Adapting an Approximate Level Method to the Two-Stage Stochastic Programming Problem

Csaba I. Fábián *
Department of Operations Research, Eötvös Loránd University
Budapest, Kecskeméti u. 10-12
H-1053 Hungary
E-mail: fabian@cs.elte.hu

Abstract

We present a decomposition method for the solution of two-stage stochastic programming problems. This is an approximate method that can handle problems with large numbers of scenarios. At the beginning, only rough approximation of the objective function is required. As the optimum is gradually approached, more and more accurate data are computed. The required accuracy is known at each step, hence efforts can be coordinated.

The present framework enables the application of interior-point methods because the convergence proof does not rely on basic solutions. Moreover, the classic discretization methods and stochastic approximation schemes naturally fit into the present framework.

Introduction

Two-stage stochastic programming problems derive from such models where decisions are made in two stages and the observation of some random event takes place in between. The second decision is made in the knowledge of the outcome of the random event. The first decision must be made in the absence of this knowledge. The customary aim is to minimize some expected cost. Comprehensive treatment of these models can be found in Kall and Wallace (1994), Prékopa (1995), Birge and Louveaux (1997).

In this paper we consider linear models and assume that there is a finite number of possible outcomes of the random event (*scenarios*). Hence the problem can be formulated as a single linear programming problem of gigantic size and a specific structure: subproblems belonging

*This work was supported by AKP, Research Fund of the Hungarian Academy of Sciences, project 98-70 2,1/31; and by OTKA, Hungarian National Fund for Scientific Research, projects T029775 and T032369.

to different scenarios (*recourse problems*) are coupled by the variables describing the first-stage decision. This structure enables the use of decomposition methods. A state-of-art summary of decomposition methods in stochastic programming can be found in Ruszczyński (1997). In order to place the new method into context, we mention the most important direct decomposition methods:

The L-Shaped Method of Van Slyke and Wets (1969) is a version of the Benders decomposition method (1962), specialized for the two-stage stochastic problem. From a convex programming view, it uses the classical cutting-plane method. Cutting-plane methods are known to be unstable, and the convergence is slow. The main difficulty implementations must encounter is the constantly growing size of the master problem. There is no effective reduction strategy, because the master problem tends to be dual degenerate. Moreover, in its original form, the scheme aggregates information obtained by the solutions of recourse problems belonging to different scenarios. This aggregation means a loss of information.

The Regularized Decomposition Method of Ruszczyński (1986) is a bundle-type method for the minimization of the sum of polyhedral convex functions over a convex polyhedron. Application of this method to the two-stage stochastic problem is described by Ruszczyński and Świetanowski (1996). This method lays the emphasis on keeping the master problem of a tractable size. The bundle-type penalty term in the objective function of the master problem eliminates dual degeneracy, and enables an effective constraint reduction strategy. Besides, the penalty term has a stabilizing effect. Moreover, the method does not aggregate information, hence data from different scenarios can be combined in various ways. (Such disaggregative methods are termed *multicut methods* after Birge and Louveaux (1988).) The numerical experience of Ruszczyński and Świetanowski indicates that the Regularized Decomposition Method has a potential for solving very large stochastic programming problems. A drawback of the method is that feasibility of the solution can only be attained by imposing explicit constraints (*feasibility cuts*) on the variables describing the first-stage decision.

The aim of the present paper is to further increase the size of the stochastic programming problems we can handle. The proposed method is less sensitive to the number of the scenarios then the classic decomposition methods.

In Section 1.1 we introduce some notation and formulate the two-stage stochastic programming problem.

In Section 1.2 we sketch the Constrained Level Method of Lemaréchal, Nemirovskii, and Nesterov (1995). This is a bundle-type method that can be presented in simple form, and proved very effective. We also sketch an approximate version of the Constrained Level Method proposed by Fábián (2000a).

In Section 2, we apply the approximate method to the two-stage stochastic programming problem. With this method, we need not introduce feasibility cuts. Moreover, the method uses resources economically: At the beginning, only rough approximation of the objective function is required. As the optimum is gradually approached, more and more accurate data are computed. This idea seems natural and, in a heuristic form, is widely used in different optimization frameworks (see, e.g. Szántai (1988)). The present procedure is, however, not

heuristic. The required accuracy is known at each step, hence coordination of efforts is possible. We can keep balance between the amount of work invested into the solution of the recourse problems on the one hand, and into the actual optimization process on the other hand.

The present framework enables the application of interior-point methods for the (approximate) solution of the recourse problems, because the convergence proof does not rely on basic solutions. Moreover, discretization methods and stochastic approximation schemes naturally fit into this framework.

A detailed version of the present material can be found in the research report Fábián (2000b).

1 Preliminaries

1.1 Notation and Problem Formulation

The first-stage and second-stage decision vectors will be denoted by $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{y} \in \mathbb{R}^s$, respectively. Assume there are N scenarios, the ith one occuring with probability p_i ($p_i > 0$, $\sum_{i=1}^{N} p_i = 1$).

