations								CPU time (sec.) per iteration			
		25	50	100	250	500	1000	2500	5000	100	500	1000	5000
PA	SPAR	10.18	2.39	0.81	0.37	0.21	0.15	0.07	0.02	0.036	0.057	0.070	0.083
	L-Shaped	29.88	16.95	11.68	2.95	0.44	0.03	0.00		0.937	1.907	2.410	
	CUPPS	20.74	14.87	9.86	7.22	5.21	2.39	0.18	0.00	0.056	0.178	0.378	9.406
	SD	47.63	17.90	15.64	9.94	11.42	7.56	2.52	2.48	0.054	0.207	0.465	3.038
PB	SPAR	9.28	2.41	0.77	0.34	0.23	0.26	0.11	0.04	0.032	0.050	0.062	0.074
	L-Shaped	44.19	20.93	8.28	1.54	0.49	0.05	0.00		0.937	1.945	2.465	
	CUPPS	49.43	17.68	13.58	8.63	6.17	1.95	0.29	0.00	0.057	0.178	0.378	9.494
	SD	24.76	19.44	17.80	11.25	8.56	8.54	4.12	1.62	0.057	0.209	0.466	3.126
PC	SPAR	9.99	2.60	1.18	0.48	0.26	0.30	0.12	0.05	0.030	0.048	0.060	0.076
	L-Shaped	42.56	20.30	6.07	1.49	0.52	0.04	0.00		0.950	1.980	2.510	
	CUPPS	34.93	9.91	19.30	11.71	5.09	1.38	0.32	0.00	0.057	0.178	0.377	9.382
	SD	43.18	29.81	17.94	8.09	5.91	6.25	2.73	1.02	0.056	0.207	0.466	3.112
PD	SPAR	10.72	3.07	1.64	0.58	0.39	0.44	0.14	0.10	0.029	0.046	0.055	0.063
	L-Shaped	37.07	15.26	8.16	2.77	0.50	0.00			0.963	1.987	2.510	
	CUPPS	36.36	22.47	10.99	8.10	6.30	2.25	0.21	0.00	0.058	0.178	0.376	9.431
	SD	35.37	20.05	17.52	11.36	9.14	6.53	3.08	2.28	0.055	0.203	0.454	3.064
PE	SPAR	10.43	3.88	2.41	0.87	0.67	0.65	0.15	0.08	0.026	0.039	0.047	0.053
	L-Shaped	36.18	18.54	6.72	1.91	0.46	0.00			0.946	1.934	2.031	
	CUPPS	41.79	30.11	22.02	12.53	9.14	2.94	0.18	0.00	0.068	0.285	0.649	2.498
	SD	57.06	40.78	25.40	21.36	23.11	12.02	4.37	4.45	0.054	0.201	0.451	3.038
PF	SPAR	9.58	3.56	3.61	1.30	0.53	0.74	0.25	0.07	0.025	0.037	0.044	0.048
	L-Shaped	29.53	23.92	11.49	1.28	0.33	0.00			0.925	1.880	1.688	
	CUPPS	36.28	32.67	21.34	17.27	9.26	2.50	0.05	0.00	0.067	0.284	0.645	2.489
	SD	25.37	38.73	22.00	18.03	23.89	25.93	9.46	3.25	0.053	0.199	0.445	3.016
PG	SPAR	8.95	4.76	4.42	1.03	0.75	0.87	0.21	0.09	0.023	0.034	0.040	0.042
	L-Shaped	40.68	25.32	1.91	0.08	0.00				0.892	0.973		
	CUPPS	38.97	37.99	6.02	5.08	4.77	0.53	0.00		0.064	0.281	0.644	2.475
	SD	40.64	65.54	11.37	25.01	9.22	6.78	1.66	1.64	0.051	0.194	0.438	2.977

Table 2: Numerical results on problems with different first stage costs

significant way with the number of iterations.

