

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

JOHN 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

Technical Report # 85-11

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

JOHN 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

Abstract: We consider a linear stochastic program where the elements of the right-hand side are linear transformations of independent stochastic variables, and show how bounds on the recourse (second-stage) problem can be found by working directly on the independent stochastic variables, rather than on the elements of the right-hand side themselves.

Keywords: 655 Stochastic programming, bounds, dependent right-hand side.

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

Reprint order form can be sent to Stein W. Wallace, address as above.

1 INTRODUCTION

When stochastic programs with recourse are solved in practice, it is usually assumed that the stochastic variables are independent. In this note we show how algorithmic obstacles can be overcome for one type of dependency. We consider the case where only the right-hand side is stochastic, and assume that each element of the right-hand side is a linear combination of independent stochastic variables.

2 PROBLEM FORMULATION

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

$$Ax = b$$

$$x \geq 0$$

and

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

where

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

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 [8]. This decomposition principle is a specialized variant of Benders decomposition [1]. Different algorithmic procedures based on this can be found in Wets [11,12], Wallace [9,10] and Birge [3,4].

In all algorithmic work so far, the usual assumption is that the different elements in the ξ -vector are independent. Let us now drop this assumption, and instead assume that

$$\xi = H\eta$$

where H is some deterministic matrix and the vector η 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 ξ (or η) is too large, we cannot find $Q(x)$, but have to work with lower bounds $L(x)$ and upper bound $U(x)$. These bounds can be found by approximating the distribution of the stochastic variables. A description of such procedures can be found in Birge and Wets [5].

The distributions are approximated by partitioning the support of the stochastic variables into smaller rectangles, hereafter called cells. Note that since the η -variables are independent, the support of η is a rectangle. The support of the ξ -variables, on the other hand, is not a rectangle. The question now is the following. Should we work in the ξ -space or the η -space? In order to answer this, let us consider the calculations that are necessary in order to find the upper and the lower bounds.

To find the lower bounds we shall need:

- I The conditional expected values within each cell.
- II The probability attached to each cell.

To find the upper bound, on the other hand, might require that some probabilities attached to each extreme point of every cell is computed. See [5, Section 5] for a general discussion of this problem. In the case of independent right-hand sides, very good schemes exist. Some new work has also been done for the dependent case, see Gassmann and Ziemba [6]. Hence we might need to calculate

- III Probabilities attached to the extreme points.

Finally there is the question of how to refine the approximation of the distributions if the lower and upper bounds are not good enough. This is usually done by partitioning one of the cells into two new cells. The question to answer is

- IV Which cell should we partition?

Let us now see how these 4 questions can be answered depending on whether we work in the ξ - or the η -space.

Question I. This is very simple in the η -space since the variables are independent. All we need to do is to find the expected value of one variable at a time. On the other hand, finding the conditional expected value in the ξ -space is rather cumbersome since the variables are dependent within each cell. This problem can, however, be regarded as a purely statistical problem, and for some distributions the problem is relatively easy to solve. Most of the known results in the dependent case treat the case when the ξ -variables are jointly distributed (e.g. a multinormal distribution) and not when ξ is the sum of a set of independent stochastic variables. One problem is, for example, that most distributions are not closed under summation.

Question II. The problem is approximately the same here. Things are easier in the η -space than in the ξ -space. In both cases, the general case in the ξ -space requires multiple integration, which becomes extremely difficult in higher dimensions.

Question III. As mentioned above, this problem is easy in the independent case. Kall and Stoyan [7] proposed using the extreme points with highest value as an upper bound, and in [5] is formulated a general problem that results in a unique determination of appropriate extreme points weights for cells divided into simplices. This result is also in [6] where a linear program that can be solved for finding appropriate extreme point weights is provided.

Question IV. The general belief today is that we should partition the cone which contains the vector Tx_v where x_v is the current approximation of the optimal solution. The reason is twofold. First, the polyhedral function $Q(x)$ (and also $L(x)$ and $U(x)$) is most non-linear close to Tx_v . Second, the point Tx_v most certainly is a non-differentiable point of $Q(x)$. It is therefore a point at which we would like to partition so that our linear approximations become more accurate. It is of course easy to find in which cell in the ξ -space Tx_v can be found. But where is (are) the point(s) corresponding to Tx_v in the η -space? To our knowledge this is an unanswered question. We shall in the next section show how the point(s) in the η -space that correspond(s) to the point Tx_v in the ξ -space can be found.

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_v in the ξ -space. Which point(s) in the η -space correspond(s) to Tx_v in the ξ -space?

In order to answer this question, we have to recapture the importance of the point Tx_v . First note that it does not represent the minimizing point of $Q(x_v, \dots)$. Figure 1 illustrates this in the one-dimensional case.

