Journal of the Korean
OR/MS Society
Vol. 22, No. 1, March 1997

A Study on Solution Methods of Two-stage Stochastic LP Problems

Sang Jin Lee*

Abstract

In this paper, we have proposed new solution methods to solve TSLP(two-stage stochastic linear programming) problems. One solution method is to combine the analytic center concept with Benders' decomposition strategy to solve TSLP problems. Another method is to apply an idea proposed by Geoffrion and Graves to modify the L-shaped algorithm and the analytic center algorithm. We have compared the numerical performance of the proposed algorithms to that of the existing algorithm, the L-shaped algorithm. To effectively compare those algorithms, we have had computational experiments for seven test problems.

1. Introduction

Business managers are required to deal with uncertainty inherent in decision making because of rapid changes in today's environments. For example, investment managers might be thought of as making portfolio decisions against alternative futures, called scenarios, which capture possible outcomes of uncertain parameters such as external cash flows, interest rates, and/or legal/policy restrictions. They must make decisions on the basis of existing information about such uncertain parameters, without the opportunity to make additional observation of future realizations. This investment problem can be formulated as a two-stage stochastic linear programming problem(TSLP problem).

In addition to financial planning problems, there are several situations that require decision making under uncertainty such as facility location, agriculture management, aircraft allocation, flood control, energy investment, gas distribution, environment and natural resources planning(see [11,12] for a detailed description of such applications).

* 국방대학원 국방관리학처 조교수

This paper has two-fold objectives. One is to propose and test new solution methods for TSLP problem. The linear cutting plane algorithm to solve TSLP problem is known for its slow convergence. (see [20,33,35]) This disadvantage has motivated researchers to develop methods which have faster convergence utilizing nonliner programming techniques such as the progressive hedging algorithm [26,28], the bundle decomposition algorithm [6,7], and the regularized decomposition algorithm [29,30,31]. Here, we will develop another nonlinear technique, "the analytic center algorithm," to solve TSLP problems. There have been some recent theoretical advances in solution methods for convex nondifferentiable problems, in particular, applying the analytic center concept. Our solution method is to combine the analytic center concept with Benders' decomposition strategy to solve TSLP problem. The other objective is to apply an idea due to Geoffrion and Graves [13] to enhance the efficiency of the L-shaped algorithm and the analytic center algorithm. To expediate their idea, the variation to the L-shaped algorithm and the relaxation strategy to the analytic center algorithm are proposed.

In this paper, we will deal with only a two-stage problem. Any multistage problem can be reformulate into an equivalent two-stage problem. An equivalent two-stage problem can be formulated by making the second stage objective include the expected value of latter stage objective. The two-stage problem forms are quite important because numerous applications require only two or three stages, either per se, or as a result of formulating an equivalent two-stage or three-stage problem for a multiperiod situation [38].

2. Two-stage Stochastic LP Problems

2.1. Mathematical Formulation of TSLP Problems

The two-stage stochastic linear program with fixed recourse was first formulated by Dantzig [8] and Beale [2]. Walkup and Wets[35] generalized it by introducing a random recourse matrix. Consider the following example of a two-stage stochastic program with random recourse:

$$
\begin{aligned}
&\min\ z = c_1x_1 + \textstyle\sum_{\omega=1}^{L} p_2(\omega)c_2(\omega)x_2(\omega) \\
&\text{s.t.} \\
&\qquad A_1x_1 \qquad\qquad\qquad\qquad = b_1 \\
&\qquad B_1(\omega)x_1 + A_2(\omega)x_2(\omega) = b_2(\omega)
\end{aligned}
\qquad (2.1)
$$

$$x_1 \geq 0,\ x_2(\omega) \geq 0,\ \omega \in \Omega=\{1, \ldots, L\},$$

where c_1 is a known vector in $\mathbb{R}^{n1}$, b_1 is a known vector in $\mathbb{R}^{m1}$, and A_1 is a known matrix in $\mathbb{R}^{m1 \times n1}$. For each ω, $c_2(\omega) \in \mathbb{R}^{n2(\omega)}$ and $b_2(\omega) \in \mathbb{R}^{m2(\omega)}$. $A_2(\omega)$ and $B_1(\omega)$ are matrices of sizes $m_2(\omega)$ x $n_2(\omega)$ and $n_2(\omega)$ x n_1, respectively, with a discrete probability distribution $p_2(\omega)$ such that the sum of $p_2(\omega)$ over all ω is equal to unity. It is worthwhile to consider discrete random variables in our research since there are several practical applications such as financial planning problems [25,26,27,40] based on discrete random variables. Moreover, usual solution procedures for stochastic programs with continuous random variables discretize the problems either by approzimation schemes (see [5,38]) or by sampling methods (see [9,10,19]). Thus, it is assumed here that the random vectors are discretly distributed with a finite number of scenarios ω and corresponding probabilities $p(\omega)$. We shall assume that problem (2.1) is solvable.

Problem (2.1) can be formulated in the following equivalent form [35]:

$$\begin{aligned} & \min\ z=c_1x_1+Q(x_1) \\ & \text{s.t.} \\ & \qquad A_1x_1=b_1,\ x_1 \geq 0, \\ & \text{where} \quad Q(x_1)=\min\left\{ \sum_{\omega=1}^{L} p_2(\omega)c_2(\omega)x_2(\omega) \mid \right. \\ & \qquad\qquad \left. A_2(\omega)x_2(\omega)=b_2(\omega)-B_1(\omega)x_1,\ x_2(\omega) \geq 0 \right\}. \end{aligned} \tag{2.2}$$

Properties of problem form (2.2) have been extensively studied in [35,36,37]. They show that $Q(x_1)$ is convex and piecewise linear. Since $Q(x_1)$ is convex, it is possible to apply an outer linearization procedure. For fixed x_1, problem (2.2) is separable into L independent second-stage linear programming problems. These subproblems are of the form:

$$\begin{aligned} q(x_1,\ \omega)= \quad & \min\ c_2(\omega)x_2(\omega) \\ & \text{s.t.} \\ & A_2(\omega)x_2(\omega)=b_2(\omega)-B_1(\omega)x_1 \\ & x_2(\omega) \geq 0. \end{aligned} \tag{2.3}$$

To enable a later result we will assume that $A_2(\omega)$ has full row rank. In particular we assume that the feasible region for the dual of problem (2.3) possesses extreme points (i.e., contains no lines). It can be shown that $q(x_1, \omega)$ is a convex piecewise linear function of x_1 [35,36]. Each second-stage subproblem affects the first-stage decision vector, x_1, in two ways. First, the first-stage decision must be in an acceptable set for all subproblems because all subproblems should be feasible. It is necessary for x_1 to be selected so that the equations $A_2(\omega)x_2(\omega)=b_2(\omega)-B_1(\omega)x_1$ are solvable in nonnegative $x_2(\omega)$ for all scenarios $\omega \in \Omega$. When x_1 is temporarily held fixed for certain values, the feasibility of each subproblem (2.3) should be tested. Some subproblems may be infeasible for some fixed x_1 values, which limits the set of acceptable first-stage solutions. Second, the expected value of the second-stage subproblems may vary according to each selection of a vector x_1.

