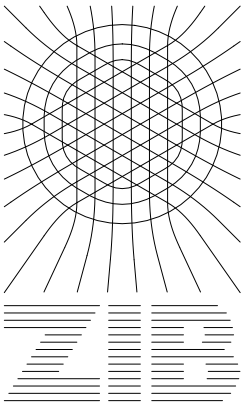
Konrad-Zuse-Zentrum für Informationstechnik Berlin
Heilbronner Str. 10, D-10711 Berlin - Wilmersdorf



Rüdiger Schultz

Discontinuous Optimization Problems in Stochastic Integer Programming

Preprint SC 95–20 (September 1995)

Discontinuous Optimization Problems in Stochastic Integer Programming*

Rüdiger Schultz

Abstract

Integer stochastic linear programming is considered from the viewpoint of discontinuous optimization. After reviewing solution approaches via mollifier subgradients and decomposition we outline how to base a solution method on efficient pointwise calculation of the objective employing computer algebra.

Key Words: Stochastic integer programming, decomposition, Gröbner bases

1991 Mathematics Subject Classification: 90 C 15, 90 C 11, 90 C 31

* presented in the Minisymposium "Stochastic Optimization" at the Third International Congress on Industrial and Applied Mathematics, Hamburg, July 1995.

This research was supported by the Deutsche Forschungsgemeinschaft.

1 Introduction

We consider the following linear stochastic program with integer recourse

$$\min\{c^T x + Q(x) : x \in C\} \tag{1.1}$$

where

$$Q(x) = \int_{\mathbb{R}^s} \Phi(z - Ax)\mu(dz) \tag{1.2}$$

and

$$\Phi(t) = \min\{q^T y : Wy \geq t, y \in \mathbb{Z}_+^{\bar{m}}\}. \tag{1.3}$$

Here, $c \in \mathbb{R}^m, C \subset \mathbb{R}^m$ is a non-empty polyhedron, $q \in \mathbb{R}^{\bar{m}}, W \in L(\mathbb{R}^{\bar{m}}, \mathbb{R}^s)$ is an integral matrix, and μ is a Borel probability measure on $\mathbb{R}^s$. Basically, we assume that for each $t \in \mathbb{R}^s$ there exists a $y \in \mathbb{Z}_+^{\bar{m}}$ such that $Wy \geq t$, that there exists a $u \in \mathbb{R}_+^s$ such that $W^T u \leq q$ and that $\int_{\mathbb{R}^s} \|z\|\mu(dz) < \infty$.

Problem (1.1) - (1.3) arises as a deterministic equivalent to a random mixed-integer linear program where decisions x and y have to be taken before and after the realization of the random parameter z, respectively. Compared with conventional stochastic programs with linear recourse it is integrality of the recourse variable y that is specific with (1.1) - (1.3) and prevents the application of standard techniques. For further details on modeling in recourse stochastic programming we refer to [6].

In [8] it is shown that the above basic assumptions imply Q to be real-valued and lower semicontinuous on $\mathbb{R}^m$. As a value function of an integer linear program, Φ is typically discontinuous. If μ is discrete with finite support then the function Q computes as a convex combination of discontinuous functions. Continuity of Q is gained when μ has a density (Proposition 3.2 in [8]). In that case, however, the integral in (1.2) causes tremendous numerical difficulties, since the dimension of the space where μ lives on is too large for numerical integration procedures. In [8] it is also shown (Proposition 4.1) that (1.1) - (1.3) is stable in that, with suitable topologies, optimal values and optimal solutions are (semi-)continuous (multi-)functions of the probability measure μ. Hence, a discrete measure μ in (1.1) - (1.3) is acceptable both from the theoretical and the practical side. Then, (1.1) - (1.3) is a discontinuous minimization problem. In what follows we discuss three approaches to solving this non-standard minimization problem: mollifier subgradients [4], decompositon [1] and pointwise computation of the objective [9].

2 Mollifier Subgradients

In [4], discontinuous minmization is tackled by averaging the objective via convolution with smooth kernels (mollifiers). Under mild assumptions met in the present context, the epigraph of the original objective is the Kuratowski set limit of the epigraphs of the averaged functions (epi-convergence, Theorem 3.7 in [4]). Optimal values and optimal solutions of the approximate

minimization problems hence converge to those of the original one. The averaged objectives being at least locally Lipschitzian, (sub-)gradient methods can be used for their minimization.