There are two optimization problems to be solved. The *second-stage problem* or recourse problem is formulated by assuming that the first-stage decision has been made, and the random event has been observed. Supposing that the ith scenario has occurred ($1 \leq i \leq N$), the recourse problem is

$$\begin{array}{c} \min \ \boldsymbol{q}_i{}^T \boldsymbol{y} \\ \text{subject to} \\ T_i \boldsymbol{x} + W_i \boldsymbol{y} = \boldsymbol{\xi}_i, \\ \boldsymbol{y} \geq \boldsymbol{0}, \end{array} \tag{1}$$

where $\boldsymbol{q}_i \in \mathbb{R}^s$, $\boldsymbol{\xi}_i \in \mathbb{R}^r$ are given vectors, and $T_i \in \mathbb{R}^{r \times n}$, $W_i \in \mathbb{R}^{r \times s}$ are given matrices. (Formally, $\boldsymbol{q}$ and $\boldsymbol{\xi}$ are random vectors ; T and W are random matrices, each having N possible realizations. In case of the ith scenario ($1 \leq i \leq N$), the realizations are $\boldsymbol{q}_i, \boldsymbol{\xi}_i, T_i, W_i$, respectively.)

Let K_i denote the set of those $\boldsymbol{x}$ vectors for which the ith recourse problem (1) has a feasible solution. This is a convex polyhedron that we assume to be nonempty. For $\boldsymbol{x} \in K_i$, let $f_i(\boldsymbol{x})$ denote the optimal objective value of the recourse problem. We assume that $f_i(\boldsymbol{x}) > -\infty$. Or equivalently, we assume that the dual of the ith recourse problem has a feasible solution:

$$\begin{array}{c} \max \ \boldsymbol{z}^T(\boldsymbol{\xi}_i - T_i \boldsymbol{x}) \\ \text{subject to} \\ \boldsymbol{z}^T W_i \leq \boldsymbol{q}_i{}^T. \end{array} \tag{2}$$

Solvability of the dual problem does not depend on $\boldsymbol{x}$. The function $f_i : K_i \to \mathbb{R}$ is a *polyhedral* (i.e., piecewise linear) convex function.

The customary formulation of the *first-stage problem* that represents the first decision, is as follows:

$$\begin{array}{c} \min\ \boldsymbol{c}^T\boldsymbol{x} + \sum\limits_{i=1}^{N} p_i\, f_i(\boldsymbol{x}) \\ \text{subject to} \\ A\boldsymbol{x} = \boldsymbol{b}, \\ \boldsymbol{x} \geq \boldsymbol{0}, \\ \boldsymbol{x} \in K_i \ \ (i = 1, \ldots, N), \end{array} \tag{3}$$

where $\boldsymbol{c} \in \mathbb{R}^n$, $\boldsymbol{b} \in \mathbb{R}^m$ are given vectors, and $A \in \mathbb{R}^{m \times n}$ is a given matrix. We assume that the feasible domain is nonempty. The expectation in the objective, $F(\boldsymbol{x}) = \sum\limits_{i=1}^{N} p_i\, f_i(\boldsymbol{x})$, is called the *recourse function*. This is a polyhedral convex function with the domain $K = K_1 \cap \ldots \cap K_N$.

In this paper, we assume that the polyhedron $X = \{\boldsymbol{x} \,|\, A\boldsymbol{x} = \boldsymbol{b},\ \boldsymbol{x} \geq \boldsymbol{0}\}$ is bounded. Moreover, the domains of the functions f_i $(i = 1, \ldots, N)$ need to be extended to $\mathbb{R}^n$. It can be done by introducing additional variables in the recourse problem (1). The new variables are represented by the vector $\boldsymbol{s} = (s_1, \ldots, s_r)^T$. We add a new term to the objective function which penalizes infeasibility. Here w is a fixed (large) positive weight, and $\|\boldsymbol{s}\|_1 = \sum\limits_{j=1}^{r} |s_j|$ is the norm of the slack vector:

$$\begin{array}{c} \min\ \boldsymbol{q}_i{}^T\boldsymbol{y} + w\|\boldsymbol{s}\|_1 \\ \text{subject to} \\ T_i\boldsymbol{x} + W_i\boldsymbol{y} + \boldsymbol{s} = \boldsymbol{\xi}_i, \\ \boldsymbol{y} \geq \boldsymbol{0}. \end{array} \tag{4}$$

This extended recourse problem is solvable for any $\boldsymbol{x} \in \mathbb{R}^n$. Let $\tilde{f}_i(\boldsymbol{x})$ denote the optimal objective value. Then $\tilde{f}_i$ is a polyhedral convex function, and $\tilde{f}_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x})$ holds for any $\boldsymbol{x} \in K_i$. Moreover, given $\bar{\boldsymbol{x}} \in K_i$, it is easy to show that $\tilde{f}_i(\bar{\boldsymbol{x}}) = f_i(\bar{\boldsymbol{x}})$ holds for large enough w.