2.2. The L-shaped Algorithm

A TSLP problem can be solved by the so-called L-shaped algorithm which is directly related to Benders' decomposition approach. The L-shaped algorithm given here is actually a variant of the original one [34] in the sense that a random recourse matrix is allowed. The L-shaped algorithm forms a linear approximation to problem(2.2) using outer linearization of $Q(x_1)$. Two types of cuts (i.e., constraints) are sequentially added: (1) feasibility cuts based on enforcing $x_1 \in K$(here $K=\{x_1 \in \mathbf{R}^{m1} \mid (A_1x_1=b_1,\ x_1 \geq 0) \cap (\forall\ \omega \in \Omega,\ \exists x_2(\omega) \geq 0$ such that $A_2(\omega)x_2(\omega)=b_2(\omega)-B_1(\omega)x_1)\}$ and (2) optimality cuts generated by based on objective value bounding properties of the outer linearization of $Q(x_1)$.

Proceeding with the outer linearization procedure, we derive the following "full master problem"

$$\min\ c_1x_1+\boldsymbol{\theta}$$

$$\text{s.t.} \qquad (2.4.a)$$

$$\sum_{\omega=1}^{L} p_2(\omega)(b_2(\omega)-B_1(\omega)x_1)^T\boldsymbol{\pi}_i(\omega) \leq 0,\ i=1,\ \dots,\ m, \qquad (2.4.b)$$

$$(b_2(\omega)-B_1(\omega)x_1)^T\sigma_j(\omega) \leq 0,\ j=1,\ \dots,\ n(\omega), \qquad (2.4.c)$$

$$A_1x_1=b_1$$

$$x_1 \geq 0.$$

Constraints (2.4.b) are "optimality cuts," and constraints (2.4.c) are "feasibility cuts." See [24] for a detailed description to derive constraints (2.4.b) and (2.4.c) and dual prices, $\sigma_j(\omega)$ or $\pi_i(\omega)$. To derive an explicit set of optimality constraints, it is necessary to employ an approximation of $Q(x_1)$. The approach is based on the work of Lasdon [23]. While Lasdon's work is concerned with Benders' decomposition approach, this study is concerned with stochastic programming problems. We must deal with the same number of subproblems as the number of scenarios in the second stage, while Lasdon's account of Benders' decomposition approach deals with only one subproblem.

Now, we provide details of the L-shaped algorithm and the termination criteria. Master problem (2.4) is called a "full master problem" since both constraint sets involve a full contingent of constraints. The full set of constraints are not explicitly available and the number of all constraints is usually greater than the number of variables [4]. It becomes necessary to employ a relaxation strategy to address problem (2.4).

In the L-shaped algorithm, we first solve the first-stage original master problem, obtaining a primal solution x_1^0 to:

$$\begin{aligned} \min \quad & c_1x_1 \\ \text{s.t.} \quad & \\ & A_1x_1 = b_1,\ x_1 \geq 0. \end{aligned} \tag{2.5}$$

We modify the right-hand sides of L subproblems (2.3) with a primal solution x_1^0, then we attempt to solve the subproblem in $x_2(\omega)$ for all ω, obtaining dual prices, $\sigma_1(\omega)$ or $\pi_1(\omega)$. We generate cut(s) (either optimality or feasibility) with dual prices and append them to a relaxed master problem. Then we solve the augmented master problem, obtaining a "better" x_1^k, $k \geq 1$. We terminate this procedure in a finite number of iterations. The reason of finite iterations is as follows: Problem (2.4) has only a finite number of constraints. If the optimality test is not passed, then one or more new constraints are added. Thus, in a finite number of iterations either the optimality test is passed or a full set of constraints will be obtainted.

Details of the L-shaped algorithm are shown in the Nassi-Schneiderman chart, given in figure 1. The algorithm terminates under two conditions, when either the initial master or the augmented master problem is infeasible, or when the termination criterion is satisfied in a finite number of iterations. The termination criterion requires the lower bound θ_L^K to be within ε of the upper bound θ_U^k. While the lower bound increases monotonically, the upper bound value is not strictly monotonically decreasing. The lower bound θ_L^K is the optimal solution value when the relaxed master problem is solved at iteration k. The upper bound value is the minimum of the following two values, the currently computed sum of objective function value for the stage and all subproblems(i.e., $c_1 x_1^{k-1} + \sum_{\omega=1}^{L} p_2(\omega)\ z^k(\omega)$) and the previous least upper bound value θ_L^{k-1}

step 1: solve the original master problem initialize data		
while the terminatation criterion is not satisfied		
	step 2: for all subproblems	
		setup subproblem
		solve subproblem
	step 3: setup the augmented master problem solve the augmented master problem	

(a) main procedure

solve the original master problem (2.4)
set m = n = k = 0, and lower bound (θ_L^K) = $-\infty$ upper bound (θ_L^K) = ∞ let the optimal solution be x_1^k

(b) step 1: procedure

set k = k+1 modify right hand side with x_1^{k-1}
solve subproblem (2.3) let $z^k(\omega)$ be the optimal value of ω^{th} subproblem for $\omega = 1, \cdots, L$ store dual price

(c) step 2: procedure

If all subproblems are feasible		
	then generate optimality cut (2.4.b) set m = m+1, update upper bound, $\min\{\theta_U^{k-1},\ c_1 x_1^{k-1} + \sum_{\omega=1}^{L} p_2(\omega) z^k(\omega)\}$	else generate feasibility cut(s) (2.4.c) set n = n+1
append corresponding cut(s) to the master problem solve the augmented master problem (2.4)		
If the master problem is feasible		
	then let the optimal solution be (x_1^k, θ^k) update lower bound, $\theta_L^k = c_1 x_1^k + \theta^k$	else terminate

(d) step 3: procedure

Figure 1. Procedural Logic of the L-shaped Algorithm

2.3. Computational Example.

A numerical example is used here to elaborate various algorithms' solution procedures and to focus on specific features inherent in the L-shaped and the analytic center algorithm. The example is a slight modification of an example given in Birge and Louveaux [4]. The numerical example is as follows.:

$$\min\ cx+Q(x)$$
$$\text{s.t.} \qquad (2.6)$$
$$0 \le x \le 10$$

where $Q(x)=E_{\xi}\, q(x, \omega)$,

$q(x, \omega)=\{\xi - x, \text{ if } x \le \xi,\ x-\xi, \text{ if } x \ge \xi\}$, where

ξ can take on the values 1, 2, and 8 each with probability $^1/_3$,

and c is a null vector.

