Saeed Ketabchi; Malihe Behboodi-Kahoo

Augmented Lagrangian method for recourse problem of two-stage stochastic linear programming

*Kybernetika*, Vol. 49 (2013), No. 1, 188--198

Persistent URL: http://dml.cz/dmlcz/143249

KYBERNETIKA — VOLUME *49* (2013), NUMBER 1, PAGES 188–198

# AUGMENTED LAGRANGIAN METHOD FOR RECOURSE PROBLEM OF TWO-STAGE STOCHASTIC LINEAR PROGRAMMING

Saeed Ketabchi and Malihe Behboodi-Kahoo

In this paper, the augmented Lagrangian method is investigated for solving recourse problems and obtaining their normal solution in solving two-stage stochastic linear programming problems. The objective function of stochastic linear programming problem is piecewise linear and non-differentiable. Therefore, to use a smooth optimization methods, the objective function is approximated by a differentiable and piecewise quadratic function. Using quadratic approximation, it is required to obtain the least 2-norm solution for many linear programming problems in each iteration. To obtain the least 2-norm solution for inner problems based on the augmented Lagrangian method, the generalized Newton method is applied.

*Keywords:* two-stage stochastic linear programming, recourse problem, normal solution, augmented Lagrangian method

*Classification:* 90C15, 90C05, 90C20

## 1. INTRODUCTION

In mathematical linear programming, the elements of the vectors and matrices are assumed to have exact values. However, in practical problems, the data are not definite because of several uncertainties such as measurement errors, incomplete information about the future and events that have not yet occurred. In stochastic programming, some data are random variables with a specific probability distribution. This concept was first introduced in [12] by Georg Dantzig, the designer of linear programming. After that, many scholars such as Van Slyke and Wets [28], Higle and Sen [16], Infanger [18], Dantzig and Wolfe and Glynn [13, 14], Prekopa [25] and Branda [5] presented different methods for modelling and solving problems with uncertainties. A thorough introduction to this type of problem was presented in [4, 20]. Recently, a method based on the approximation of inner problems by quadratic problems, has been proposed by Chen et al. [3, 6, 7, 8, 9, 10, 11]. These authors present all needed solution methods as well as their parallelization.

Consider the following stochastic linear programming problem

$$\min_{x\in X} \;\; c^T x + E_\omega(Q(x,\omega)), \quad X = \{x \in \mathbb{R}^n : Ax = b, x \geq 0\}, \tag{1}$$

where $E$ is the expectation of function $Q(x, \omega)$ depending on the random variable $\omega$, and the function $Q$ is defined as follows:

$$Q(x, \omega) = \min_{y \in \mathbb{R}^{n_2}} \{q^T(\omega)y \mid W^T(\omega)y \geq h(\omega) - T(\omega)x\}, \tag{2}$$

where $A \in \mathbb{R}^{m \times n}, c \in \mathbb{R}^n$ and $b \in \mathbb{R}^m$. Also, in problem (2), the vector of coefficients $q(.) \in \mathbb{R}^{n_2}$, matrix of coefficients $W^T(.) \in \mathbb{R}^{m_2 \times n_2}$, demand vector $h(.) \in \mathbb{R}^{m_2}$ and matrix $T(.) \in \mathbb{R}^{m_2 \times n}$ depend on the random variable $\omega$ with support space $\Omega$. Problem (2) is called the recourse problem of stochastic programming.

If in the second stage, the vector $q(\omega)$ and the matrix $W(\omega)$ do not depend on the random variable $\omega$, or, in other words, they are held constant $q(\omega) = q$ and $W(\omega) = W$, the problem (1) turns to a stochastic linear programming problem with a fixed recourse

$$Q(x, \omega) = \min_{y \in \mathbb{R}^{n_2}} \{q^T y \mid W^T y \geq h(\omega) - T(\omega)x\}. \tag{3}$$

In addition, assume that the problem (3) has an optimal solution i. e. $Q(x, \omega) \in \mathbb{R}$ for each $x \in X$ and $\omega \in \Omega$.

In this paper, the random variable is assumed to be discrete. If the random variable is continuous, the expectation of function $Q(x, \omega)$ can be approximated as the sum of $N$ functions using the Monte Carlo method and other numerical integration methods [18, 19]

$$\phi(x) = E(Q(x, \omega)) = \sum_{i=1}^{N} Q(x, \omega_i)\rho(\omega_i),$$

where $\rho(.)$ is a probability density function that satisfies the following conditions

$$\rho(\omega_i) \geq 0, \quad \sum_{i=1}^{N} \rho(\omega_i) = 1.$$

Therefore, the objective function of the problem (1) can be rewritten as follows:

$$f(x) = c^T x + \phi(x). \tag{4}$$

This function is piecewise linear [20, 27].

