

## REFINING BOUNDS FOR STOCHASTIC LINEAR PROGRAMS WITH LINEARLY TRANSFORMED INDEPENDENT RANDOM VARIABLES

John R. BIRGE \*

*Department of Industrial and Operations Engineering, University of Michigan, Ann Arbor, MI 48109, USA*

Stein W. WALLACE

*Chr. Michelsen Institute, N-5036 FANTOFT, Bergen, Norway*

Received April 1985

Revised March 1986

A linear stochastic program where the right-hand side elements are linear transformations of independent stochastic variables is considered. We show how bounds on the recourse (second-stage) problem can be found by working directly on the independent stochastic variables instead of the right-hand side elements.

655 stochastic programming • bounds • dependent right-hand side

### 1. Introduction

When stochastic programs with recourse are solved in practice, it is usually necessary to approximate the distributions of the stochastic variables. This results in a lower and an upper bound on the recourse (second-stage) problem. The approximations are refined until the bounds satisfy a user-specified stopping rule. The algorithms used to refine the approximations usually assume that the stochastic variables are independent. In this note it is shown how algorithmic obstacles can be overcome for one type of dependency, namely the case where only the right-hand side is stochastic, and where each element of the right-hand side is a linear combination of independent stochastic variables.

### 2. Stochastic programs with recourse

A stochastic program with recourse can be formulated as follows. Find the vector  $x$  to minimize

$z$  where

$$z = cx + Q(x), \quad (1)$$

subject to

$$Ax = b, \quad x \geq 0$$

and

$$Q(x) = \sum Q(x, \xi_i) p_i,$$

where

$$Q(x, \xi_i) = \min \{ qy \mid Wy = \xi_i - Tx, y \geq 0 \}.$$

Here  $p_i$  is the probability that  $\xi = \xi_i$ . These problems can be solved by the L-shaped decomposition method developed by Van Slyke and Wets [11], a specialized variant of Benders' [1] decomposition. Different algorithmic procedures based on this can be found in Wets [14,15], Wallace [12,13] and Birge [4,5].

A problem closely related to the two-stage stochastic programming problem with recourse is the distribution problem [9, ch. II]. Dependent stochastic variables have in that case been treated by Boreau [2].

In all algorithmic work so far on the recourse problem, the usual assumption is that the different elements in the  $\xi$ -vector are independent. In this

\* Supported by National Science Foundation Grant No. ECS-8304065.

paper it is instead assumed that

$$\xi = H\eta,$$

where  $H$  is some deterministic matrix and the vector  $\eta$  is a vector of independent stochastic variables. In a production context this could mean that the demand for raincoats in several neighboring cities depends on the same stochastic variables, namely some measures of the local weather conditions. The cities' dependencies on these conditions will, however, not be the same. Therefore the rows of  $H$  will not in general be linearly dependent.

### 3. Approximation schemes

If the number of possible outcomes for  $\xi$  (or  $\eta$ ) is too large,  $Q(x)$  cannot be found. Hence a lower bound  $L(x)$  and an upper bound  $U(x)$  must be constructed. These bounds can be found by approximating the distribution of the stochastic variables. A description of such approximation procedures can be found in Birge and Wets [6]. We also refer to [6] for convergence proofs, which cover all methods considered in this paper.

The basic approximation procedures partition the rectangular support of the independent stochastic variables into smaller rectangles, hereafter called *cells*. Note that since the  $\eta$ -variables are independent, the support of  $\eta$  is a rectangle. The support of the  $\xi$ -variables, on the other hand, is not a rectangle. The question now is the following. Is it best to work in the  $\xi$ -space or the  $\eta$ -space? To answer this, consider the calculations that are necessary to find the upper and the lower bounds.

Since  $Q(x)$  is convex [11], we know from Jensen's inequality that  $Q(x) \geq Q(x, \xi)$ , where  $\xi$  is the expected value of  $\xi$ . Applying this bound on each cell (letting  $\xi$  be the conditional expected value of  $\xi$  within the cell), and weighting the results on each cell by the probabilities attached to that cell, a total lower bound can be found. The following are needed to find this lower bound:

- (1) the conditional expected values of the stochastic variables within each cell,
- (2) the probability attached to each cell.

Finding an upper bound requires the computation of probabilities attached to each extreme point of every cell. In [6] these probabilities are derived as imputed measures on extreme points and good schemes are given for their computation

with independent right-hand sides. Some new work has also been done for the dependent case; see Gassmann and Ziemba [8] and Birge and Wets [7]. Hence, finding upper bounds requires

- (3) probabilities attached to the extreme points.