Remark 1 *In order to find such an estimate for w that does not depend on $\boldsymbol{x}$, let us consider the dual of the extended recourse problem:*

$$\begin{array}{c} \max\ \boldsymbol{z}^T(\boldsymbol{\xi}_i - T_i\boldsymbol{x}) \\ \textit{subject to} \\ \boldsymbol{z}^T W_i \leq \boldsymbol{q}_i{}^T, \\ \|\boldsymbol{z}\|_{\max} \leq w. \end{array} \tag{5}$$

Compare this to the dual of the original recourse problem, that is, to (2). The only difference is the bound $\|\boldsymbol{z}\|_{\max} \leq w$ in (5).

Considering (2) with some $\bar{\boldsymbol{x}} \in K_i$, there exists a basic feasible solution $\bar{\boldsymbol{z}}$ such that $f_i(\bar{\boldsymbol{x}}) = \bar{\boldsymbol{z}}^T(\boldsymbol{\xi}_i - T_i\bar{\boldsymbol{x}})$. If $\|\bar{\boldsymbol{z}}\|_{\max} \leq w$ holds, then $\tilde{f}_i(\bar{\boldsymbol{x}}) = f_i(\bar{\boldsymbol{x}})$ follows.

There are well-known upper bounds for the norm of a basic feasible solution of a linear programming problem (see Schrijver (1986)). In general, however, these bounds are too large for practical applications.

In many special cases of practical interest, usable bounds can be computed for the weight w:

- If the matrices W_i $(i = 1, \ldots, N)$ contain only non-negative components, then a bound can be constructed using the minimal nonzero element.
- If W_i $(i = 1, \ldots, N)$ are integer matrices, then the bound given by Klafszky and Terlaky (1992) will often be of a usable magnitude.
- Combinatorial nature of the recourse problems can often be exploited. The case of network problems was examined by Prékopa and Fábián (2001). (The aim was to handle probabilistic constraints in a model more general than the one described in Section 1.1.)

In what follows we assume that a usable weight w can be found which guarantees that $\tilde{f}_i(\boldsymbol{x}) = f_i(\boldsymbol{x})$ holds for $(\boldsymbol{x} \in K_i)$, i.e., $\tilde{f}_i$ is an extension of f_i $(i = 1, \ldots, N)$. For the sake of simplicity, we will denote the extended functions by f_i, and the expectation of the extended functions by $F(\boldsymbol{x}) = \sum_{i=1}^{N} p_i f_i(\boldsymbol{x})$. This is a polyhedral convex function. Given $\bar{\boldsymbol{x}} \in \mathbb{R}^n$, the function value $F(\bar{\boldsymbol{x}})$, and a subgradient $\nabla F(\bar{\boldsymbol{x}})$ can be computed by the solution of N extended recourse problems of the type (4). Since f_i is Lipschitz continuous with the constant $w\|T_i\|$, it follows that F is Lipschitz continuous with the constant $w \sum_{i=1}^{N} p_i \|T_i\|$.

Let $g_i(\boldsymbol{x})$ denote the measure of the inconsistency of the recourse problem, defined as follows:

$$\begin{gathered} g_i(\boldsymbol{x}) = \min \ \|\boldsymbol{s}\|_1 \\ \text{subject to} \\ T_i\boldsymbol{x} + W_i\boldsymbol{y} + \boldsymbol{s} = \boldsymbol{\xi}_i, \\ \boldsymbol{y} \geq \boldsymbol{0}. \end{gathered} \tag{6}$$

This problem is solvable for any $\boldsymbol{x} \in \mathbb{R}^n$. g_i is a non-negative polyhedral convex function, and we have $g_i(\boldsymbol{x}) = 0$ if and only if $\boldsymbol{x} \in K_i$. Let $G(\boldsymbol{x}) = \sum_{i=1}^{N} p_i g_i(\boldsymbol{x})$ denote the expectation of the inconsistency measure. This again is a polyhedral convex function. Given $\bar{\boldsymbol{x}} \in \mathbb{R}^n$, the function value $G(\bar{\boldsymbol{x}})$, and a subgradient $\nabla G(\bar{\boldsymbol{x}})$ can be computed by the solution of N linear programming problems of the type (6). Due to the non-negativity of g_i, we have that $\boldsymbol{x} \in K_1 \cap \ldots \cap K_N$ is equivalent to $G(\boldsymbol{x}) \leq 0$. This function is Lipschitz continuous with the constant $\sum_{i=1}^{N} p_i \|T_i\|$.

With the above notation, the first-stage problem (3) can be reformatted as

$$\begin{gathered} \min \ \boldsymbol{c}^T\boldsymbol{x} + F(\boldsymbol{x}) \\ \text{subject to} \\ \boldsymbol{x} \in X, \\ G(\boldsymbol{x}) \leq 0. \end{gathered} \tag{7}$$

Remark 2 *The assumption we made on the magnitude of w does not make the constraint $G(\boldsymbol{x}) \leq 0$ redundant in the above convex programming problem. Although this constraint indeed becomes redundant for large enough w, this magnitude may be excessively large. Typically this is the case if we have a large number of scenarios with small individual probabilities.*

1.2 The Constrained Level Method and an Approximate Version

The Constrained Level Method proposed by Lemaréchal, Nemirovskii, and Nesterov (1995) is a bundle-type method that can be presented in simple form, and proved very effective. We sketch the method as applied to the problem (7). The convex polyhedron X is assumed to be bounded. The real-valued convex functions F and G are Lipschitzian relative to X.