In order to measure the rate of convergence of different methods, we consider 25, 50, 100, 250, 500, 1000, 2500, 5000 independent samples (with replacement) from the discrete outcome space. In the L-shaped decomposition method we use them to construct approximations of the recourse function, say \hat{Q}^k for $k = 25, \dots, 5000$. Having constructed these approximations, we set $(x^k, s^k) = \arg \max \sum_{i,j \in \mathcal{I}} c_{ij} x_{ij} + \sum_{i \in \mathcal{I}} \hat{Q}_i^k(s_i)$ for $k = 25, \dots, 5000$. Then we compute $\sum_{i,j \in \mathcal{I}} c_{ij} x_{ij}^k + \mathbb{E}Q(s^k, \omega)$, to measure the performance of the solution (x^k, s^k) provided by the approximation \hat{Q}^k . In the iterative stochastic methods we use these observations in a sequential manner. The numbers in the tables in this section represent the percent deviation from the optimal objective value of the solution obtained with the use of k observations. In order to avoid statistical issues when comparing solution quality, we worked on problems with finite numbers of outcomes. Therefore, given a first stage solution (x^k, s^k) , we can compute the *exact* objective value corresponding to it.

Tables 2 and 3 summarize the results of the experiments which varied the first and second stage costs, respectively. Table 4 gives the results for problems with different numbers of locations (which determines the dimensionality of the problem).

We first note the ability of SPAR to achieve high quality solutions very quickly. It outperformed all the other algorithms by wide margins after 50 iterations, with the single

Problem	Method	Number of independent demand observations								CPU time (sec.) per iteration			
		25	50	100	250	500	1000	2500	5000	100	500	1000	5000
PBA	SPAR	4.19	2.01	1.28	0.91	0.63	0.43	0.31	0.24	0.042	0.066	0.076	0.093
	L-Shaped	0.00								0.251			
	CUPPS	18.41	13.27	0.00						0.060	0.174	0.364	9.252
	SD	4.65	0.24	0.01	0.19	0.00	0.17	0.01	0.00	0.058	0.197	0.434	2.892
PBB	SPAR	9.99	2.60	1.18	0.48	0.26	0.30	0.12	0.05	0.030	0.048	0.060	0.076
	L-Shaped	42.56	20.30	6.07	1.49	0.52	0.04	0.00		0.950	1.980	2.510	
	CUPPS	34.93	9.91	19.30	11.71	5.09	1.38	0.32	0.00	0.057	0.178	0.377	9.382
	SD	43.18	29.81	17.94	8.09	5.91	6.25	2.73	1.02	0.056	0.207	0.466	3.112
PBC	SPAR	11.03	2.52	1.25	0.51	0.20	0.26	0.14	0.06	0.033	0.051	0.062	0.083
	L-Shaped	49.59	22.46	3.97	2.49	0.49	0.01	0.00		0.981	2.062	2.613	
	CUPPS	26.22	16.04	16.09	11.16	8.70	3.33	0.20	0.00	0.058	0.178	0.377	9.416
	SD	21.90	28.25	10.75	14.38	9.01	6.70	1.78	1.59	0.056	0.207	0.462	3.116
PBD	SPAR	16.15	6.21	2.51	1.01	0.50	0.52	0.12	0.08	0.034	0.048	0.062	0.071
	L-Shaped	120.23	60.33	22.64	5.67	1.13	0.11	0.00		0.928	1.970	2.518	
	CUPPS	138.10	41.18	33.45	27.96	15.01	6.54	0.49	0.00	0.056	0.178	0.378	9.283
	SD	44.39	57.31	32.76	15.42	18.56	15.60	3.75	2.15	0.054	0.209	0.470	3.144
PBE	SPAR	15.97	6.11	2.63	0.97	0.44	0.44	0.09	0.11	0.030	0.047	0.058	0.071
	L-Shaped	164.19	51.01	27.91	4.99	1.13	0.09	0.00		0.926	1.959	2.509	
	CUPPS	143.50	108.47	47.62	26.60	23.21	5.03	0.63	0.00	0.055	0.177	0.379	9.264
	SD	66.89	65.16	44.30	22.61	22.39	13.23	3.69	2.70	0.053	0.208	0.470	3.164
PBF	SPAR	19.87	8.99	4.18	1.30	0.32	0.47	0.09	0.10	0.029	0.044	0.053	0.064
	L-Shaped	174.86	140.90	34.94	4.31	0.76	0.00			0.902	1.918	2.108	
	CUPPS	146.50	97.14	54.25	37.94	19.96	3.96	0.45	0.00	0.054	0.176	0.376	9.409
	SD	64.56	45.29	54.41	25.85	23.73	19.71	4.28	2.04	0.053	0.207	0.462	3.096
PBG	SPAR	19.94	9.01	4.20	1.32	0.35	0.49	0.07	0.13	0.029	0.044	0.054	0.069
	L-Shaped	187.23	151.50	38.38	5.85	1.08	0.00			0.900	1.910	2.077	
	CUPPS	124.92	52.52	53.05	44.33	11.75	6.56	0.84	0.00	0.055	0.176	0.376	9.250
	SD	40.91	59.27	30.73	30.82	23.57	16.56	4.06	2.27	0.061	0.244	0.502	3.413