Finally there is the question of how to refine the approximation of the distributions if the difference between the lower and upper bounds is not within some chosen tolerance. This is usually done by partitioning one of the cells into two new cells. This refinement must choose

- (4) the cell to partition.

These 4 tasks are described below for both the  $\xi$ - and the  $\eta$ -space.

*Task I.* Since the  $\eta$ -variables are independent, it is only necessary in the  $\eta$ -space to find the expected value of one variable at a time. Finding conditional expected values in the  $\xi$ -space, however, is rather cumbersome since the variables are dependent within each cell. This problem can be regarded as purely statistical. For some distributions, it is relatively easy to solve. Most of the known results in the dependent case treat jointly distributed  $\xi$ -variables (e.g., a multinormal distribution). They do not consider  $\xi$  as the sum of a set of independent stochastic variables, because, for example, most distributions are not closed under summation.

*Task II.* The problem is approximately the same here. Calculations are easier in the  $\eta$ -space than in the  $\xi$ -space. For both Tasks I and II, the general  $\xi$ -space case requires multiple integration, which becomes extremely difficult in higher dimensions.

*Task III.* As mentioned above, there are several solutions to this problem in the independent case. Kall and Stoyan [10] proposed using the extreme point with highest value as an upper bound. In [6] a general problem is formulated that results in a unique determination of appropriate extreme point weights for cells divided into simplices. This result is also in [8] where a linear program is given for finding appropriate extreme point weights.

*Task IV.* Let  $x_n$  be the current approximation of the optimal solution. The general belief today is that the cell containing the vector  $Tx_n$  should be partitioned. The reason for this choice is twofold. First, the polyhedral function  $Q(x)$  (and also  $L(x)$  and  $U(x)$ ) is most non-linear close to  $Tx_n$ . Second, the point  $Tx_n$  most certainly is a non-differentiable point of  $Q(x, \cdot)$ . It is therefore a point at

which a partitioning would make the linear approximations more accurate. It is of course easy to find the cell containing  $Tx_r$  in the  $\xi$ -space. But where is (are) the point(s) corresponding to  $Tx_r$  in the  $\eta$ -space? To our knowledge this is an unanswered question. The next section shows how to find point(s) in the  $\eta$ -space corresponding to  $Tx_r$  in the  $\xi$ -space.

**4. Finding the partitioning point**

The question to answer in this section is as follows. Given the linear combination  $\xi = H\eta$  and the point  $Tx_r$  in the  $\xi$ -space, which point(s) in the  $\eta$ -space correspond(s) to  $Tx_r$  in the  $\xi$ -space?

**Proposition.** Consider the problem

$$\begin{aligned} \text{Min} \quad & w = ez^+ + ez^-, \\ \text{subject to} \quad & H\eta + Iz^+ - Iz^- = Tx_r, \\ & l \leq \eta \leq u, \\ & z \geq 0, \end{aligned} \tag{2}$$

where  $I$  is a unit matrix of appropriate size,  $e$  is a vector of ones and the vectors  $l$  and  $u$  represent the lower and upper bounds on the support of  $\eta$ .

Let  $w^*$  be the minimal value of  $w$  in (2), and let  $\eta^*$  be the value of  $\eta$  at the optimal solution of (2). Then, if  $w^* = 0$ ,  $\eta^*$  is a point in the support of  $\eta$  that corresponds to  $Tx_r$  in the  $\xi$ -space. If  $w^* > 0$ ,  $\eta^*$  is a point in the support of  $\eta$  that maps into a point in the support of  $\xi$  which is closest in  $l_1$  norm to  $Tx_r$ .

**Proof.** *Case 1.* If  $w^* = 0$ , we have found a point  $\eta^*$ , such that  $H\eta^* = Tx_r$  and  $l \leq \eta^* \leq u$ . Hence  $\eta^*$  is a point in the support of  $\eta$  that maps into  $Tx_r$  in the support of  $\xi$ . The point  $\eta^*$  is not necessarily unique. If it is not unique, the  $\eta$  values that map into  $Tx_r$  form a convex set. It is then best to partition at an interior point in the convex set since it is not possible to partition a variable that is at one of its bounds.