But although Tx_v is not the minimizing point, it is a very interesting point because in almost all cases $Q(x_v, \dots)$ will be non-differentiable here. In other words, if a parametric analysis is performed on ξ , a pivot will have to take place when ξ reaches the value Tx_v . This is a

very important observation. The vector Tx_v represents the only point where a pivot will have to be performed whichever element of ξ the parametric analysis is performed on. (We cannot guarantee, however, that Tx_v is in the support.)

Fig. 1. Possible form of $Q(x_v, \xi)$.

Hence, if we partition $[1, u]$ in Figure 1 into $[1, Tx_v]$ and $[Tx_v, u]$ both the lower and the upper bound will become exact. In the general case, the bounds will become better, and Tx_v will be the best point to partition, but the bounds will not become exact because of other nonlinearities elsewhere.

Let us then return to the question of where to partition in the η -space when Tx_v is known in the ξ -space. There are several cases to consider here. First let us split the discussion into three parts. When doing this we shall assume that H has m rows and n columns and that $\text{rank } H = \min\{m, n\}$, i.e. if $m \leq n$ all the rows of H are linearly independent and if $m \geq n$ all the columns of H are linearly independent.

1. m = n.

In this case we simply have that $\eta^* = H^{-1}Tx_v$ is the best place to partition. If this is not in the support of η , then we partition the cell adjacent to η^* which has the most "non-linearity" according to the partition direction finding methods of Section 5.

2. m < n.

In this case, the number of η -variables is larger than the number of ξ -variables, a possible but rather improbable situation in most applications, as we see it. What we seek in this case is some η such that $H\eta = Tx_v$, $l \leq \eta \leq u$ where the vectors l and u represent the lower and upper bounds on the support of η . The problem is not necessarily feasible in this case. But if it is not, we know that Tx_v is outside the support of ξ . This is not an unusual case, it can happen if cx in (1) is much larger than $Q(x)$ or if $\text{Min } Q(x, \xi)$ over ξ is always a corner solution.

Consider the following program

$$\text{Min } w = ez^+ + ez^-$$

subject to

$$H\eta + Iz^+ - Iz^- = Tx_v \quad (2)$$

$$l \leq \eta \leq u$$

$$z \geq 0$$

where I is a unit matrix of appropriate size. We shall in the following denote by w^* the minimal value of w in (2). We shall also let η^* be the value of η at the optimal solution of (2).

If Tx_v is inside the support of ξ , $w^* = 0$ will be the optimal solution of (2). The corresponding η^* will be a point in the support of η that maps into Tx_v in the support of ξ .

The optimal solution above is in general not unique. If it is not, the η -values that map into Tx_v is a convex set. We then suggest to find an interior point in this set as a partitioning point so that we have the possibility of partitioning in all directions. (It is not possible to partition on a variable that is on one of its bounds.)

If Tx_v is outside the support of ξ , w^* in (2) will be larger than zero. In this case the optimal solution η^* will represent a point in the support of η that maps into the point in the support of ξ which is closest to Tx_v in the sense of the l_1 norm.

In all cases, (2) should therefore give us a useful point in the support of η at which we can partition.

3. $m > n$.

In the previous case there will always be a point in the η -space (but possibly outside the support of η) that maps into Tx_v . When $m > n$ this is no longer the case.

Consider again problem (2). Still we have that $w^* = 0$ if Tx_v is in the support of ξ . (The support of ξ is a part of a n -dimensional hyperplane in R^m .) The unique point η^* is again an interesting partitioning point.

If $w^* > 0$, Tx_v is either inside the range of H , but outside the map of the support of η , or Tx_v is outside the range of H . If we replace $l < \eta < u$ in (2) by $\eta \in R^m$ we can decide if Tx_v is in the range of H or not.

In both cases η^* will correspond to a point in the support of η that maps into a point in the support of ξ which is closest to Tx_v in the sense of the l_1 norm. In these cases η^* might not be unique.

This should fully describe the different possible situations. By solving (2) we get a point η^* in the support of η where it is relevant to partition.

We mentioned earlier in this section that Tx_v in general does not correspond to the minimizing point of $Q(x_v, \cdot)$. There is one exception. Assume that $q > 0$. In this case $\text{Min } Q(x_v, \cdot) = Q(x_v, Tx_v) = 0$. If this is the case we can choose to solve the following problem in order to find η^* .

$$\text{Min } v = qy$$

subject to

$$Wy - H\eta = -Tx_v \quad (3)$$

$$l \leq \eta \leq u$$

$$y \geq 0$$

Problem (3) will result in a point in the support of η that minimizes $Q(x_v, \cdot)$. If $v^* = 0$, Tx_v was in the support of ξ , otherwise it was not.

5. FINDING THE PARTITION DIRECTION

Section 4 describes procedures for finding a cell S^* (containing η^*) that is in the support of η and that will be partitioned to improve the bounds $L(x)$ and $U(x)$. There are however many ways to partition S^* . We could, for example, split S^* along all coordinate directions, but, when n is large, it may be impractical to do this at every refinement of the bounds. Instead, we describe here a method for determining a single coordinate direction through which to divide S^* .