Function data are provided by an *oracle*: given a point $\bar{\boldsymbol{x}} \in X$, it returns function values $\boldsymbol{c}^T\bar{\boldsymbol{x}} + F(\bar{\boldsymbol{x}})$, $G(\bar{\boldsymbol{x}})$ and subgradients $\boldsymbol{c} + \nabla F(\bar{\boldsymbol{x}})$, $\nabla G(\bar{\boldsymbol{x}})$. The method requires no Slater assumption and the efficiency is not affected by large Lagrange multipliers.

Let $O^\star$ denote the optimal objective value of problem (7). If $O^\star$ is known in advance, then the quality of an approximate solution $\boldsymbol{x} \in X$ can be measured by

$$e(\boldsymbol{x}) = \max\left(\boldsymbol{c}^T\boldsymbol{x} + F(\boldsymbol{x}) - O^\star,\ G(\boldsymbol{x})\right).$$

E.g., $e(\boldsymbol{x}) = 0$ means that $\boldsymbol{x}$ is optimal. The original problem is equivalent to

$$\min\ e(\boldsymbol{x}) \quad \text{subject to} \quad \boldsymbol{x} \in X. \tag{8}$$

The method is iterative. Having generated a sequence of iterates, and gathered function values and subgradients at the known iterates, lower approximating models are constructed for the functions $\boldsymbol{c}^T\boldsymbol{x} + F(\boldsymbol{x})$ and $G(\boldsymbol{x})$; and for the optimum $O^\star$. Moreover, an upper approximating model is constructed for the dual function of the problem (8). Hence an upper bound is known for $e(\boldsymbol{x}^\star)$, where $\boldsymbol{x}^\star$ is the currently known best solution. This bound is called the *gap*. The gap is tightening at each iteration. By appropriate selection of the iterates, the gap tightens at a steady rate.

Lemaréchal, Nemirovskii, and Nesterov prove the following efficiency estimate: To obtain a gap smaller then ϵ, it suffices to perform

$$\kappa\ \left(\frac{D\Lambda}{\epsilon}\right)^2\ \ln\left(\frac{D\Lambda}{\epsilon}\right) \tag{9}$$

iterations, where κ is a constant that depends only on the parameters of the method ; D is the diameter of the feasible polyhedron X, and Λ is a common Lipschitz constant of the functions $\boldsymbol{c}^T\boldsymbol{x} + F(\boldsymbol{x})$ and $G(\boldsymbol{x})$. (Observe that the algorithm is not polynomial in the sense as e.g., interior-point methods for linear programming problems are polynomial. Such polynomiality would request that the estimate depends on the logarithm of the precision and not the precision itself.) Lemaréchal, Nemirovskii, and Nesterov report on successful application of the method to a variety of problems. The experimental results suggest much better practical behavior than the above estimate.

At each iteration, a linear programming and a convex quadratic programming problem needs to be solved. The current sizes of these problems depend on the size of the *bundle*, i.e., on the amount of the information used to construct the current approximating models. Lemaréchal, Nemirovskii, and Nesterov propose a bundle reduction strategy that preserves the above theoretical efficiency estimate of the method. The reduction strategy is heuristic

in the sense that there is no theoretical bound on the size of the bundle. Lemaréchal, Nemirovskii, and Nesterov also report on the practical behavior of the reduction strategy. In their experiments, the number of iterations did not increase substantially, and the size of the bundle did never exceed $2n$, where n is the dimension of the space in which the optimization is performed. (These tests were made with a simpler form of the method devised for unconstrained optimization. However, the results are instructive for the general case; because the general, constrained method consists of runs of the unconstrained method.)[1]

Fábián (2000a) proposed an inexact version of the Constrained Level Method. Here the oracle is assumed to supply approximate data: given a point $\bar{\boldsymbol{x}} \in X$ and some accuracy of $\delta > 0$, it returns linear functions $L(\boldsymbol{x})$ and $L^{\mathrm{I}}(\boldsymbol{x})$ satisfying

$$\boldsymbol{c}^T\boldsymbol{x} + F(\boldsymbol{x}) \geq L(\boldsymbol{x}) \ \ (\boldsymbol{x} \in X) \quad \text{and} \quad \boldsymbol{c}^T\bar{\boldsymbol{x}} + F(\bar{\boldsymbol{x}}) - L(\bar{\boldsymbol{x}}) \ \leq \ \delta,$$

$$G(\boldsymbol{x}) \geq L^{\mathrm{I}}(\boldsymbol{x}) \ \ (\boldsymbol{x} \in X) \quad \text{and} \quad G(\bar{\boldsymbol{x}}) - L^{\mathrm{I}}(\bar{\boldsymbol{x}}) \ \leq \ \delta.$$

($L(\boldsymbol{x})$ and $L^{\mathrm{I}}(\boldsymbol{x})$ are *approximate linear support functions* of the functions $\boldsymbol{c}^T\boldsymbol{x} + F(\boldsymbol{x})$ and $G(\boldsymbol{x})$, respectively.) Moreover, the returned linear functions are assumed to satisfy the Lipschitz condition with a constant not depending on $\bar{\boldsymbol{x}}$ or δ. (In the original method, the linear support functions inherit Lipschitz continuity from the functions $\boldsymbol{c}^T\boldsymbol{x} + F(\boldsymbol{x})$ and $G(\boldsymbol{x})$. In case of approximate linear support functions, we must ensure that there is an upper bound on the steepness of the linear functions that the oracle may return.)