To see the difficulties with the above method consider the example where, in (1.1) - (1.3), $\Phi(t) = \min\{y_1 + y_2 : y_1 - y_2 \geq t, y \in \mathbb{Z}^2_+\}$, $c = -\frac{1}{2}$, $C = \mathbb{R}^1_+$, $A = -1$ and μ assigns probability 1 to $z = 0$. Then $c^T x + Q(x) = -\frac{1}{2}x + \lceil x \rceil$ and any non-negative integer is a local solution to our problem. By epi-convergence, approximations sooner or later will obey as many local solutions as the original. Subgradient methods at best produce local solutions such that the global optimization issue is left open with this approach.

3 Decomposition

Let μ have mass points $z_1, \ldots, z_N$ with probabilities $p_1, \ldots, p_N$. Problem (1.1) - (1.3) then can be written as a (large-scale) mixed-integer linear program:

$$\begin{array}{lrcll} \min\{c^T x + p_1 q^T y_1 + \cdots + p_N q^T y_N \ : \ x \in C, & Ax + W y_1 & \geq & z_1, & y_1 \in \mathbb{Z}^{\bar{m}}_+, \\ & & \vdots & & \\ & Ax + W y_N & \geq & z_N, & y_N \in \mathbb{Z}^{\bar{m}}_+\} \end{array}$$

Decomposition approaches take advantage of the fact that, for x fixed, the above minimization separates into

$$\Phi(z_i - Ax) = \min\{q^T y_i : W y_i \geq z_i - Ax, y_i \in \mathbb{Z}^{\bar{m}}_+\}, \ \ i = 1, \ldots, N.$$

Without integer requirements on the variables y_i the function

$$Q(x) = \sum_{i=1}^{N} p_i \Phi(z_i - Ax)$$

is convex and with $d_j (j = 1, \ldots, J)$ denoting the vertices of $\{u \in \mathbb{R}^s_+ : W^T u \leq q\}$ it holds

$$\Phi(z_i - Ax) = \max_{j=1,\ldots,J} d_j^T (z_i - Ax).$$

Decomposition methods for the non-integer version of problem (1.1) - (1.3) then in principle proceed as follows (cf. e.g. [6]): The variable x is iterated via a master program

$$\min\{c^T x + \tilde{Q}(x) : x \in C\}$$

where $\tilde{Q}$ is a convex (e.g. piecewise linear) lower approximate of Q. Given x, the separate minimizations are conventional linear programs with differing right-hand sides. These are usually solved with dual simplex techniques yielding vertices from $\{d_1, \ldots, d_J\}$. The latter are used to update the maximum defining Φ and, thus, to update the approximate $\tilde{Q}$ (optimality cut). By our basic assumptions, the separate linear programs are always solvable. In general, this is not the case, and feasibility cuts improving the approximation of the domain of Φ can be read off the final tableau. The algorithm stops if, for the actual iterate, objective function values in the master and in the original program are the same or close to each other.

Let us now outline properties of the master program and the separate optimization problems if integrality is taken into account. This issue is treated in detail by Carøe and Tind in [1]. The following is based on this paper.

Let $t \in \mathbb{R}^s$ be arbitrary and denote

$$\mathcal{P} = \{y \in \mathbb{R}^{\bar{m}}_+ : Wy \geq t\}, \quad \mathcal{P}_I = \text{conv}\{y \in \mathbb{Z}^{\bar{m}}_+ : Wy \geq t\}.$$

If

$$\mathcal{H} = \{y \in \mathbb{R}^{\bar{m}} : \gamma^T y \geq \delta\}$$

denotes a half space such that $\mathcal{H} \supseteq \mathcal{P}$ then

$$\lceil \gamma \rceil^T y \geq \delta \ \ (\text{since } y \geq 0)$$

and

$$\lceil \gamma \rceil^T y \geq \lceil \delta \rceil \ \ (\text{since } y \in \mathbb{Z}^{\bar{m}}),$$

for all $y \in \mathcal{P}_I$. Here, $\lceil . \rceil$ denotes the componentwise integer round-up operation.

Denote

$$\mathcal{H}_I = \{y \in \mathbb{R}^{\bar{m}} : \lceil \gamma \rceil^T y \geq \lceil \delta \rceil\} \ \text{ and } \ \mathcal{P}' = \cap_{\mathcal{H} \supseteq \mathcal{P}} \mathcal{H}_I.$$

Now iterate this process by setting

$$\mathcal{P}^o = \mathcal{P}, \ldots, \mathcal{P}^{k+1} = (\mathcal{P}^k)', \ldots.$$