This procedure follows the development in [5]. Consider the function $h_v(\eta) = Q(x_v, H\eta)$. This function is piecewise linear on S^* and we are currently approximating it on S^* by using its value at the conditional expectation of η on S^* and at the extreme points of S^* . To better

approximate h_v , we would like to split S^* into two new cells so that h_v is as nearly linear as possible on these new cells. In other words, we would like to cut across the direction in which h_v is most non-linear.

Birge and Wets described a procedure for doing this in which one partitions a cell through the edge along which its subgradients vary most. This then represents the edge with the greatest non-linearity. We can do this for h_v provided we can find the subgradients of h_v at the extreme points of S^* . Subgradients at neighboring extreme points would be compared and the pair having the greatest difference (in the direction between the points) would give the edge through which to split S^* .

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

$$\text{Maximize } w^i = \pi(Hs^i - Tx_v)$$

subject to (4)

$$\pi W \leq q.$$

A subgradient of h_v at s^i is then $\pi^i H$ where π^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, we consider the following small example.

$$\text{Minimize } z = x_1 + x_2 + E_{\eta} \{2y_1^+ + 5y_1^- + 2y_2^+ + 5y_2^- + 2y_3^+ + 5y_3^-\}$$

subject to

$$\begin{aligned} x_1 + x_2 &\geq 0.5 \\ x_1 + y_1^+ - y_1^- &= h_1\eta \\ x_2 + y_2^+ - y_2^- &= h_2\eta \\ x_1 + x_2 + y_3^+ - y_3^- &= h_3\eta \\ x_1, x_2, y_1^+, y_1^-, y_2^+, y_2^-, y_3^+, y_3^- &\geq 0 \end{aligned} \tag{5}$$

and η is uniform $[0, 2]$.

Problem (5) is an example of a simple recourse problem. In the table below, we solve (5) using the L-shaped code [4] and the general refinement scheme presented in [2]. We present two possibilities for partitioning the support of η . The first (Method 1) solves (2) and partitions through η^* as described above. The second (Method 2) partitions the subinterval with the greatest probability attached to it. The table presents results for varying values of $H = (h_1, h_2, h_3)$. In each of the examples, Method 1 required fewer partitions. In all but one example, Method 1 was also faster. We note, however, that in that example Method 1 had achieved a tighter bound than Method 2 when it terminated. In two of the examples, Method 2 required more than fifty major iterations of the L-shaped code and was stopped before achieving convergence.

Table I. Comparison of alternative refinements. The columns are I - Number of partitions, II - Number of L-shaped iterations, III - Number of simplex iterations, IV - CPU seconds. The number of simplex iterations of Method 1 includes iterations in Problem (2).

Ex	H	Method 1				Method 2			
		I	II	III	IV	I	II	III	IV
1	(1,1,1)	4	11	132	.25	8	11	180	.45
2	(2,1,1)	7	9	187	.38	8	6	96	.30
3	(1.5,1,1)	3	8	91	.16	15	12	314	1.16
4	(2,0,1)	2	9	46	.08	8	8	117	.37
5	(1.5,1.5,0)	8	11	228	.48	-	50+	-	.8+
6	(1.5,0.5,1)	12	10	290	.86	-	50+	-	.8+

7 REFERENCES

- [1] J. F. Benders, Partitioning procedures for solving mixed-variables programming problems. Numerische Mathematic 4 (1962) 238-252.
- [2] J. R. Birge, Using sequential approximations in the L-shaped and generalized programming algorithms for stochastic linear programs. Department of Industrial and Operations Engineering, The University of Michigan, Technical Report 83-12 (1983).
- [3] J. R. Birge, Decomposition and partitioning methods for multi-stage stochastic linear programs. Operations Research, to appear.
- [4] 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, to appear.
- [5] 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, IIASA, Laxenburg, Austria (1983).
- [6] 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.
- [7] P. Kall and D. Stoyan, Solving stochastic programming problems with recourse including error bounds. Math. Operationsforsch., Statist., Ser. Optimization 13 (1982) 431-447.
- [8] 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 (1969) 638-663.
- [9] 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, to appear.

[10] S. W. Wallace, On network structured stochastic optimization problems. Report no. 842555-8, Chr. Michelsen Institute, Bergen, Norway (1984).

[11] R. J-B. Wets, Large scale programming techniques in stochastic programming. Working Paper WP-84-90, IIASA, Austria (1984). To appear in a volume on stochastic programming edited by Prekopa and Wets.

[12] R. J-B. Wets, Algorithmic procedures for stochastic optimization. Report no. 842650-2, Chr. Michelsen Institute, Bergen, Norway (1984).