Table 3: Numerical results on problems with different second stage costs.

exception of very low dimensional problems (10 locations). These numbers support the intuition that Benders-based methods will exhibit slow convergence as the dimensionality of the problem rises. Asset allocation problems arising in freight transportation can have hundreds and even thousands of dimensions.

The separable approximation procedure that we propose also appears to exhibit the fastest run times per iteration, although we have to emphasize that CPU times can depend to a large extent on how the algorithm is implemented. For these experiments, our algorithm exhibits fast convergence and fast CPU times per iteration, suggesting that it will be the fastest algorithm for this class of problems.

Table 5 reports the results of the runs with 1000 scenarios. For these runs, L-Shaped decomposition was not able to provide the optimal solution, and instead all comparisons are against the best available solution. The results again support the conclusion that the separable approximation method exhibits very fast convergence. This property is particularly useful on large, production applications where the execution time per iteration can grow substantially, and for operational applications where fast response times are necessary.

Problem	Method	Number of independent demand observations								CPU time (sec.) per iteration			
		25	50	100	250	500	1000	2500	5000	100	500	1000	5000
PCA	SPAR	18.65	12.21	7.07	2.25	0.48	0.28	0.04	0.15	0.005	0.006	0.007	0.007
	L-Shaped	3.30	0.19	0.00						0.060			
	CUPPS	5.84	4.19	0.50	0.29	0.00				0.008	0.032	0.071	2.260
	SD	45.45	9.18	11.35	2.35	2.30	1.12	0.30	0.26	0.009	0.044	0.107	0.798
PCB	SPAR	11.73	4.04	2.92	0.89	0.34	0.13	0.19	0.06	0.012	0.017	0.020	0.021
	L-Shaped	19.88	15.98	2.14	0.11	0.00				0.297	0.423	0.269	0.082
	CUPPS	8.27	13.40	4.33	4.02	1.47	0.16	0.00		0.027	0.135	0.317	1.232
	SD	40.55	29.79	22.22	12.80	4.24	4.80	1.06	0.95	0.020	0.094	0.225	1.760
PCC	SPAR	9.99	2.60	1.18	0.48	0.26	0.30	0.12	0.05	0.030	0.048	0.060	0.076
	L-Shaped	42.56	20.30	6.07	1.49	0.52	0.04	0.00		0.950	1.980	2.510	
	CUPPS	34.93	9.91	19.30	11.71	5.09	1.38	0.32	0.00	0.057	0.178	0.377	9.382
	SD	43.18	29.81	17.94	8.09	5.91	6.25	2.73	1.02	0.056	0.207	0.466	3.112
PCD*	SPAR	8.74	4.61	1.20	0.45	0.16	0.05	0.01	0.00	0.127	0.188	0.225	0.278
	L-Shaped	74.52	29.79	26.21	7.30	2.32	0.85	0.11	0.02	3.906	8.012	10.147	12.978
	CUPPS	54.59	35.54	23.99	17.58	14.68	14.13	5.36	0.91	0.223	0.675	1.422	5.227
	SD	62.63	34.82	40.73	12.14	15.22	17.43	17.49	9.42	0.203	0.580	1.305	8.692

* Optimal solution not found, figures represent the deviation from the best objective value known.