Using the approximate linear support functions returned by the oracle, lower approximating models are constructed for the functions $\boldsymbol{c}^T\boldsymbol{x} + F(\boldsymbol{x})$ and $G(\boldsymbol{x})$; and for the optimum $O^\star$. Moreover, an upper approximating model is constructed for the dual function of the problem (8). The construction is similar to that of the original method, but here the inaccuracy of the different support functions must also be taken into consideration.

Before each request to the oracle, the tolerance δ to be prescribed is computed from the current size of the gap. If these tolerances are appropriately prescribed, then the gap tightens at a steady rate. Fábián (2000a) proves that the efficiency estimate (9) also holds for the approximate method (only the constant κ is different from that of the original method). The bundle reduction technique proposed for the original method can also be applied here.

2 Level Decomposition

Due to the assumptions made in Section 1.1, the Constrained Level Method can be applied to the aggregate problem (7) in a straightforward manner. Since the constraints $\boldsymbol{x} \in K_i$ $(i = 1, \ldots, N)$ are formulated as $G(\boldsymbol{x}) \leq 0$, there is no need for introducing feasibilty cuts. Hence the regularization is extended to the measure of the inconsistency. (The variation in the measure of the infeasibility is leveled off, because the steadily tightening gap means a

[1] We have also implemented the method, and have performed some preliminary experiments. Our experiments seem to confirm the cited results.

tightening bound on the measure of possible infeasibilty.) We expect that this feature speeds up convergence.

The size of the master problem is kept under control by the bundle reduction procedure. Although there is no theoretical bound for the size of the bundle, the experimental bound $2n$ is attractive, because it does not depend on the number of the scenarios.

A drawback of the method is that it does not exploit the fact that the functions $\boldsymbol{c}^T\boldsymbol{x}+F(\boldsymbol{x})$ and $G(\boldsymbol{x})$ are the sums of polyhedral convex functions. Generating a cut in the master problem requires the solution of $2N$ recourse problems, in contrast to multicut methods which require the solution of a single recourse problem per cut.

In the remaining part of the paper, we deploy the approximate version of the Constrained Level Method. The resulting method can be regarded a multicut method in a somewhat broader sense (explanation will follow later, in Remark 3).

By applying the approximate method for the solution of the master problem, we can keep balance between the amount of work invested into the computation of the $\boldsymbol{c}^T\boldsymbol{x}+F(\boldsymbol{x})$ and $G(\boldsymbol{x})$ function data on the one hand, and into the actual optimization process on the other hand. As the optimum of the master problem is gradually approached, more and more accurate approximations are computed for the functions $\boldsymbol{c}^T\boldsymbol{x}+F(\boldsymbol{x})$ and $G(\boldsymbol{x})$.

In what follows, we describe the construction of approximate linear support functions for the recourse function $F(\boldsymbol{x})$ at the point $\bar{\boldsymbol{x}}$ with the tolerance δ. (Using those, approximate support functions for the objective function $\boldsymbol{c}^T\boldsymbol{x}+F(\boldsymbol{x})$ can be constructed in the obvious way.) The constraint function $G(\boldsymbol{x})$ can be handled in a similar manner. There are two ways of effective computation:

1. Approximate solution of the recourse problems.

   Suppose we have found near-optimal solutions for the relevant recourse problems and their duals: Let $\boldsymbol{y}_i$ be a feasible solution, and $\boldsymbol{z}_i$ a dual feasible solution of the ith recourse problem, such that the LP duality gap $\boldsymbol{q}^T\boldsymbol{y}_i-\boldsymbol{z}_i{}^T(\boldsymbol{\xi}_i-T\bar{\boldsymbol{x}})$ is less then δ. Then the function

   $$L(\boldsymbol{x})=\sum_{i=1}^{N}p_i\,\boldsymbol{z}_i{}^T(\boldsymbol{\xi}_i-T\boldsymbol{x})\qquad(\boldsymbol{x}\in\mathbb{R}^n)$$

   meets the requirements. (The function $L(\boldsymbol{x})$ is Lipschitz continuous with the constant $w\|T\|$.)

   In Section 2.1, we show that approximate solutions can be effectively found by the use of modern interior-point methods.

2. Approximation of the random distribution.

   This can be done by applying the classic discretization methods and stochastic approximation schemes. These schemes replace the existing random distribution with a pair of coarse discrete distributions. Based on the coarse approximating distributions, easily computable convex polyhedral functions $\underline{F}(\boldsymbol{x})$ and $\bar{F}(\boldsymbol{x})$ are constructed, such that $\underline{F}(\boldsymbol{x})\le F(\boldsymbol{x})\le\bar{F}(\boldsymbol{x})$ holds for any $\boldsymbol{x}\in\mathbb{R}^n$.