*Case 2.* If  $w^* > 0$ , it was not possible to find a point in the support of  $\eta$  that mapped into  $Tx_r$ . The point  $\eta^*$  will then be such that it is a point in the support of  $\eta$  that maps into a point in the support of  $\xi$  which is closest in  $l_1$  norm to  $Tx_r$ . (Note that  $ez^+ + ez^-$  gives the distance between  $Tx_r$  and  $H\eta^*$  in terms of the  $l_1$  norm.) This completes the proof.  $\square$

Case 2 above can occur repeatedly in a two-stage stochastic programming problem if either  $cx$  in (1) is much larger than  $Q(x)$  (in which case the problem is 'almost' deterministic), or if  $\min Q(x, \xi)$  over  $\xi$  is always a corner solution.

If  $q > 0$ ,  $Tx_r$  is the minimizing point of  $Q(x_r, \cdot)$ . In this case  $\eta^*$  can be found by solving

$$\begin{aligned} \text{Min} \quad & v = qv, \\ \text{subject to} \quad & Wy - H\eta = -Tx_r, \\ & l \leq \eta \leq u, \\ & y \geq 0. \end{aligned} \tag{3}$$

We see that  $\eta^*$  corresponds to  $Tx_r$  in a different sense here than it did for problem (2). If  $v^* = 0$ ,  $\eta^*$  still maps into  $Tx_r$ , but if  $v^* > 0$ , we can no longer claim that  $H\eta^*$  is the point in the support of  $\xi$  that in  $l_1$  norm is closest to  $Tx_r$ .

**5. Finding the partition direction**

Section 4 describes procedures for finding a cell containing  $\eta^*$  that is in the support of  $\eta$  and that is partitioned to improve the bounds  $L(x)$  and  $U(x)$ . Let this cell containing  $\eta^*$  be  $S^*$ . There are many ways to partition  $S^*$ . One could, for example, split  $S^*$  along all coordinate directions. When  $\eta$  is large, it may be impractical to partition in all coordinate directions at every refinement of the bounds. Instead, we give a method for determining a single coordinate direction through which to divide  $S^*$ .

This procedure follows the development in [6]. Consider the function  $h_r(\eta) = Q(x_r, H\eta)$ . This function is piecewise linear on  $S^*$ . It is currently approximated on  $S^*$  by using its value at the conditional expectation of  $\eta$  on  $S^*$  and at the extreme points of  $S^*$ . To better approximate  $h_r$ ,  $S^*$  should be partitioned into two new cells so that  $h_r$  is as nearly linear as possible on these new cells. In other words, it should be cut across the direction in which  $h_r$  is most non-linear.

Birge and Wets [6] described a procedure for doing this in which one partitions a cell through the edge along which the subgradients of  $h_r$  vary most. This, then, represents the edge with the greatest non-linearity. We can do this for  $h_r$  provided we can find the subgradients of  $h_r$  at the extreme points of  $S^*$ . Subgradients at neighboring extreme points are compared. The pair with the greatest difference (in the direction between the

Table 1  
Comparison of alternative refinements <sup>a</sup>

Ex.	H	Non-linearity cell method				Greatest probability cell method			
		(1)	(2)	(3)	(4)	(1)	(2)	(3)	(4)
1	(1, 1, 1)	4	11	132	0.25	8	11	180	0.45
2	(2, 1, 1)	7	9	187	0.38	8	6	96	0.30
3	(1.5, 1, 1)	3	8	91	0.16	15	12	314	1.16
4	(2, 0, 1)	2	9	46	0.08	8	8	117	0.37
5	(1.5, 1.5, 0)	8	11	228	0.48	-	50+	-	0.8+
6	(1.5, 0.5, 1)	12	10	290	0.86	-	50+	-	0.8+

<sup>a</sup> The columns are (1) number of partitions, (2) number of L-shaped iterations, (3) number of simplex iterations, (4) CPU seconds. The number of simplex iterations of NCM includes iterations in problem (2).

points) provides the edge through which to split  $S^*$ .

To find the subgradients at extreme points,  $s^i$ ,  $i = 1, \dots, r$ , of  $S^*$ , we solve

$$\begin{aligned} \text{Maximize } & w^i = \pi(Hs^i - Tx_v), \\ \text{subject to } & \pi W \leq q. \end{aligned} \tag{4}$$

A subgradient of  $h_v$  at  $s^i$  is then  $\pi^i H$  where  $\pi^i$  solves (4). This is clear from the definition of  $h_v$ .