Table 4: Numerical results for problems with different numbers of locations

References

- Birge, J. & Wallace, S. W. (1988), ‘A separable piecewise linear upper bound for stochastic linear programs’, *SIAM J. Control and Optimization* **26**(3), 1–14.
- Chen, Z.-L. & Powell, W. (1999), ‘A convergent cutting-plane and partial-sampling algorithm for multistage linear programs with recourse’, *Journal of Optimization Theory and Applications* **103**(3), 497–524.
- Cheung, R. & Powell, W. B. (1996), ‘An algorithm for multistage dynamic networks with random arc capacities, with an application to dynamic fleet management’, *Operations Research* **44**(6), 951–963.
- Cheung, R. K.-M. & Powell, W. B. (2000), ‘SHAPE: A stochastic hybrid approximation procedure for two-stage stochastic programs’, *Operations Research* **48**(1), 73–79.
- Culioli, J.-C. & Cohen, G. (1990), ‘Decomposition/coordination algorithms in stochastic optimization’, *SIAM Journal of Control and Optimization* **28**, 1372–1403.
- Ding, X., Puterman, M. L. & Bisi, A. (2002), ‘The censored newsvendor and the optimal acquisition of information’, *Operations Research* **50**(3), 517–527.
- Ermoliev, Y. (1988), Stochastic quasigradient methods, in Y. Ermoliev & R. Wets, eds, ‘*Numerical Techniques for Stochastic Optimization*’, Springer-Verlag, Berlin.
- Frantzeskakis, L. & Powell, W. B. (1990), ‘A successive linear approximation procedure for stochastic dynamic vehicle allocation problems’, *Transportation Science* **24**(1), 40–57.
- Godfrey, G. A. & Powell, W. B. (2001), ‘An adaptive, distribution-free approximation for the newsvendor problem with censored demands, with applications to inventory and distribution problems’, *Management Science* **47**(8), 1101–1112.

Problem	Method	Number of independent demand observations								CPU time (sec.) per iteration			
		25	50	100	250	500	1000	2500	5000	100	500	1000	5000
PAA*	SPAR	9.81	3.14	0.71	0.31	0.20	0.11	0.04	0.00	0.024	0.028	0.032	0.052
	CUPPS	23.71	19.51	11.65	9.78	4.82	2.42	1.10	0.23	0.071	0.470	1.399	10.102
	SD	28.07	20.24	18.37	12.11	11.22	9.18	6.68	2.76	0.043	0.118	0.430	7.821
PAB*	SPAR	8.40	3.22	0.90	0.35	0.21	0.17	0.05	0.00	0.022	0.025	0.029	0.046
	CUPPS	38.14	25.89	9.31	6.87	3.01	4.01	1.96	0.24	0.071	0.470	1.391	10.082
	SD	67.77	46.70	21.50	19.33	24.35	15.17	9.75	3.83	0.044	0.120	0.437	7.885
PAC*	SPAR	7.49	2.82	0.98	0.65	0.43	0.27	0.05	0.00	0.019	0.023	0.026	0.044
	CUPPS	21.87	15.88	17.71	4.44	3.71	4.46	0.91	0.27	0.072	0.472	1.392	10.065
	SD	39.17	31.47	31.39	18.85	17.40	14.30	5.12	4.40	0.045	0.119	0.435	7.915
PAD*	SPAR	8.40	3.63	1.28	1.08	0.46	0.30	0.08	0.00	0.018	0.021	0.023	0.038
	CUPPS	37.94	16.71	12.30	7.13	6.50	2.42	0.90	0.42	0.072	0.471	1.390	10.071
	SD	58.13	49.91	43.14	30.07	22.58	10.11	7.15	3.66	0.042	0.117	0.434	7.830
PAE*	SPAR	7.55	4.31	1.28	1.35	0.55	0.29	0.10	0.00	0.016	0.019	0.021	0.033
	CUPPS	40.23	27.44	20.24	7.37	5.09	2.42	2.67	0.30	0.072	0.471	1.391	10.068
	SD	20.32	17.11	12.33	9.01	6.97	7.57	5.11	5.27	0.043	0.121	0.428	7.965
PAF*	SPAR	7.14	4.21	1.79	1.62	0.90	0.42	0.19	0.00	0.015	0.017	0.018	0.030
	CUPPS	36.42	25.06	38.47	16.22	9.05	4.19	3.08	0.28	0.071	0.470	1.390	10.065
	SD	69.95	72.85	50.15	42.10	11.32	9.82	4.94	2.12	0.047	0.118	0.434	7.991
PAG*	SPAR	7.60	5.06	2.63	2.38	0.98	0.22	0.15	0.20	0.013	0.014	0.015	0.023
	CUPPS	44.74	15.86	7.49	4.51	1.49	4.69	1.14	0.00	0.069	0.472	1.390	10.076
	SD	62.58	40.47	31.06	25.24	16.83	17.91	11.52	5.74	0.044	0.116	0.436	7.784