Problem (1) contains two types of variables: present variable $x$ and future variable $y$ that belong to different categories in the following sense. $x$ is a deterministic variable and an optimum $x$ is optimal for all scenarios $\omega \in \Omega$ while $y(\omega)$ is a random variable and an optimum $y(\omega)$ is optimal for a single scenario $\omega$. The decomposition methods exploit the nature of random variables and the structure of problem (1). All decomposition algorithms such as $L$-shaped [28] are based on the Benders decomposition [2] whose original goal was to solve mixed integer programming problems.

The decomposition methods split the original problem into a master problem (1) and a series of independent subproblems (2) for each $\omega \in \Omega$.

In general, the objective function (4) is nonlinear and non-smooth, and hence, prevents the use of a smooth optimization algorithm. In this paper, we propose a method based on a decomposition method that uses the quadratic approach to smooth the objective function (4). This method paves the way for an optimization method such as sequential linear programming that solves the master problem (1). We note that it is very expensive to evaluate the objective function (4) or the gradient of a smooth function due to the need to solve $N$ linear programs (3). For this reason, we focus on the augmented Lagrangian method for solving the subproblem (3).

This paper is organized as follows. In the next section, some properties of recourse problems are mentioned in brief; accordingly, there is an attempt to find a solution based on eliminating their undesirable properties. This attempt leads to the presentation of the concept of normal solutions for linear problems (Section 3). For convenience, in this paper Euclidean least 2-norm solution of linear programming problem is named normal solution [23]. To find normal solutions of recourse problems, the augmented Lagrangian method is used (Section 4). In Section (5), the generalized Newton algorithm is applied for solving the obtained unconstrained problem and the numerical results are also presented. Also, concluding remarks are given in Section 6.

We now describe our notation. Let $a = [a_i]$ be a vector in $R^n$. By $a_+$ we mean a vector in $R^n$ whose $ith$ entry is 0 if $a_i < 0$ and equals $a_i$ if $a_i \geq 0$. By $A^T$ we mean the transpose of matrix A, and $\nabla f(x_0)$ is the gradient of $f$ at $x_0$. For $x \in R^n$, $\|x\|$ and $\|x\|_\infty$ denote $2-$norm and infinity norm respectively. Also, for convenience the index $i$ of $\omega_i$ is omitted.

## 2. PROPERTIES OF RECOURSE PROBLEM

In this section, the properties of function $Q(x, \omega)$ are investigated which include convexity and piecewise linearity. Furthermore, considering that the function is non-differentiable, there is an attempt to approximate it to a differentiable function based on the following Theorem; and using the approximated function, optimization methods are presented for solving the problem (1). This discussion starts with the convexity property of $Q(x, \omega)$.

**Theorem 2.1.** For every $\omega$ given, the function $Q(x, \omega)$ defined in (3) is a convex function on $x \in \mathbb{R}^n$.

Proof. To prove see subsection (2.2) in [28]. □

Considering the convexity of function $Q(x, \omega)$ , it can be easily concluded that $\phi(x)$ is also a convex function and, as a result, the objective function of the problem (1); that is, $f(x)$ is a convex function.

Using dual the problem (3), function $Q(x, \omega)$ can be written as follows:

$$\begin{aligned} Q(x, \omega) = \max_{z \in \mathbb{R}^{m_2}} \quad & (h(\omega) - T(\omega)x)^T z \\ s.t. \quad & Wz = q, \qquad z \geq 0. \end{aligned} \tag{5}$$

One of difficulties in solving the problem (1) is that the objective function is non-differentiable at some points because we can rewrite recourse function as $Q(x, \omega) = \max_{j \in \{1,2,\ldots,J\}} (h(\omega) - T(\omega)x)^T z_j$ where $J$ is number of extreme points of the problem (5) [27]. Hence, the function $Q(x, \omega)$ is piecewise linear of $x$ for each realization $\omega$. As follows, an example is presented in order to show the piecewise, linearity property of function $Q(x, \omega)$ depend on vector $x$.