A theorem dating back to Chvátal and Schrijver (cf. [7] and the references therein) then says that $\mathcal{P}'$ is already given as the intersection of *finitely* many $\mathcal{H}_I$ and that there exists a $k_o \in \mathbb{N}$ (only depending on W) such that $\mathcal{P}_I = \mathcal{P}^{k_o}$. As a conclusion we obtain that there are matrices $M_1, \ldots, M_{k_o}$, only depending on W, such that $\mathcal{P}_I$ admits a representation

$$\mathcal{P}_I = \{y \in \mathbb{R}^{\bar{m}}_+ : \lceil M_{k_o} \ldots \lceil M_2 \lceil M_1 W \rceil \rceil \ldots \rceil \geq \lceil M_{k_o} \ldots \lceil M_2 \lceil M_1 t \rceil \rceil \ldots \rceil\}$$

Denoting by $d_j (j = 1, \ldots, J)$ the vertices of the polyhedron $\{u \in \mathbb{R}^s_+ : \lceil M_{k_o} \ldots \lceil M_1 W \rceil \ldots \rceil^T u \leq q\}$ we obtain the following representation for Φ:

$$\begin{aligned}
&\Phi(z_i - Ax) \\
&\quad = \min\{q^T y : \lceil M_{k_o} \ldots \lceil M_1 W \rceil \ldots \rceil y \geq \lceil M_{k_o} \ldots \lceil M_1 (z_i - Ax) \rceil \ldots \rceil, y \in \mathbb{R}^{\bar{m}}_+\} \\
&\quad = \max\{\lceil M_{k_o} \ldots \lceil M_1 (z_i - Ax) \rceil \ldots \rceil^T u : \lceil M_{k_o} \ldots \lceil M_1 W \rceil \ldots \rceil^T u \leq q, u \in \mathbb{R}^s_+\} \\
&\quad = \max_{j=1,\ldots,J} d_j^T \lceil M_{k_o} \ldots \lceil M_1 (z_i - Ax) \rceil \ldots \rceil.
\end{aligned}$$

Whereas, in the non-integer case, $\Phi(z_i - Ax)$ computes as the finite maximum of affine linear functions in x, we now end up with a finite maximum of functions (Chvátal functions) that arise from affine linear functions by taking finitely many linear combinations and integer round-ups. For the decomposition approach this has drastic consequences in that the master, formerly a convex problem, now becomes discontinuous and that the separate optimization problems now are *integer* linear programs with varying right-hand sides. Optimality cuts now are Chvátal instead of linear functions. One possibility to compute them is tracing back the rounding operations in the course of Gomory's cutting plane method (for details see [1], [7]).

4 Pointwise Computation

The above decomposition approach aims at aquiring both pointwise and global information on the value function Φ with the difficulty that a discontinuous master comes into play for which, up to now, no efficient minimization methods are known. In the present section, we describe an approach via pointwise computation of Φ which, on the one hand, uses less information on Φ but, on the other hand, leads to an algorithm for solving (1.1) - (1.3).
At first, we observe that, by the integrality of W, it holds $\Phi(t) = \Phi(\lceil t \rceil)$ for all $t \in I\!R^s$ since the feasible sets in (1.3) do not change when passing to the integer round-up.
Consider the lattice

$$\mathbf{L} = \times_{j=1}^{s} \cup_{i=1}^{N} \{-\{z_{ij}\} + Z\!\!\!Z\}$$

where $\{.\}$ denotes the fractional part. For each $\lambda \in \mathbf{L}$ there exists a $\lambda_o \in \mathbf{L}$ such that $C_\lambda = \{t \in I\!R^s : \lambda_o < t \leq \lambda\}$ does not contain further elements of $\mathbf{L}$. Of course, $\cup_{\lambda \in \mathrm{L}} C_\lambda = I\!R^s$. The introduction of $\mathbf{L}$ is motivated by the following result.

Proposition 4.1 *It holds*

$$\min\{c^T x + Q(x) : \ x \in C\}$$

$$= \min_{\lambda \in \mathrm{L}_o} \{\sum_{i=1}^{N} p_i \Phi(\lceil z_i + \lambda \rceil) + \min\{c^T x \ : x \in C, \lambda_o \leq -Ax \leq \lambda\}\}$$

where $\mathbf{L}_o = \{\lambda \in \mathbf{L} : \{x \in C : \lambda_o < -Ax \leq \lambda\} \neq \emptyset\}$.