This problem has functions q(x, 1), q(x, 2), and q(x, 3) (i.e., $|x-1|$, $|x-2|$, and $|x-8|$ respectively) each of which is convex. Recall that the L-shaped algorithm generates a single cut from three functions, and then computes the current optimal solution and lower bound from a relaxed master problem. The current optimal solution is passed to the subproblems to generate a cut. This procedure is repeated until the algorithm converges. The L-shaped algorithm outerlinearizes Q(x) by identifying a single hyperplane at each iteration.

Iteration 1: Assume the starting point $x^0 = 0$ and lower bound $\theta_L^0 = -\infty$.

We have the upper bound $\theta_U^1 = 11/3$ and a new cut from the functions (q(x, 1), q(x, 2), and (q, 3)). The augmented master problem is

$$\min\ \theta$$
$$\text{s.t.} \quad 0 \le x \le 10. \qquad (2.7)$$

Solving problem (2.7) produces the lower bound $\theta_L^1 = -(19/3)$ and $x^1 = 10$, which is not optimal.

Iteration 2: We have the upper bound $\theta_U^2 = \min\{11/3,\ 19/3\} = 11/3$ and a new cut from the functions. Appending the cut $\theta \ge x - 11/3$ to master problem (2.7) and solving it produces the lower bound $\theta_L^2 = 0$ and $x^2 = 11/3$, which is not optimal.

Iteration 3: We have the upper bound $\theta_U^3 = \min\{11/3,\ 26/9\} = 26/9$ and a new cut from the functions. Appending the cut $\theta \ge (1/3)x + 5/3$ to master problem and solving it produces the lower bound $\theta_L^3 = 0$ and $x^3 = 1.5$, which is not optimal.

Iteration 4: We have the upper bound $\theta_U^4 = \min\{26/9, 2.5\} = 2.5$ and a new cut from the functions. Appending the cut $\theta \geq 3 - (1/3)x$ to master problem and solving it produces the lower bound $\theta_L^4 = 7/3$ and $x^4 = 2$, which is not optimal.

Iteration 5: We have the upper bound $\theta_U^5 = \min\{2.5, 7/3\} = 7/3$ from the functions. It is optimal because $\theta_L^4 = \theta_U^5$.

We observe that the master program is unstable as new cuts are appended. When a new cut is appended at iteration 1, the current optimal solution changes rapidly from the previous optimal solution. That is, from $x^0 = 0$ to $x^1 = 10$. As iterations proceed, the optimal solution is computed subject to ever more appended cuts and the instability of solutions diminishes. But the initial instability has been observed as a reason for slow convergence [30]. As expected, the lower bound is monotonically increasing, the current least upper bound is selected as the lesser of the previous least upper bound and the currently computed upper bound.

3. Analytic Center Algorithm

The L-shaped algorithm, which is a cutting plane algorithm, is based on the selection of a primal vector x that is sent to subproblems in order to modify their corresponding right-hand sides. A purely linear master problem like problem (2.4) selects a primal vector from the extreme points of the so-called "localization set" where the objective function value is minimized. The localization set is constructed from a relaxed master problem and a given upper bound on the overall problem (2.1). As previously noted, cutting plane algorithms have exhibited slow convergence.

The analytic center of a polyhedral set, as proposed initially in [14,15,32], is used to avoid the instabilities experienced by classic cutting plane methods. An analytic center is a center point of the localization set. These early studies have shown that convergence rates may be accelerated if cutting planes are generated from analytic centers of the encountered localization sets instead of generating them from extreme points. This analytic center concept was further elaborated by Goffin and colleagues [16,17] in proposing a new treatment of the master problem in the classical Dantzig-Wolfe approach. A method is to associate a projective algorithm [39] with a cutting plane algorithm. The projective algorithm is used to get a center point in the localization set. However, early studies [14,15,16] were theoretical in nature and did not report any numerical experimentation. Numerical experimentation in [1] was reported on a set of convex constrained geometric programming problems, provided by Kortanek and No [22], showing a good convergence rate. The study in [17] reports computational results showing that the analytic center method may perform better than the classical DW approach, although experimentations are based on small convex nondifferentiable test problems. We are aware of no published work relating the analytic center concept to solving stochastic programming problems. Hence, one objective of this study is to tailor the analytic center concept to solving stochastic LP problems and to report on initial computational experiments.

The master problem of an analytic center algorithm selects a primal vector which is the analytic center of the localization set. The localization set is constructed from a relaxed master problem and a given upper bound. A general way to obtain an analytic center is based on the work given by Goffin et al. [16], but it is modified here to apply to TSLP problems.

3.1. Analytic Center of the Localization Set

The Master problem of algorithm is the same as the master problem (2.4) of the L-shaped algorithm. A tractable approximation of the master problem must consider only subsets of the full set of constraints. Let $\bar{I}$ and $\bar{J}$ be subsets of optimality constraints and feasibility constraints, respectively. Then we have a relaxed master problem

$$\min\ c_1x_1 + \theta \tag{3.1.a}$$

s.t.

$$\Sigma_{\omega}\, p_2(\omega)(b_2(\omega)-B_1(\omega)x_1)^T \pi_i(\omega) \leq \theta\, 1,\ i\in \bar{I} \tag{3.1.b}$$

$$(b_2(\omega)-B_1(\omega)x_1)^T \ \sigma_j(\omega) \leq 0,\ j \in \bar{J} \tag{3.1.c}$$

$$A_1x_1 \geq b_1, \tag{3.1.d}$$

$$x_1 \geq 0. \tag{3.1.e}$$

Here, c_1 is a vector in $\mathbb{R}^{n(1)}$, b_1 is a vector in $\mathbb{R}^{m(1)}$, A_1 is a matrix of real numbers and size $m_{(1)}$ x $n_{(1)}$. Constraints (3.1.d) are actually intended to represent both equality and inequality constraints. We consider an inequlity constraints separately in (3.1.d) for special treatment in the analytic center problem. Any equality constraints are considered explicitly to be presented potential function. The true inequality constraints may accommodate upper and lower bounds in each decision variable in x_1. The localization set from the relaxed master problem (3.1) and a given upper bound θ_U on the overall problem (2.1) is defined to be:

$$\overline{F}(\theta_U)=\{(x_1,\theta)\ ;\ c_1 x_1+\theta \le \theta_U,$$

$$\theta 1 \ge \Sigma_\omega\, p_2(\omega)(b_2(\omega)-B_1(\omega)x_1)^T \pi_i(\omega), i\in \overline{I} \quad (3.2)$$

$$A_1 x_1 \ge b_1,\ x_1 \ge 0,\ x_1 \in \mathbb{R}^{n(1)}\}$$

where 1 is the vector of ones.