Traditionally, these schemes are used in the following framework: Two approximating problems are constructed instead of the problem (7). The first problem is constructed by replacing the recourse function with the lower approximating function $\underline{F}(\boldsymbol{x})$. The second problem is constructed by replacing the recourse function with the upper approximating function $\bar{F}(\boldsymbol{x})$. The first problem is solved ; let $\boldsymbol{x}^\star$ denote the optimal solution. The upper approximating function $\bar{F}(\boldsymbol{x})$ is evaluated at $\boldsymbol{x}^\star$. If the difference $\bar{F}(\boldsymbol{x}^\star) - \underline{F}(\boldsymbol{x}^\star)$ is not significant, then the procedure stops. Otherwise the second problem is solved. If there is no significant difference between the respective optimum values of the first and second the problem, then the procedure stops. Otherwise, the approximating coarse distributions are refined; and the consruction is repeated.

Hence, the traditional framework prescribes the (exact) solution of a series of approximate problems. In the present framework, however, we solve the single problem (7). The approximating coarse distributions are refined in accord with the solution process: Suppose we need an approximate linear support functions for the recourse function $F(\boldsymbol{x})$ at the point $\bar{\boldsymbol{x}}$ with the tolerance δ. The function

$$L(\boldsymbol{x}) = \underline{F}(\bar{\boldsymbol{x}}) + [\nabla \underline{F}(\bar{\boldsymbol{x}})]^T (\boldsymbol{x} - \bar{\boldsymbol{x}}) \qquad (\boldsymbol{x} \in \mathbb{R}^n)$$

obviously satisfies $L(\boldsymbol{x}) \leq F(\boldsymbol{x})$. If $\bar{F}(\bar{\boldsymbol{x}}) - \underline{F}(\bar{\boldsymbol{x}}) \leq \delta$ holds, then $L(\boldsymbol{x})$ is an approximate linear support function of $F(\boldsymbol{x})$ at the point $\bar{\boldsymbol{x}}$ with the precision δ. Otherwise, the approximating coarse distributions are refined.

In Section 2.2, we show that the classic distcretization methods and stochastic approximation schemes naturally fit into the present framework.

2.1 Using Interior-Point Methods for the Approximate Solution of the Recourse Problems

There are effective interior-point methods that generate pairs solutions ; one of the pair being primal feasible, the other dual feasible. The LP duality gap (i.e., the gap between the respective objective values of the primal and the dual feasible solutions) is decreasing at each step. (When applying these methods to the solution of the recourse problems, the LP duality gap measures the accuray of the current solution.) A state-of-art survey of interior-point methods can be found in Roos, Terlaky and Vial (1997). In this book, the following efficiency estimate is proven: Let $\delta > 0$ be a prescribed accuracy. In order to decrease the LP duality gap below δ, the *primal-dual logarithmic barrier method* and the *predictor-corrector method* require $2\sqrt{d} \ln \frac{d\mu^0}{\delta}$ iterations, where d is the dimension of the problem (in our case, $\max(r, s)$), and $d\mu^0$ measures the LP duality gap at the starting point.

In practice, these methods also have a very good reputation, supported by many computational studies. This is especially true of the predictor-corrector method ; see, e.g., Andersen and Andersen (2000). In the present decomposition framework, the solution processes of the respective recourse problems can be terminated as soon as the required accuracy is reached.

Experience with interior-point methods suggests that a considerable part of the computational effort is devoted to the initial symbolic factorization, i.e., to exploring the structure of the matrix of the linear programming problem. In the case $W_1 = \ldots = W_N$, this symbolic factorization needs to be done only once.

2.2 Approximation of the Random Distribution

Assume that $\boldsymbol{q}_1 = \ldots = \boldsymbol{q}_N$, $T_1 = \ldots = T_N$, $W_1 = \ldots = W_N$, and hence only the vector $\boldsymbol{\xi}$ is random. Assume, moreover, that $\boldsymbol{\xi}$ linearly depends on a low-dimensional random vector, formally, we have

$$\boldsymbol{\xi}_i = S\boldsymbol{\eta}_i \quad (i = 1, \ldots, N),$$

where S is an $r \times \acute{r}$-matrix, and $\boldsymbol{\eta}_1, \ldots, \boldsymbol{\eta}_N$ are the possible vector values of an $\acute{r}$-dimensional discrete random vector $\boldsymbol{\eta}$. (The corresponding probabilities are $p_1, \ldots, p_N$, respectively.) Since $\acute{r}$ is assumed to be small, we can exploit nearness in the space of the $\boldsymbol{\eta}$ values by applying the classic approximation schemes of Kall (1980), Birge and Wets (1986), Frauendorfer and Kall (1988), Frauendorfer (1988). In the present case, the aim is not discretization of a continuous distribution, but the replacement of the present discrete distribution with a coarser one.