Proof: Let us first confirm that $Q(x)$ is constant while $-Ax \in C_\lambda$. By $\lambda_o < -Ax \leq \lambda$ we have

$$z_i + \lambda_o < z_i - Ax \leq z_i + \lambda$$

implying $\lceil z_i - Ax \rceil \leq \lceil z_i + \lambda \rceil$ for all $i = 1, \ldots, N$ and all $-Ax \in C_\lambda$. To see that even equality holds assume on the contrary that there were i, j such that $n_j = \lceil z_{ij} - (Ax)_j \rceil < \lceil z_{ij} + \lambda_j \rceil$. This would imply

$$\lambda_{oj} < -(Ax)_j \leq n_j - z_{ij} < \lambda_j$$

contradicting that there are no further lattice points in C_λ. Hence for all $x \in I\!R^m$ such that $-Ax \in C_\lambda$

$$Q(x) = \sum_{i=1}^{N} p_i \Phi(z_i - Ax) = \sum_{i=1}^{N} p_i \Phi(\lceil z_i - Ax \rceil) = \sum_{i=1}^{N} p_i \Phi(\lceil z_i + \lambda \rceil).$$

This implies

$$\min\{c^T x + Q(x) : \ x \in C\}$$

$$\begin{aligned}
&= \min\{c^T x + Q(x) : x \in C, -Ax \in \cup_{\lambda \in \mathrm{L}} C_\lambda\} \\
&= \min_{\lambda \in \mathrm{L}_o} \min_x \{c^T x + Q(x) : x \in C, \lambda_o < -Ax \leq \lambda\} \\
&= \min_{\lambda \in \mathrm{L}_o} \min_x \{c^T x + \sum_{i=1}^{N} p_i \Phi(\lceil z_i + \lambda \rceil) : x \in C, \lambda_o < -Ax \leq \lambda\} \\
&= \min_{\lambda \in \mathrm{L}_o} \{\sum_{i=1}^{N} p_i \Phi(\lceil z_i + \lambda \rceil) + \min\{c^T x \ : x \in C, \lambda_o \leq -Ax \leq \lambda\}\}
\end{aligned}$$

where relaxation of the strict inequality in the last row is only possible due to the lower semi-continuity of Q quoted in the introduction. The proof is complete. ■

Dropping the integrality constraint in (1.3) we obtain the continuous relaxation of (1.1) - (1.3). By Q_R we denote the relaxed expected recourse function according to (1.2). It is well known that Q_R is convex and, in case μ is discrete, piecewise linear. By $L(\alpha)$ we denote the lower level set $\{x \in C : c^T x + Q_R(x) \leq \alpha\}$. For the subsequent considerations, instead of $L(\alpha)$ also some outer polyhedral approximate would be sufficient.

Proposition 4.2 *Let Ψ denote the solution set to (1.1) - (1.3) and $x_o \in C$. Then $\Psi \subseteq L(c^T x_o + Q(x_o))$.*

Proof: If there exists $\bar{x} \in \Psi$ with $\bar{x} \notin L(c^T x_o + Q(x_o))$ then

$$c^T x_o + Q(x_o) \geq c^T \bar{x} + Q(\bar{x}) \geq c^T \bar{x} + Q_R(\bar{x}) > c^T x_o + Q(x_o)$$

which is an obvious contradiction. ■

Proposition 4.1 reduces solving (1.1) - (1.3) to solving a series of integer and conventional linear programs differing only in their right-hand sides. For linear programs, solution techniques exploiting this similarity are well known. For integer linear programs this is different, and at the end of the paper we sketch a first method in this respect which is due to Conti/Traverso [2] and uses machinery from computational algebra. Its details are also described in [9].
Proposition 4.2 provides information on the location of the solution set Ψ in terms of lower level sets of the continuous relaxation. Based on the above statements, [9] contains an algorithm for (1.1) - (1.3) that combines efficient calculation of Φ with a searching procedure involving (approximations of) level sets of the type $L(\alpha)$. This algorithm roughly outlines as follows:

1. The optimal solution to the continuous relaxation provides an initial iteration x_{best} and an initial search region $L_{best} = L(c^T x_{best} + Q(x_{best}))$.

2. L_{best} is searched by checking lattice points λ in
$$\{\lambda \in \mathbf{L}_o : \{x \in L_{best} : \lambda_o < -Ax \leq \lambda\} \neq \emptyset\}.$$

3. Each search step consists of calculating
$$\sum_{i=1}^{N} p_i \Phi(\lceil z_i + \lambda \rceil) + \min\{c^T x \ : x \in C, \lambda_o \leq -Ax \leq \lambda\}$$
and setting
$$x(\lambda) \in \operatorname{argmin}\{c^T x \ : x \in C, \lambda_o \leq -Ax \leq \lambda\}.$$

4. If $c^T x(\lambda) + Q(x(\lambda)) < c^T x_{best} + Q(x_{best})$, then both x_{best} and L_{best} are updated accordingly; otherwise, a new lattice point λ is checked.