This polyhedron has dimensiion $\mathbb{R}^{n(1)+1}$ and contains the optimal solution of the current master problem. Two assumptions assure the existence of the analytic center of (3.2). The first assumption is that the subset $\overline{I}$ is not empty. This assumption with a given upper bound and constraints (3.1.d) and (3.1.e) implies that the localization set is bounded. The other assumption is that the interior of the polyhedron is not empty.

Now, let us consider a way to obtain an analytic center of the localization set (3.2). The analytic center is based on the maximization of a so-called potential function. To simply express the potential function, let us introduce the follwing slack variables:

$$\begin{aligned}
S_0 &= \theta_U - c_1 x_1 - \theta \\
S_i &= \theta - \Sigma_\omega\, p_2(\omega)(b_2(\omega)-B_1(\omega)x_1)^T \pi_i(\omega), i\in \overline{I} \\
S_j &= -(b_2(\omega)-B_1(\omega)x_1)^T \sigma_j(\omega), j\in \overline{J} \\
S_g &= A_1^g X_1^g - b_1,\ g\in \underline{G}.
\end{aligned} \quad (3.3)$$

$\underline{G}$ is a subset(i.e., inequality constraints) of the first stage constraints. Then we have the following potential function:

$$\overline{\phi}(\theta_U)=\ln s_0+\Sigma_{i\in\overline{I}}\ln s_j+\Sigma_{j\in\overline{J}}\ln s_j+\Sigma_{g\in\underline{G}}\ln s_g \quad (3.4)$$

s.t. any ture equality constraints in (3.1.d).

This function is well defined on the interior of the localization set, since all the slacks are then strictly positive and the logarithms are defined. The potential function is strongly concave on this domain [17] and it tends to $-\infty$ towards the boundary. Thus it achieves a global maximum on its domain and its maximizer is unique due to a strong concavity property. The anaytic center of the localization set is defined to be precisely this maximizer. Using (3.3), we eliminate the slack variables in the definitions of $\overline{\phi}(\theta_U)$.

Maximizing of the potential function, $\overline{\phi}(\theta_U)$, we obtain the analytic center value $(\hat{x}_1, \theta_c)$. However, if the value θ_c is not optimal in the current master problem, we can recover an optimal value θ, denoted θ_m. The value θ_m represents a minimum value of the relaxed master problem (3.1) associated with the corresponding vector $\hat{x}_1$. The value θ_m is recovered as:

$$\theta_m = \max_{i \in \bar{I}} \Sigma_\omega p_2(\omega)(b_2(\omega) - B_1(\omega)\hat{x}_1)^T \pi_i(\omega). \tag{3.5}$$

At iteration k, θ^k_U(the upper bound on the overall problem), θ^k_c, and θ^k_m may be different. The value θ^k_m must be less than or equal to the value θ^k_c at iteration k.

3.2. Convergence of Analytic Center Algorithm

The procedural logic of analytic center algorithm is depicted in figure 2.

step 1: initialize data		
while the termination criterion is not satisfied		
	step 2: for all subproblems.	
		setup subproblem
		solve subproblem
	setp 3: generate corresponding cut(s) construct augmented master problem maximize the potential function	

(a) main procedure

compute the center of an initial feasible set
set m = n = 0, ε , k = 0, $\theta^K_U = \infty$, and $\theta^K_L = -\infty$ let the optimal solution be x^k_1

(b) step 1: procedure

set k = k + 1 and modify right-hand side with X_1^{K-1}
solve subproblem (2.3) let $z^k(\omega)$ be the optimal value of ω^{th} subproblem store dual price

(c) setp 2: procedure

<table>
<tr><td colspan="4">If any of subproblems is infeasible</td></tr>
<tr><td rowspan="3"></td><td rowspan="3">then
generate feasibility cut(s) (3.1.c)
set n = n +1
append cuts to master problem</td><td colspan="2">else
generate optimality cut (3.1.b)
update upper bound $\theta_U^k =$
$\min\{\theta_U^{k-1},\ c_1 x_1^{k-1} + \sum_{\omega=1}^{L} p_2(\omega) z^k(\omega)\}$</td></tr>
<tr><td colspan="2">If the cut is not the same as previous ones</td></tr>
<tr><td>then
set m = m + 1
append cut</td><td>else</td></tr>
<tr><td colspan="4">construct the augmented master problem (3.1)
construct the localization set (3.2) and compute the analytic center of (3.2)</td></tr>
<tr><td colspan="4">If the potential function is feasible</td></tr>
<tr><td></td><td>then
let the analytic center be $(\hat{x}_1, \theta_c)$
update lower bound
$\max\{\theta_L^{k-1}$, from (3.10)}</td><td colspan="2">else</td></tr>
</table>

(d) step 3: procedure

Figure 2. Procedural Logic of the Analytic Center Algorithm

A biref description of algorithm will be given below. In the beginning, it is necessary to comptute the analytic center of the initial feasible set: $\{x_1 : A_1 x_1 \geq b_1,\ x_1 \geq 0\}$. The step 2 procedure is to pass down a primal solution x_1 to subproblems, to modify their right-hand side value, and to solve them. The step 3 procedure has three cases. The first case occurs when any subproblem is infeasible. Then generate corresponding feasibility cuts, append them to the relaxed master problem, and solve its corresponding potential function. The second case occurs when all subproblems are feasible and the vector of dual prices for the subproblems, produced in the current iteration, is not the same as the previous vector. By aggregating dual prices from all subproblems, we produce one single optimality cut. Then append it to the relaxed master problem, and solve the corresponding potential function. The third case occurs when either one of the two conditions in the second case is not satisfied. Then only update θ_U(the upper bound on the overall problem), and carry out the maximization of the corresponding potential function.

Algorithm terminates when $\theta_L^K + \varepsilon \geq \theta_U^k$. While we obtain the value of θ_U form the solution of the subproblems, we obtain the value of θ_L from the computation of the analytic center, as will now be described. While linear cutting palne algorithms such as the L-shaped algorithm obtain the lower bound value directly from the computation of the relaxed master problem, we obtain the lower bound value from the analytic center.