Approximation schemes divide the space of the $\boldsymbol{\eta}$ values into cells. For each cell C, the conditional expectation

$$F_C(\boldsymbol{x}) \;=\; \sum_{\boldsymbol{\eta}_i \in C} p_i \, f_i(\boldsymbol{x}) \;\Big/\; \sum_{\boldsymbol{\eta}_i \in C} p_i \tag{10}$$

is considered as a function of $\boldsymbol{x} \in \mathbb{R}^n$. (In both the numerator and the denominator, the summation goes for i $(1 \leq i \leq N)$ such that $\boldsymbol{\eta}_i$ falls into the cell C. For empty cells, the conditional expectation is defined as 0.) Obviously we have

$$F(\boldsymbol{x}) = \sum_C \; F_C(\boldsymbol{x}) \;\; \mathrm{P}(\, \boldsymbol{\eta} \in C \,) \qquad (\boldsymbol{x} \in \mathbb{R}^n).$$

Here $\mathrm{P}(\, \boldsymbol{\eta} \in C \,)$ is easy to compute, but computation of the conditional expectations would require the solution of N recourse problems. In order to facilitate easy computation, two coarsely distributed discrete random vectors are constructed instead of $\boldsymbol{\eta}$; namely $\underline{\boldsymbol{\eta}}$ for lower approximation, and $\bar{\boldsymbol{\eta}}$ for upper approximation. ($\underline{\boldsymbol{\eta}}$ has as many realizations as the number of the cells. The realization corresponding to cell C is constructed by concentrating into the barycenter those $\boldsymbol{\eta}$ realizations that fall into the cell C. The discrete random vector $\bar{\boldsymbol{\eta}}$ has as many realizations as the number of the vertices of the cells.)

The construction is such that $\underline{\boldsymbol{\eta}}$ yields lower approximating functions

$$\underline{F_C}(\boldsymbol{x}) \leq F_C(\boldsymbol{x}) \qquad (\boldsymbol{x} \in \mathbb{R}^n)$$

for each cell C ; and $\bar{\boldsymbol{\eta}}$ yields upper approximating functions

$$\bar{F_C}(\boldsymbol{x}) \geq F_C(\boldsymbol{x}) \qquad (\boldsymbol{x} \in \mathbb{R}^n).$$

Lower and upper approximating functions for the recourse function are constructed as

$$\underline{F}(\boldsymbol{x}) \;=\; \sum_{C} \; \underline{F_C}(\boldsymbol{x}) \;\; \mathrm{P}(\,\boldsymbol{\eta} \in C\,) \qquad (\boldsymbol{x} \in \mathrm{I\!R}^n),$$

$$\bar{F}(\boldsymbol{x}) \;=\; \sum_{C} \; \bar{F}_C(\boldsymbol{x}) \;\; \mathrm{P}(\,\boldsymbol{\eta} \in C\,) \qquad (\boldsymbol{x} \in \mathrm{I\!R}^n).$$

The approximation is refined by subdividing existing cells one at a time.

In the present framework, the existence of the prescribed accuracy δ helps in selecting the cells to be subdivided. The aim is to reach the prescribed accuracy with as few cells as possible. We advise a greedy approach: at each step, let us subdivide the cell whose deletion would cause the largest move in $\bar{F}(\bar{\boldsymbol{x}}) - \underline{F}(\bar{\boldsymbol{x}})$ toward the prescribed accuracy δ. (Here $\bar{\boldsymbol{x}}$ is the current iterate in the master problem.) Formally, let us select the cell for which the following product is maximal:

$$\Big(\, \bar{F}_C(\bar{\boldsymbol{x}}) - \underline{F_C}(\bar{\boldsymbol{x}}) - \delta \,\Big) \;\; \mathrm{P}(\,\boldsymbol{\eta} \in C\,).$$

If $\bar{F}_C(\bar{\boldsymbol{x}}) - \underline{F_C}(\bar{\boldsymbol{x}}) \leq \delta$ holds for each cell C, then the required accuracy has been reached.

Remark 3 *Optimal (or near-optimal) dual feasible solutions of the recourse problems can be stored and reused: let Z denote the dual feasible solutions already found. Given a point $\bar{\boldsymbol{x}} \in \mathrm{I\!R}^n$, a lower approximation of $f_i(\bar{\boldsymbol{x}})$ can be computed as*

$$\max_{\boldsymbol{z} \in Z} \; \boldsymbol{z}^T(S\boldsymbol{\eta}_i - T\bar{\boldsymbol{x}}).$$

Let $\boldsymbol{z}_i \in Z$ denote a vector maximizing the above expression for $i = 1, \ldots, N$, and define

$$L(\boldsymbol{x}) = \sum_{i=1}^{N} p_i \, \boldsymbol{z}_i{}^T(S\boldsymbol{\eta}_i - T\boldsymbol{x}) \qquad (\boldsymbol{x} \in \mathrm{I\!R}^n)$$

At any stage of the procedure, we can use $L(\boldsymbol{x})$ as a lower approximate for the recourse function.