**Example 2.2.** Consider function $Q(x,\omega)$ as follows

$$\begin{aligned} Q(x,\omega) = \max_{z_1,z_2} \quad & (x+\omega)z_1 + (2x+3\omega)z_2 \\ s.t. \quad & z_1 + z_2 = 2, \\ & z_1, z_2 \geq 0. \end{aligned}$$

The dual of above problem is:

$$\begin{aligned} Q(x,\omega) = \min_{y} \quad & 2y \\ s.t. \quad & y \geq (x+\omega), \\ & y \geq (2x+3\omega). \end{aligned}$$

If $x + 2\omega \geq 0$ , the solution of this problem is $2x+3\omega$ ; otherwise, $x+\omega$ . Therefore, function $Q$ is obtained in the following way:

$$Q(x,\omega) = \begin{cases} 2(2x+3\omega), & x \geq -2\omega, \\ 2(x+\omega), & x \leq -2\omega. \end{cases}$$

It is obvious that this function is continuously convex and piecewise linear which is non-differentiable in $-2\omega$.

## 3. THE NORMAL SOLUTION

As mentioned in the previous section, the function $Q(x,\omega)$ is piecewise linear; therefore, if at any point there are more than one sub-gradients, then there will be infinite. It can be easily shown that vectors $-T^T(\omega)z^*(x,\omega)$ are the sub-gradients of this function in which $z^*(x,\omega)$ is a solution for the problem (5). In this section, the sub-gradient $-T^T(\omega)z^*(x,\omega)$ are investigated in which $z^*(x,\omega)$ is the normal solution for the problem (5). To this end, function $Q_\epsilon(x,\omega)$ can be defined as follows:

$$\begin{aligned} Q_\epsilon(x,\omega) = \max_{z\in\mathbb{R}^{m_2}} \quad & (h(\omega) - T(\omega)x)^T z - \frac{\epsilon}{2}\|z\|^2 \\ s.t. \quad & Wz = q, \qquad z \geq 0. \end{aligned} \tag{6}$$

The following theorem shows that, for the sufficiently small $\epsilon > 0$ , the solution of this problem is the normal solution of the problem (5).

**Theorem 3.1.** For functions $Q(x,\omega)$ and $Q_\epsilon(x,\omega)$ introduced in (5) and (6), the following can be presented:

1. $\exists\,\bar{\epsilon} > 0$ such that, for each $\epsilon \in (0,\bar{\epsilon}]$ , the solution for the problem (6) is the normal solution for the problem (5).

2. For each $\epsilon > 0$ , function $Q_\epsilon(x,\omega)$ is differentiable depend on $x$.

3. The gradient of function $Q_\epsilon(x,\omega)$ at point $x$ is $\nabla Q_\epsilon(x,\omega) = -T^T(\omega)z_\epsilon^*(x,\omega)$ in which $z_\epsilon^*(x,\omega)$ is the solution for problem (6).

Proof. To prove 1 see Theorem 2.1 in [21].

Also, 2 and 3 can be easily proved considering that function $Q_\epsilon(x,\omega)$ is the conjugate function of

$$p(z) = \begin{cases} \frac{\epsilon}{2}\|z\|^2, & z \in Z, \\ \infty, & z \notin Z, \end{cases}$$

where $Z = \{z \in \mathbb{R}^{m_2} : Wz = q, z \geq 0\}$ (see Theorems 23.5 and 26.3 in [26]). □

In optimization methods for solving the problem (1), the gradient of objective function is required but it is not differentiable. Therefore, the objective function of the problem (1) is substituted by $f_\epsilon(x) = \sum_{i=1}^{N} Q_\epsilon(x,\omega_i)\rho(\omega_i)$. According to the previous theorem, it can be found that for obtaining the gradient of function $f_\epsilon(x)$ in each iteration, we need the normal solution of $N$ linear programming problem (5). In this paper, the augmented Lagrangian method [15] is used for this purpose.

## 4. AUGMENTED LAGRANGIAN METHOD

In order to find the normal solution, dual penalty problem can be used [24]. The objective function of dual penalty problem is piecewise quadratic, convex and once-differentiable. Moreover, since the objective function gradient of dual penalty problem satisfies the Lipschitz conditions, generalized Hessian of this function exists everywhere [17, 22]. Utilizing these properties, Mangasarian used generalized Newton method for solving convex, piecewise quadratic and unconstrained minimization problems [24].

In the augmented Lagrangian method, the unconstrained maximization problem is solved which gives the projection of a point on the solution set of the problem (5).