Before getting the lower bound, we will use simplified notation to aid the exposition. Consider the following simplified relaxed master problem instead of using (3.2) and its dual.

$$\min \quad \{c^T x : A^T x - s = b,\ s \geq 0\} \text{ where } A \in \mathbb{R}^{m \times n}. \tag{3.6}$$

$$\max \quad \{b^T u : Au = c,\ u \geq 0\}. \tag{3.7}$$

With the same assumptions and procedures previously mentioned, we can derive the localization set:

$$\overline{F(\theta_U)} = \{x : c^T x \leq \theta_U ,\ A^T x \geq b\}. \tag{3.8}$$

We define the exact analytic center as the point in the localization set which maximizes the following potential function:

$$\begin{aligned} \max \quad & \ln S_0 + \textstyle\sum_{i=1}^{n} \ln S_i \\ \text{s.t.} \quad & \\ & S_0 = \theta u - c^T x,\ S_0 > 0 \\ & S_i = A^T x - b,\ S_i > 0 \end{aligned} \tag{3.9}$$

Now, We can get the lower bound using the notation in (3.6)-(3.9), due to the theorem in [16]. Let $\hat{x}$ be the analytic center of the localization set $F(\theta_U) = \{x : c^T x \leq \theta_U,\ A^T x \geq b\}$. Then, (3.1) is a lower bound on the optimal value of problem (3.6).

$$\theta_L = (n+1)c^T \hat{x} - n\theta_U \tag{3.10}$$

Now consider the convergence of algorithm. If the termination test is not passed, one or more constraints are added to a relaxed master program. In a finite number of iterations, exactly one of three possible conditions ensues: (1) the optimality test is passed; or (2) the optimality test is not passed and cuts are still produced; or (3) the optimality test is not passed and no more cuts are produced. When the optimality test is not passed after a finite number of iterations, algorithm does not produce condition (2) any more [24].

Thus, the algorithm must be in condition (3) after a finite number of iterations. Then algorithm terminates because of the following property: The volume of the localization set is shrinking as iterations proceed. With a small enough volume of the localization set, algorithm terminates because $\theta_U \leq \theta_L - \varepsilon$.

3.3. Computational Example

Here, we consider the same numerical example as used previously.

Iteration 1: Compute the center point $x_1 = (0+10)/2$ from the box constraint, set $\epsilon = 0.005$, $\theta_U = \infty$, and $\theta_L = -\infty$. We have the upper bound $\theta_U^1 = 10/3$ and a new cut from the functions.

(Master) Append the cut $\theta \geq (1/3)x + 5/3$ to the current master problem and update the upper bound. Construct the localization set

$$\overline{F}(\theta_U) = \{(x, \theta) : \theta \leq \theta_U^1, \theta \geq (1/3)x + 5/3, 0 \leq x \leq 10\}.$$

To obtain the analytic center, maximize the corresponding potential function

$$\overline{\phi}(\theta_U) = \ln(10/3 - \theta) + \ln(\theta - (1/3)x - 5/3) + \ln(10 - x) + \ln x$$

to produce $(x^1, \theta_c^1) = (1.454, 2.744)$. Update $\theta_L^1 = 0.976$ and proceed.

Iteration 2: We have the upper bound $\theta_U^2 = (7.536/3)$ and a new cut from the functions. Append the cut $\theta \geq -(1/3)x + 3$ and update the upper bound.

Maximize the corresponding potential function

$\overline{\phi}(\theta_U) = \ln(7.536/3 - \theta) + \ln(\theta - (1/3)x - 5/3) + \ln(\theta + (1/3)x - 3) + \ln(10 - x) + \ln x$ to produce $(x^2, \theta_c^2) = (2.024, 2.453)$. Update $\theta_L^2 = 2.217$ and proceed.

Iteration 3: We have the upper bound $\theta_U^3 = (7.024/3)$ which produces no new cut from the functions. Update only the upper bound on the overall problem.

Maximize the corresponding potential function

$\overline{\phi}(\theta_U) = \ln(7.024/3 - \theta) + \ln(\theta - (1/3)x - 5/3) + \ln(\theta + (1/3)x - 3) + \ln(10 - x)$

to produce $(x^3, \theta_c^3) = (2.0, 2.339)$. Update $\theta_L^3 = 2.330$ and proceed.

Iteration 4: We have the upper bound $\theta_U^4 = 7/3$ from the functions. The value x^3 is optimal because $\theta_L^3 = \theta_U^4$. The value θ_m^4 is 7/3. The algorithm terminates.

We observe that the algorithm terminates at iteration four. The first two iterations generate optimality cuts and update the upper bound, while the remaining iterations only update the upper bound and shrink the volume of the localization set. We get the first analytic center from the localization set that is the bounded set from the first cut, the box constraint $(10 \geq x \geq 0)$, and upper bound 1. At iteration two, we have a reduced localization set that is the bounded set from the first cut, the second cut, and upper bound 2. At iteration 3, the localization set is reduced again because of the updated least upper bound 3, although there is no new optimality cut.

4. Variations to the L-shaped and the Analytic Center Algorithm

A further objective of this study is to incorporate an idea of [13] into both the L-shaped algorithm and the analytic center algorithm. Their idea is not to optimize the current master problem but to find any feasible point to the master problem that provides an improved bound on the overall objective value.

We, first, develop a variation of the L-shaped algorithm through incorporation of an idea of Geoffrion and Graves. The main difference between the L-shaped algorithm and the variation to be presented is in formulating the current master problem and the termination criterion. While the L-shaped algorithm seeks to optimize the current master problem, the variation seeks to find only a feasible solution of the current master problem with an improved upper bound. Section 4.1 contains the derivation of the master problem and defines the variation algorithm derived form the L-shaped algorithm.

The analytic center method uses a nonlinear masterm problem to get analytic center. When a center point is passed down to the subproblems, it is not necessarily a good point to produce a deep cut. As we observed in the computational example in section 2.3, an extreme point is not necessarily a good point, either. Some point in the localization set may be a better point for generating a cut. As it takes time to compute an exact analytic center of the localization set, the prospect emerges that a suboptimization of the potential function may accelerate empirical convergence of the analytic center algorithm.

4.1. The Variation to the L-shaped Algorithm

The differences between the L-shaped and the variation are as follows. While the L-shaped algorithm optimizes the current master problem to find an optimal solution, the variation seeks a feasible solution of the current master problem having a value less than UB- ε. Therefore, the current master problem cannot produce a lower bound any more. We have the following full master problem

$$\min \ \phi(x_1, \theta) \quad (4.1.a)$$

s.t.

$$c_1 x_1 + \theta \leq \theta_U - \varepsilon \quad (4.1.b)$$

$$\theta 1 \geq \Sigma_{\omega=1}^{L} p_2(\omega)\ (b_2(\omega) - B_1(\omega)x_1)^T \pi_i(\omega), \ \ i = 1, \cdots, \ m, \quad (4.1.c)$$

$$0 \geq (b_2(\omega) - B_1(\omega)x_1)^T \sigma_j(\omega), j = 1, \ldots, n(\omega), \quad (4.1.d)$$

$$A_1 x_1 \geq b_1, \quad (4.1.e)$$

$$x_1 \geq 0. \quad (4.1.f)$$

From master problem (2.4), the objective function value of the master problem is bounded by UB- ε , shown in constraint (4.1.b) here. Eliminating the variable θ, we have the following equivalent master problem

$$\min \ \phi(x_1) \quad (4.2.a)$$

s.t.