This is a multicut method in the sense that the set Z represents individual (disaggregate) cuts for the functions $f_i(\boldsymbol{x})$ $(i = 1, \ldots, N)$. In the master problem, however, we use aggregate constraints that are specific combinations of the individual cuts.

Acknowledgement

I wish to express my thanks to András Prékopa for his suggestion that bundle methods should be examined from a stochastic programming point of view, and for his support in the project.

I am also grateful to Tamás Terlaky for his constructive remarks.

References

[1] ANDERSEN, E.D. and K.D. ANDERSEN. (2000). The MOSEK Interior-Point Optimizer for Linear Programming: An Implementation of the Homogeneous Algorithm. In: *High Perfomance Optimization* (H. Frenk, K. Roos, T. Terlaky, and S. Zhang, eds.), Kluwer Academic Publishers, Dordrecht / Boston / London, 197-232.

[2] BENDERS, J.F. (1962). Partitioning Procedures for Solving Mixed-Variables Programming Problems. *Numerische Mathematic* **4**, 238-252.

[3] BIRGE, J.R. and F.V. LOUVEAUX. (1988). A Multicut Algorithm for Two-Stage Stochastic Linear Programs. *European Journal of Operational Research* **34**, 384-392.

[4] BIRGE, J.R. and F.V. LOUVEAUX. (1997). *Introduction to Stochastic Programming.* Springer-Verlag, New York.

[5] BIRGE, J.R. and R.J.-B. WETS. (1986). Designing Approximation Schemes for Stochastic Optimization Problems, in particular for Stochastic Programs with Recourse. In: *Stochastic Programming 84* (A. Prékopa and R.J.-B. Wets, eds.), Vol. 1, *Mathematical Programming Study* **27** 54-102.

[6] FÁBIÁN, CS.I. (2000a). Bundle-Type Methods for Inexact Data. *Central European Journal of Operations Research* **8**, 35-55.

[7] FÁBIÁN, CS.I. (2000b). Bundle-Type Methods for Two-Stage Stochastic Programming Problems. *RUTCOR Research Report* 31-2000. [http://rutcor.rutgers.edu]

[8] FRAUENDORFER, K. (1988). Solving SLP Recourse Problems with Arbitrary Multivariate Distributions – The Dependent Case. *Mathematics of Operations Research* **13**, 377-394.

[9] FRAUENDORFER, K. and P. KALL. (1988). A Solution Method for SLP Recourse Problems with Arbitrary Distributions – The Independent Case. *Problems of Control and Information Theory* **17**, 177-205.

[10] KALL, P. (1980). Solving Complete Fixed Recourse Problems by Successive Discretization. In *Recent Results in Stochastic Programming, Lecture Notes in Economics and Math. Systems* **170** (P. Kall and A. Prékopa, eds.), Springer-Verlag, Berlin / New York, 135-138.

[11] KALL, P. and S.W. WALLACE. (1994). *Stochastic Programming.* John Wiley & Sons, Chichester.

[12] KLAFSZKY, E. and T. TERLAKY. (1992). On the Ellipsoid Method. *Radovi Matematicki* **8**, 269-280.

[13] LEMARÉCHAL, C., A. NEMIROVSKII, and YU. NESTEROV. (1995). New Variants of Bundle Methods. *Mathematical Programming* **69**, 111-147.

[14] PRÉKOPA, A. (1995). *Stochastic Programming.* Kluwer Academic Publishers, Dordrecht / Boston / London.

[15] PRÉKOPA, A. and CS.I. FÁBIÁN. (2001). A Solution Method for the Two-Stage Stochastic Network Design Problem. *Invited talk given at the EURO 2001 conference*, Rotterdam, July 2001.

[16] ROOS, C., T. TERLAKY, and J.-PH. VIAL. (1997). *Theory and Algorithms for Linear optimization.* John Wiley & Sons, Chichester / New York / Brisbane / Toronto / Singapore.

[17] RUSZCZYŃSKI, A. (1986). A Regularized Decomposition Method for Minimizing the Sum of Polyhedral Functions. *Mathematical Programming* **35**, 309-333.

[18] RUSZCZYŃSKI, A. (1997). Decomposition Methods in Stochastic Programming. *Mathematical Programming* **79**, 333-353.

[19] RUSZCZYŃSKI, A. and A. ŚWIETANOWSKI. (1996). On the Regularized Decomposition Method for Two-Stage Stochastic Linear Problems. *Working Paper* WP-96-014, International Institute for Applied Systems Analysis, Laxenburg, Austria.

[20] SCHRIJVER, A. (1986). *Theory of Linear and Integer Programming.* John Wiley & Sons, Chichester.

[21] SZÁNTAI, T. (1988). A Computer Code for the Solution of Probabilistic-Constrained Stochastic Programming Problems. In *Numerical Techniques for Stochastic Optimization* (Yu. Ermoliev and R.J.-B. Wets, eds.), Springer-Verlag, New York.

[22] VAN SLYKE, R. and R.J.-B. WETS. (1969). L-Shaped Linear Programs with Applications to Optimal Control and Stochastic Programming. *SIAM Journal on Applied Mathematics* **17**, 638-663.