Assume that $\hat{z}$ is an arbitrary vector. Consider the problem of finding the least 2-norm projection $\hat{z}_*$ of $\hat{z}$ on the solution set $Z_*$ of the problem (5)

$$\begin{aligned} &\frac{1}{2}\|\hat{z}_* - \hat{z}\|^2 = \min_{z \in Z_*} \frac{1}{2}\|z - \hat{z}\|^2, \qquad (7) \\ &Z_* = \{z \in \mathbb{R}^{m_2} : Wz = q,\ \xi^T z = Q(x,\omega),\quad z \geq 0\}. \end{aligned}$$

In this problem, vector $x$ and random variable $\omega$ are constants; therefore, for simplicity, this is assumed $\xi = h(\omega) - T(\omega)x$ and function $\hat{Q}(\xi)$ is defined in a way that $\hat{Q}(\xi) = Q(x,\omega)$.

Considering that the objective function of the problem (7) is strictly convex; therefore, its solution is unique. The Lagrange function of the problem (7) is as follows:

$$L(z,p,\beta,\hat{z},\xi) = \frac{1}{2}\|z - \hat{z}\|^2 + p^T(Wz - q) + \beta(\xi^T z - \hat{Q}(\xi)),$$

where $p \in \mathbb{R}^{n_2}$ and $\beta \in \mathbb{R}$ are Lagrange multipliers and $\xi, \hat{z}$ are constant values. The dual problem of (7) has the form:

$$\max_{\beta \in \mathbb{R}} \max_{p \in \mathbb{R}^{n_2}} \min_{z \in \mathbb{R}^{m_2}_+} L(z,p,\beta,\hat{z},\xi). \qquad (8)$$

We note that the solution of the inner minimization problem in (8) is (see [22, 24])

$$z = (\hat{z} + W^T p + \beta\xi)_+. \qquad (9)$$

By substituting (9) into $L(z, p, \beta, \hat{z}, \xi)$, we have

$$\hat{L}(p, \beta, \hat{z}, \xi) := \min_{z \in \mathbb{R}_+^{m_2}} L(z, p, \beta, \hat{z}, \xi) = q^T p - \frac{1}{2}\|(\hat{z} + W^T p + \beta\xi)_+\|^2 + \beta\hat{Q}(\xi) + \frac{1}{2}\|\hat{z}\|^2.$$

Therefore, the dual problem of (7) is given by the following formula

$$\max_{\beta \in \mathbb{R}} \max_{p \in \mathbb{R}^{n_2}} \hat{L}(p, \beta, \hat{z}, \xi). \tag{10}$$

The following theorem states that if $\beta$ is sufficiently large, solving the inner maximization problem in (10) gives the unique solution of the problem (7).

**Theorem 4.1.** Consider the following maximization problem

$$\max_{p \in \mathbb{R}^{n_2}} S(p, \beta, \hat{z}, \xi) \tag{11}$$

in which $\beta, \hat{z}, \xi$ are constants and function $S(p, \beta, \hat{z}, \xi)$ is introduced as follows:

$$S(p, \beta, \hat{z}, \xi) = q^T p - \frac{1}{2}\|(\hat{z} + W^T p + \beta\xi)_+\|^2. \tag{12}$$

Also, assume that the set $Z_*$ is non-empty and the rank of sub-matrix $W_l$ of $W$ corresponding to nonzero components of $\hat{z}_*$ is $n_2$. In such a case, there is $\beta^*$ which for all $\beta \geq \beta^*$, $\hat{z}_* = (\hat{z} + W^T p(\beta) + \beta\xi)_+$ is the unique and exact solution for the problem (7) where $p(\beta)$ is the point obtained from solving the problem (11).

P r o o f. See Theorem 2.1 in ref. [15]. □

Also, in special conditions, the solution for the problem (3) can be also obtained and the following theorem expresses this issue.

**Theorem 4.2.** Assume that the solution set $Z_*$ is non-empty. For each $\beta > 0$ and $\hat{z} \in Z_*$, $y_* = \frac{p(\beta)}{\beta}$ is one exact solution for the linear programming problem (3) where $p(\beta)$ is the solution for the problem (11).

P r o o f. See Theorem 3.1 in ref. [15]. □

According to the theorems mentioned above, augmented Lagrangian method presents the following iteration process for solving the problem (7):

$$\begin{aligned} p_{k+1} &\in \arg\max_{p \in \mathbb{R}^{n_2}} \{q^T p - \frac{1}{2}\|(z_k + W^T p + \beta\xi)_+\|^2, \\ z_{k+1} &= (z_k + W^T p_{k+1} + \beta\xi)_+. \end{aligned} \tag{13}$$

Where $z_0$ is an arbitrary vector and here we can use of zero vector as initial vector for obtaining normal solution of the problem (5).