$$c_1 x_1 + \Sigma_{\omega=1}^{L} p_2(\omega)(b_2(\omega)x_1) - B_1(\omega)x_1)^T \pi_i(\omega) \leq \theta_U - \varepsilon, i = 1, \cdots, m, \quad (4.2.b)$$

$$0 \geq (b_2(\omega) - B_1(\omega)x_1)^T \sigma_j(\omega), \ j = 1, \cdots, n(\omega), \quad (4.2.c)$$

$$A_1 x_1 \geq b_1, \quad (4.2.d)$$

$$x_1 \geq 0. \quad (4.2.e)$$

There is a question about choosing an appropriate objective function, $\phi(x_1)$ as the problem becomes feasibility seeking only. We can use any convenient objective function in x_1 to produce a feasible solution. The x_1 function on the left-hand side of the latest optimality cut (4.2.b) is suggested for use in [13]. Then the objective function of the master problem is

$$\min \quad c_1 x_1 + \sum_{\omega=1}^{L} p_2(\omega)(b_2(\omega) - B_1(\omega) x_1)^T \pi_{\bar{i}}(\omega) \tag{4.3}$$

where $\bar{i}$ denotes the latest optimality cut.

As a lower bound is not produced from the master problem, it is necessary to modify the L-shaped algorithm. Instead of solving master problem (2.4), we solve master problem (4.2). The variation algorithm terminates when the master problem (4.2) has no feasible solution, meaning there is no solution to the original problem better than the incumbent by at least ϵ in objective value. In particular, the current best solution is ϵ -optimal, i.e., having an objective value within ϵ of the optimal value.

4.2. Relaxation Strategy for the Analytic Center Algorithm

The main idea of this implementation is to find an approximate center point, rather than spending the computation time to find an exact analytic center. As perviously pointed out, an approximate center point of the localization set may well produce a deeper cut than the exact analytic center. Instead of optimizing the potential function (3.4) with a tight stopping rule tolerance, we find an approximate center point from the localization set by relaxing the optimality tolerance level. Then when the optential function optimization process terminates, an approximate center point is obtained.

The issue of how much we can relax the optimality tolerance level does not seem to have been considered in the literature. The computational experiments reported below are intended to provide some empirical guidance.

5. Computational Experiments

5.1. Specification of the Test Problems

The algorithms were run on a set of seven TSLP problems. Table 1 shows the characteristics of the test problems.

Table 1. Characteristics of the Test Problems

problem Name	# of scenarios	Size of subproblems	Equivalent LP	Objective Value
PR1	1-2	2×2, 5×6	12×14	789.8
LP1	1-3	5×5, 11×3	38×12	37231633
SC205	1-8	12×14, 22×22	189×190	-60.4
SCRS8	1-8	28×37, 28×38	252×341	123.4
SCAGR7	1-8	15×20, 38×40	319×340	-832903.5
SCTAP1	1-8	30×48, 60×96	510×816	360.5[1]
PR2	1-8	11×10, 99×90	803×730	8275.456

1. This value, computed using GAMS, is different form that reported in [3,12].

Test problems SC205, SCRS8, SCAGR7, and SCTAP1, supplied by John Birge, are stochastic versions of problems from Ho and Loute [18]. They are deterministic multistage problems with staircase structure. These test problems are medium-sized problems, each based on a practical application (see the details [3, 18]). But they have become standardized problems since they were used in [3, 12]. Test problem LP1, taken form [7], is a small-sized two-period problem with block-angular structure. Test problems PR1 and PR2, used in [24], were formulated as investment problems with additional policy constraints and were small and medium-sized problems, respectively.

All of the full deterministic equivalent problems were solved using GAMS version 2.25x(General Algebraic Modeling System), to provide a check on the solution value and the optimal solutions. We implemented the various algorithms in FORTRAN 77 on a DEC station 5000 under the Ultrix operating system in order to test effectiveness for solving test problems. All algorithms were written by us using MINOS 5.3 to solve the various linear and nonlinear programming problems encountered. Since MINOS requires the reading of SPECS files and MPS files as input, some customizing was necessitated in using the MINOS code(see [24]).

5.2. Computational Results

With the above implementation, CPU times and the number of the master problems to be solved were used as measures of performance. In addition to the number of the master problems, we report here the number of feasibility cuts and optimality cuts to provide a general idea of the behavior of the algorithms as they were applied to the test problems. Efficiency measures such as using the last optimal basis to begin further (dual) simplex iterations upon inclusion of new constraints in the

L-shaped algorithm, or using the previous Hessian matrix when adding a term to the potential function in the analytic center method were not used. Hence numbers of the master problems solved and numbers of generated cuts are perhaps more meaningful measures of relative performance. All times reported here include the generation of input files (MPS files) and the solution of the master problems and subproblems.

Table 2 shows the computational results for each test problem. The performance of the analytic center algorithm was presented for three optimality tolerance levels: the value 1E-6 is a default value; the next column shows the best performing tolerance value; and the value in the last colunm results in the algorithm not converging. The last case happens when the optimality tolerance is set too loosely.

Based on these results, we make several observations. First, when the L-shaped algorithm is applied, most test problems are solved within a few iterations (at most 6 iterations for problem SCAGR7). Thus, it is quite difficult for the other proposed algorithms to materially reduce the number of the master problems to be solved. A strategy to eliminate inactive constraints, proposed for the analytic center algorithm in [17], might be inefficient to implement here since most test problems produce only a few cuts. Analytic and the variation to the L-shaped algorithm perform better than the L-shaped algorithm in test problems PR1 and PR2. The test problems are solved within two iterations. These two algorithms generate a feasible solution to the subproblem when they solve the initial master problem on first iteration. Thus, they eliminate the step of feasibility cut generation, while the L-shaped algorithm require that step.

Second, when the analytic center algorithm is applied to test problems, LP1, SCAGR7, and SCTAP1, the numbers of master problems to be solved increase considerably compared to the L-shaped and the variation to the L-shaped. For example, the numbers increased from 3 to 6 on problem LP1, form 6 to 13 on problem SCAGR7, and from 2 to 6 on problem SCTAP1. None of these test problems produces any new cuts after the first few iterations. They only update their upper bounds, as shown in example 3.3. It takes a few more iterations to compute the optimal solutions with only the upper bounds updated. In these instances, when we introduce a new implementation, the number of master problems to be solved can be considerably reduced. A hybrid implementation suggests itself: when no new cuts are produced by the analytic center algorithm, the L-shaped algorithm or the variation to the L-shaped is then applied to the current master problem. With this new implementation, the numbers of master problems to be solved decreased form 5 to 4 on problem LP1, from 13 to 9 on problem SCAGR7, and from 6 to 4 on problem SCTAP1. Thus, this implementatioin can be considerably more effective when no new cuts are produced by the analytic center algorithm.