6. Numerical example

To show the usefulness of the procedures in Section 4, consider the following small example.

$$\begin{aligned} \text{Minimize } z = & x_1 + x_2 + E_{\eta} \{ 2y_1^+ + 5y_1^- + 2y_2^+ \\ & + 5y_2^- + 2y_3^+ + 5y_3^- \}, \end{aligned} \tag{5}$$

subject to

$$\begin{aligned} x_1 + x_2 & \geq 0.5, \\ x_1 + y_1^+ - y_1^- & = h_1 \eta, \\ x_2 + y_2^+ - y_2^- & = h_2 \eta, \\ x_1 + x_2 + y_3^+ - y_3^- & = h_3 \eta, \\ x_1, x_2, y_1^+, y_1^-, y_2^+, y_2^-, y_3^+, y_3^- & \geq 0, \end{aligned}$$

and  $\eta$  is uniform [0, 2].

Problem (5) is an example of a simple recourse problem. In Table 1, (5) is solved using the L-shaped code [5] and the general refinement scheme presented in [3]. The method for partitioning through  $\eta^*$  as described above (Non-linearity Cell Method (NCM)) is compared against the naïve approach (Greatest Probability Cell Method (GPCM)) of partitioning the cell with the greatest

probability. The table shows the merit of our method relative to the naïve approach. Results are presented for varying values of  $H = (h_1, h_2, h_3)$ . In each of the examples, NCM required fewer partitions. In all but one example, NCM was also faster. Note, however, that in that example, NCM had achieved a tighter bound than GPCM when it terminated. In two of the examples, GPCM required more than fifty major iterations of the L-shaped code and was stopped before achieving convergence.

References

- [1] J.F. Benders. "Partitioning procedures for solving mixed-variables programming problems", *Numerische Mathematik* 4, 238-252 (1962).
- [2] B. Bercanu, "On stochastic linear programming distribution problems, Stochastic technology matrix", *Z. Wahrscheinlichkeitstheorie u. verw. Geb.* 8, 148-152 (1967).
- [3] J.R. Birge, "Using sequential approximations in the L-shaped and generalized programming algorithms for stochastic linear programs", Technical Report 83-12, Department of Industrial and Operations Engineering, The University of Michigan, Ann Arbor, MI, 1983.
- [4] J.R. Birge, "Decomposition and partitioning methods for multistage stochastic linear programs", *Operations Research* 33, 989-1007 (1985).
- [5] J.R. Birge, "An L-shaped method computer code for multi-stage stochastic linear programs", in: Y. Ermoliev and R.J.-B. Wets, eds., *Numerical Methods in Stochastic Programming*, Springer, Berlin, to appear.
- [6] J.R. Birge and R.J.-B. Wets, "Designing approximation schemes for stochastic optimization problems, in particular for stochastic programs with recourse", Working Paper WP-83-111, IASA, Laxenburg, 1983.
- [7] J.R. Birge and R.J.-B. Wets, "Generating upper bounds on the expected value of a convex function with applications to stochastic programming", Technical Report 85-14, Department of Industrial and Operations Engineering, The University of Michigan, Ann Arbor, MI, 1985.

- [8] H. Gassmann and W.T. Ziemba, "A tight upper bound for the expectation of a convex function of a multivariate random variable", to appear in *Mathematical Programming Study*.
- [9] P. Kall, *Stochastic Linear Programming*, Springer, Berlin, 1976.
- [10] P. Kall and D. Stoyan, "Solving stochastic programming problems with recourse including error bounds". *Math. Operationsforsch., Statist., Ser. Optimization* **13**, 431-447 (1982).
- [11] R. Van Slyke and R.J.-B. Wets, "L-shaped linear programs with applications to optimal control and stochastic programming", *SIAM J. Appl. Math.* **17**, 638-663 (1969).
- [12] S.W. Wallace, "A two-stage stochastic facility location problem with time-dependent supply", in: Y. Ermoliev and R.J.-B. Wets, eds., *Numerical Methods in Stochastic Programming*, Springer, Berlin, to appear.
- [13] S.W. Wallace, "Solving stochastic programs with network recourse", Report No. 852330-1, Chr. Michelsen Institute, Bergen, 1985, to appear in *Networks*.
- [14] R.J.-B. Wets, "Large scale programming techniques in stochastic programming", Working Paper WP-84-90, IIASA, Laxenburg, 1984, to appear in: Y. Ermoliev and R.J.-B. Wets, eds., *Numerical Methods in Stochastic Programming*, Springer, Berlin.
- [15] R.J.-B. Wets, "Algorithmic procedures for stochastic optimization", Report No. 842650-2, Chr. Michelsen Institute, Bergen, 1984.