* Optimal solution not found, figures represent the deviation from the best objective value known

Table 5: Numerical results on problems with different first stage costs - 1000 scenarios.

- Higle, J. & Sen, S. (1991), ‘Stochastic decomposition: An algorithm for two stage linear programs with recourse’, *Mathematics of Operations Research* **16**(3), 650–669.
- Kleywegt, A. J., Shapiro, A. & Homem-De-Mello, T. (2002), ‘The sample average approximation method for stochastic discrete optimization’, *SIAM Journal on Optimization* **12**, 479–502.
- Kushner, H. J. & Yin, G. G. (1997), *Stochastic Approximation Algorithms and Applications*, Springer-Verlag.
- Laporte, G. & Louveaux, F. (1993), ‘The integer l-shaped method for stochastic integer programs with complete recourse’, *Operations Research Letters* **13**(3), 133–142.
- Porteus, E. L. (1990), *Handbooks in Operations Research and Management Science: Stochastic Models*, Vol. 2, North Holland, Amsterdam, chapter Stochastic Inventory Theory, pp. 605–652.
- Powell, W. B. & A., C. T. (1998), ‘Dynamic control of logistics queueing network for large-scale fleet management’, *Transportation Science* **32**(2), 90–109.
- Powell, W. B. & Cheung, R. (1994), ‘Stochastic programs over trees with random arc capacities’, *Networks* **24**, 161–175.
- Powell, W. B. & Topaloglu, H. (to appear), Stochastic programming in transportation and logistics, in A. Ruszczyński & A. Shapiro, eds, ‘*Handbook in Operations Research and Management Science, Volume on Stochastic Programming*’, North Holland, Amsterdam.

- Righter, R. (1989), ‘A resource allocation problem in a random environment’, *Operations Research* **37**, 329–338.
- Ruszczynski, A. (1987), ‘A linearization method for nonsmooth stochastic programming problems’, *Mathematics of Operations Research* **12**(1), 32–49.
- Shapiro, A. & Homem-De-Mello, T. (2000), ‘On the rate of convergence of optimal solutions of monte carlo approximations of stochastic programs’, *SIAM Journal on Optimization* **11**, 70–86.
- Topaloglu, H. & Powell, W. B. (2001), An algorithm for approximating piecewise linear functions from sample gradients, Technical report, Princeton University, Department of Operations Research and Financial Engineering.
- Van Slyke, R. & Wets, R. (1969), ‘L-shaped linear programs with applications to optimal control and stochastic programming’, *SIAM Journal of Applied Mathematics* **17**(4), 638–663.
- Wallace, S. (1986), ‘Solving stochastic programs with network recourse’, *Networks* **16**, 295–317.
- Wallace, S. (1987), ‘A piecewise linear upper bound on the network recourse function’, *Mathematical Programming* **38**, 133–146.