Third, the other performance criterion, CPU time, portrays a complementary comparison of the algorithms. The variation to the L-shaped algorithms performs best among all algorithms with respect to CPU time, except in the case of the analytic algorithm with optimality tolerance value, 0.1 and 0.01, on test problems, SCRS8 and PR2 respectively. The variation to the L-shaped algorithm reduces CPU time 2% to 53% compared to the L-shaped algorithm.

Table 2. Computational Results

PR1	L-shape	variation	ACA(1E-6)	ACA(1E-1)	ACA(1E-0)
#master	3(1/1)[1]	2(0/1)	2(0/1)	2(0/1)	*[2]
time(sec)	0.55	0.36	0.48	0.42	*
LP1	L-shape	variation	ACA(!E-6)	ACA(1E-5)	ACA(1E-4)
#master	3(1/1)	3(1/1)	5(1/1)	5(1/1)	*
time(sec)	0.87	0.85	1.87	1.79	*
SC205	L-shape	variation	ACA(1E-6)	ACA(1E-1)	ACA(1E-0)
#master	4(3/0)	4(3/0)	5(3/1)	5(3/1)	*
time(sec)	5.45	5.43	7.54	7.35	*
SCRS8	L-shape	variation	ACA(1E-6)	ACA(1E-1)	ACA(1E-0)
#master	2(0/1)	2(0/1)	2(0/1)	2(0/1)	*
time(sec)	4.28	4.03	4.08	3.95	*
SCAGR7	L-shape	variation	ACA(1E-6)	ACA(1E-3)	ACA(1E-2)
#master	6(3/2)	6(3/2)	13(5/2)	13(5/2)	*
time(sec)	17.74	17.33	46.67	41.32	*
SCTAP1	L-shape	variation	ACA(1E-6)	ACA(1E-3)	ACA(1E-0)
#master	2(0/1)	2(0/1)	6(0/2)	6(0/2)	*
time(sec)	10.26	10.20	51.93	47.84	*
PR2	L-shape	variation	ACA(1E-6)	ACA(1E-2)	ACA(1E-1)
#master	3(1/1)	2(0/1)	2(0/1)	2(0/1)	*
time(sec)	22.83	15.47	16.49	15.40	*

1. The first and second numbers in parenthesis represents the number of feasibility and optimality cuts, respectively.
2. * indicates no convergence

Finally, the implementation of relaxing optimality tolerance values to the analytic center algorithm seems to be a promising strategy for the analytic center algorithm. The analytic algorithm with a relaxed parameter reduces CPU time 2% to 12% compared to the analytic algorithm with default value 1E-6(PR1 13%, LP1 2%, SC205 3%, SCRS8 3%, SCAGR 12%, SCTAP 8%, PR2 7%). Unfortunately, it appears to be difficult to determine, *a priori*, the best optimality tolerance value for all test problems since this will be problem specific. Up to some optimality tolerance level, we have observed reduced CPU times. However, the algorithm does not converge when the optimality tolerance is set too loosely.

6. Conclusions

In this paper, we have proposed new solution methods to solve TSLP problems. One solution method is to combine the analytic center concept with Benders' decomposition strategy to solve two-stage stochastic LP problems. The analytic center concept was proposed to solve convex nondifferentiable problems. This approach was tested on convex nondifferentiable problems and produced quite promising results for those problems. From the computational results for our test problems, the performance of the analytic center algorithm was not always good. The reasons for this one may be explained by two things. First, since the L-shaped algorithm performs well, it is quite hard for the analytic center algorithm to improve the number of the master problems to be solved for a small or medium-sized problems. Second, since convex nondifferentiable test problems originally applied by Bahn et al. have large volumes in the localization set (see the details of test problems [1,22]), they were performed very well with the analytic center concept. But, since our test problems have relatively small volumes in the localization set(i.e., most of our test problems have lots of equality constraints), the opportunities to generate a deep cut may be limited.

The other method is to apply an idea applied by Geoffrion and Graves to modify the L-shaped algorithm and the analytic center algorithm. The variation algorithm seeks to find a feasible solution of the current master problem's constraint set rather than to find the optimal solution since the current master problem has too little information about the full problem in early iterations. In the analytic center algorithm, we seek to find an approximate center point of the localization set rather than to find an exact center point. From the computational results, the idea to modify the L-shaped algorithm and the analytic center algorithm was found to be quite promising. The variation to the L-shaped algorithm performs better than or at least equal to the other algorithms with respect to the number of the master problems to be solved and the CPU times. The relxation strategy applied to the analytic center algorithm improved the CPU times compared to the analytic center algorithm, while it does not improve the number of master problems to be solved.

A further idea is to combine the analytic center algorithm with the L-shaped or the variation to the L-shaped algorithm. Since the analytic center algorithm can considerably reduce the number of master problems to be solved compared to the other algorithms, we can introduce a hybrid implementation. That is, when no new cuts are generated by the analytic center algorithm, the L-shaped or the variation to the L-shaped is then applied to the current master problem.

Since the computational experiments have been done on small-sized to medium-sized test problems and on test problem that have mostly equality constraints, our ability to extrapolate our conclusion is limited. If the various algorithms are tested on large-sized or other test problems, it is not guaranteed that we would get the same computational results as those of this study. Thus, a clear future research objective will be to perform computational experiments on other large-sized test problems.

References

[1] Bahn, O., J.L. Goffin, J.P. Vial, and O. Du Merle, "Implementation and behavior of an interior point cutting plane algorithm for convex programming: An application to geometric programming," ***Manuscript***, Département d'économie commerciale et industrielle, Université de Genève, Switzerland, 1991, to appear in ***Discrete Applied Mathematics***.

[2] Beale, E.M.L., "On minimizing a convex function subject to linear inequalities," ***Journal of the Royal Statistics Society***, B17(1955), 173-184.

[3] Birge, J.R., "Decomposition and partitioning methods for multistage stochastic linear programs," ***Operations Research***, 33(1985), 989-1007.

[4] Birge, J.R. and F.V. Louveaux, "A multicut algorithm for two-stage linear programs," ***European Journal of Operations Research***, 34(1988), 384-392.

[5] Birge, J.R. and R.J-B. Wets, "Designing approximation schemes for stochastic optimization problems, in particular for stochastic programs with recourse," ***Mathematical Progrmming Study***, 27(1986), 54-102.

[6] Chun, B.J., "Scenario analysis modeling and decomposition methods," ***PH.D. thesis***, Department of Industrial Engineering, Unversity of Wisconsin-Madison, 1992.

[7] Chun, B.J., S.J. Lee, and S.M. Robinson, "An Implementation of the bundle decomposition algorithm," ***Technical Report 91-6***, Department of Industrial Engineering, University of Wisconsin-Madison, 1991.