5. The algorithm stops if no further lattice points remain to be checked.

The algorithm is obviously finite in case L_{best} is bounded. Sufficient conditions for the latter are worked out in [9]. The above scheme conveys only the principal idea of the algorithm. Details on how to organize the search and to solve the (conventional) linear programs efficiently are elaborated in [9].

The algorithm's key issue, however, is the efficient computation of $\Phi(\lceil z_i + \lambda \rceil)$. Of course, one wants to avoid starting the optimization from the beginning any time a new argument (i.e. right-hand side in (1.3)) $\lceil z_i + \lambda \rceil$ arrives. Here a solution method proposed in [2] using computational algebra is employed.

The integer linear program $\min\{q^T y : Wy \geq t, y \in \mathbb{Z}_+^{\bar{m}}\}$ is translated into a subalgebra membership problem in a suitable ring of polynomials. Lattice points in $\mathbb{Z}_+^{\bar{n}}$ correspond to certain monomials in the polynomial ring. The ring is equipped with a monomial order that is compatible with the ranking of lattice points induced by the objective function $q^T y$. The columns of W determine a certain binomial ideal $\mathcal{I}$ in the ring. Then, a Gröbner basis $\mathcal{G}$ of $\mathcal{I}$ with respect to the mentioned monomial order is computed by Buchberger's algorithm. The elementary steps in Buchberger's algorithm are generalized divisions of multivariate polynomials. Given a finite set of polynomials, the generalized division algorithm rewrites an arbitrary polynomial as a (polynomial) linear combination of elements in the mentioned set plus a remainder. Computing the Gröbner basis $\mathcal{G}$ is expensive since, in general, exponentially many generalized divisions have to be performed. However, after having computed $\mathcal{G}$, solving the integer linear program is easy: a monomial determined by the right-hand side vector t is divded by $\mathcal{G}$. From the remainder, either an optimal solution can be read off or it can be decided that the integer linear program has no

feasible points. The right-hand side t enters the procedure only after the tremendous part of the work (computing $\mathcal{G}$) is done. In that sense, we avoid starting the entire solution process anew whenever another argument t arises.
General purpose implementations of Buchberger's algorithm are part of computer algebra packages. Recent research also focusses on implementations taking advantage of the special structure met in integer linear programming [5].
The geometric counterpart to the algebraic procedure described above is developed in [10]. The author shows that the Gröbner basis $\mathcal{G}$ geometrically corresponds to a test set ([7]) of the integer linear program involved.

Acknowledgement:

This paper has benefitted from discussions with Claus C. Carøe and Jørgen Tind (both University of Copenhagen) as well as Leen Stougie (University of Amsterdam) and Maarten H. van der Vlerk (CORE Louvain-la-Neuve).

References

[1] Carøe, C.C.; Tind, J.: L-shaped decomposition of two-stage stochastic programs with integer recourse, manuscript, Institute of Mathematics, University of Copenhagen, 1995.

[2] Conti, P.; Traverso, C.: Buchberger algorithm and integer programming, in: Proceedings AAECC-9 (New Orleans), Lecture Notes in Computer Science 539, Springer-Verlag Berlin, 1991, 130-139.

[3] Cox, D.; Little, J.; O'Shea, D.: Ideals, Varieties and Algorithms, Springer-Verlag New York, 1992.

[4] Ermoliev, Y.M.; Norkin, V.I.; Wets, R.J.-B.: The minmization of semicontinuous functions: mollifier subgradients, SIAM Journal on Control and Optimization 33(1995), 149-167.

[5] Hoşten, S.; Sturmfels, B.: GRIN: An implementation of Gröbner bases for integer programming, in: E. Balas, J. Clausen (Eds.), Integer Programming and Combinatorial Optimization, 4th International IPCO Conference Copenhagen, Denmark, May 1995, Lecture Notes in Computer Science 920, Springer-Verlag Berlin, 1995, 267-276.

[6] Kall, P.; Wallace, S.W.: Stochastic Programming, John Wiley, Chichester 1994.

[7] Schrijver, A.: Theory of Linear and Integer Programming, John Wiley, Chichester 1986.

[8] Schultz, R.: On structure and stability in stochastic programs with random technology matrix and complete integer recourse, Mathematical Programming (to appear).

[9] Schultz, R.; Stougie, L.; van der Vlerk, M.H.: Solving stochastic programs with complete integer recourse: a setup using Gröbner bases, manuscript in preparation, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 1995.

[10] Thomas, R.: A geometric Buchberger algorithm for integer programming, Mathematics of Operations Research (to appear).