[8] Dantzig, G.B., "Linear programming under uncertainty," ***Management Science***, 1(1955), 197-206.

[9] Dantzig, G.B., and P.W. Glynn, "Parallel processors for planning under uncertainty," ***Technical Report SOL 88-8***, Department of Operations Research, Stanford University, California, 1988.

[10] Dantzig, G.B., and G. Infanger, "Multi-stage stochastic linear programs for portfolio optimizatio n," ***Technical Report SOL 91-11***, Systems Optimization Laboratory, Stanford University, California, 1991.

[11] Ermoliev, Y., "Facility location problem," ***Numerical Techniques for Stochastic Optimization***, Edited by Y. Ermoliev and R.J-B. Wets, Springer-Verlag, Berlin, 1988, 413-434.

[12] Gassmann, H.I., "MSLiP: a computer code for the multistage linear programming problem,"

Mathematical Programming, 47(1990), 407-423.

[13] Geoffrion, A.M., and G.W. Graves, "Multicommodity distribution system design by Benders decomposition," ***Management Science,*** 20(1974), 822-843.

[14] Goffin, J.L., "Affine and projective transformations in nondifferentiable optimization," ***Trends in Mathematical Optimization,*** Edited by K.H. Hoffmann, J.B. Hiriart-Urruty, C. Lemarechal, and J. Zowe, Birkhäuser Verlag, Basel, Switzerland, 1988, 79-81.

[15] Goffin, J.L., and J.P. Vial, "Cutting planes and column generation techniques with the projective algorithm," ***Journal of Optimization Theory and Applications,*** 65(1989), 409-429.

[16] Goffin, J.L., A. Haurie, J.P. Vial, and D.L. Zhu, "Using central prices in the decomposition of linear programs," ***Manuscript,*** Départment d'économie commerciale et industrielle, Université de Genève, Switzerland, 1991.

[17] Goffin, J.L., A. Haurie, and J.P. Vial, "Decomposition and Nondifferentiable optimization with the projective algorithm," ***Management Science,*** 38(1992), 284-302.

[18] Ho, J.K., and E. Loute, "A set of staircase linear programming test problems," ***Mathematical Programming,*** 20(1981), 245-250.

[19] Infanger, G., "Monte Carlo(importance) sampling within a Benders decomposition algorithm for stochastic linear programs," ***Technical Report SOL 89-13,*** Systems Optimization Laboratory, Stanford University, California, 1989.

[20] Kelley, J.E., "The cutting-plane method for solving convex programs," ***Journal of the SIAM,*** 8(1960), 703-712.

[21] King, A.J., "Stochastic programming problems: Examples from the literature," ***Numerical Techniques for Stochastic Optimization,*** Edited by Y. Ermoliev and R. J-B, Wets, Springer-Verlag, Berlin, 1988, 543-567.

[22] Kortanek, K.O., and H. No, "A second order affine scaling algorithm for the geometric programming dual with logarithmic barrier," ***Optimization,*** 23(1992), 303-322.

[23] Lasdon, L.S., ***Optimization Theory For Large Systems,*** The Macmillan Company, 1970.

[24] Lee, S.J., "Parallel solution of multistage stochastic linear programming problems using the analytic center method," ***Ph.D. Thesis,*** School of Business, University of Wisconsin-Madison, 1993.

[25] Mulvey, J.M., and H. Vladimirou, "Stochastic network optimization models for investment planning," ***Annals of Operations Research,*** 20(1989), 187-217.

[26] Mulvey, J.M., and H. Vladimirou, "Applying the progressive hedging algorithm to stochastic generalized networks," ***Annals of Operations Research,*** 31(1991), 399-424.

[27] Nielson, S.S., and S.A. Zenios, "A massively parallel algorithm for nonlinear stochastic network problems," ***Report 90-09-08,*** The Wharton School, University of Pennsylvania, Philadelphia,

Pennsylvania, 1990.

[28] Rockafellar, R.T., and R.J-B. Wets, "Scenarios and policy aggregation in optimization under uncertainty," ***Mathematics of Operations Research***, 16(1991), 119-147.

[29] Ruszczyński, A., "A regularized decomposition method for minimizing a sum of polyhedral functions," ***Mathematical Programming***, 35(1986), 309-333.

[30] Ruszczyński, A., "Modern techniques for linear dynamic and stochastic programs," ***Aspiration Based Decision Support Systems, Lecture Notes in Economics and Mathematical Systems*** 331, Edited by A. Lewandowski and A. Wierzbicki, Springer-Verlag, Berlin, 1989, 48-67.

[31] Ruszczyński, A., "Regularized decomposition and augmented Lagrangian decomposition for angular linear programming problems," ***Aspiration Based Decision Support Systems, Lecture Notes in Ecnomics and Mathematical Systems*** 331, Edited by A. Lewandowski and A. Wierzbicki, Springer-Verlag, Berlin, 1989, 80-91.

[32] Sonnevend, G., "New algorithms in convex programming based on a notion of center for systems of analytic inequalities and on rational extrapolation," ***Trends in Mathematical Optimization***, Edited by K.H. Hoffmann, J.B. Hiriart-Urruty, C. Lemarechal, and J. Zowe, Birkhauser Verlag, Basel, Switzerland, 1988, 311-326.

[33] Topkis, J.M., "A cutting plane algorithm with linear and geometric rates of convergence," ***Journal of Optimization Theory and Applications***, 36(1982), 1-22.

[34] Van Slyke, R., and R.J-B. Wets, "L-shaped linear programs with application to optimal control and stochastic optimization," ***SIAM Journal on Applied Mathematics***, 17(1969), 638-663.

[35] Walkup, D.W., and R.J-B. Wets, "Stochastic programs with recourse," ***SIAM Journal on Applied Mathematics***, 15(1967), 1299-1314.

[36] Wets, R.J-B., "Programming under uncertainty: The equivalent convex program," ***SIAM Journal on Applied Mathematics***, 14(1966), 89-105.

[37] Wets, R.J-B., "Programming under uncertainty: The solution set," ***SIAM Journal on Applied Mathematics***, 14(1966), 1143-1151.

[38] Wets, R.J-B., "Stochastic programming: solution techniques and approximation schemes," ***Mathematical Programming: The state of the art***, Edited by A. Bachem, M. Grötschel and B. Korte, Springer-Verlag, Berlin, 1983.

[39] Ye, Y., and M. Kojima, "Recovering optimal dual solutions in Karmarkar's polynomial algorithm for linear programming," ***Mathematical Programming***, 39(1987), 305-317.

[40] Zenios, S.A., "Massively parallel computations for financial planning under uncertainty," ***Report 90-11-10***, The Wharton School, University of Pennsylvania, Philadelphia, Pennsylvania, 1990.