For arbitrary $z_0$ and $\beta > 0$ , this process converges to solution $z_* \in Z_*$ in finite number of step $M$. Also, $\frac{p_{M+1}}{\beta}$ gives the exact solution for the problem (3) [15].

If $\beta$ is sufficiently large, i.e. $\beta \geq \beta^*$, and other conditions of Theorem 4.1 are established as well, this process presents the normal solution for the problem (5). To this end, the smallest $\beta^*$ which has the conditions of Theorem (4.1) can be obtained. Assume that $\hat{z}_*$ is considered as follows:

$$\hat{z}_* = \begin{bmatrix} \hat{z}_*^l \\ \hat{z}_*^d \end{bmatrix},$$

where $\hat{z}_*^l$ is a strictly positive sub-vector and $\hat{z}_*^d$ is the zero sub-vector of $\hat{z}_*$. Now, corresponding with sub-vectors $\hat{z}_*^l$ and $\hat{z}_*^d$, matrix $W$ and vectors $\xi$ and $\hat{z}$ can be shown in the following way:

$$\hat{z} = \begin{bmatrix} \hat{z}^l \\ \hat{z}^d \end{bmatrix}, \quad \xi = \begin{bmatrix} \xi^l \\ \xi^d \end{bmatrix}, \quad W = \begin{bmatrix} W^l & W^d \end{bmatrix}.$$

Also, if $v_* = \begin{bmatrix} v_*^l \\ v_*^d \end{bmatrix}$ is the vector solution slack of the problem (3) and there is the index set $\sigma = \{i : 1 \leq i \leq d, (v_*^d)^i > 0\}$, then the value of $\beta^*$ can be determined using the following relation [15]:

$$\beta^* = \begin{cases} \max\limits_{i \in \sigma} \frac{\hat{z}_i^d + W_i^T (W_l W_l^T)^{-1} W_l (\hat{z}_*^l - \hat{z}^l)}{(v_*^d)_i}, & \sigma \neq \emptyset, \\ \alpha > -\infty, & \sigma = \emptyset, \end{cases}$$

where $\alpha$ is an arbitrary number.

## 5. NUMERICAL RESULTS AND ALGORITHM

In each iteration of the process (13), one concave, piecewise quadratic, unconstrained maximization problem is solved. For solving it, the generalized Newton method can be used. As is known, the objective function of the above problem is only once-differentiable and the concept of generalized Hessian is used for this problem. It is known that the objective function's gradient of the problem (12) is

$$\nabla_p S(p, \beta, \hat{z}, \xi) = q - W(\hat{z} + W^T p + \beta\xi)_+,$$

and also

$$\begin{aligned} &\|\nabla_p S(p, \beta, \hat{z}, \xi) - \nabla_p S(\bar{p}, \beta, \hat{z}, \xi)\| \\ &= \|q - W(\hat{z} + W^T p + \beta\xi)_+ - (q - W(\hat{z} + W^T \bar{p} + \beta\xi)_+)\| \\ &\leq \|W\| \|(\hat{z} + W^T \bar{p} + \beta\xi)_+ - (\hat{z} + W^T p + \beta\xi)_+)\| \\ &\leq \|W\| \|W^T\| \|p - \bar{p}\|. \end{aligned}$$

This means that the gradient function satisfies the Lipschitz condition with the constant $k = \|W\| \|W^T\|$ . Therefore, generalized Hessian exists everywhere [17, 22] and is the symmetric matrix $n_2 \times n_2$ [15] as follows:

$$\partial_p^2 S(p, \beta, \hat{z}, \xi) = -WD(\kappa)W^T,$$

where $D(\kappa)$ is the diagonal matrix where the $i$th-diagonal element $\kappa$ equals to 1, if $\hat{z} + W^T p + \beta\xi > 0$ and equals to 0, if $\hat{z} + W^T p + \beta\xi \leq 0$.

In the algorithm, the generalized Hessian may be singular, thus we used a modified Newton. The direction in each iteration for solving (11) is obtained through the following relation:

$$d_s = -(\partial_p^2 S(p,\beta,\hat{z},\xi) - \delta I_{n_2})^{-1}(\nabla_p S(p,\beta,\hat{z},\xi)), \tag{14}$$
$$p_{s+1} = p_s + \lambda_s d_s,$$

where $\delta$ is a small positive number and $I_{n_2}$ is the identity matrix of order $n_2$, $\lambda_s$ is the suitable step length that Armijo algorithm [1, 24] is used for determining it.

As mentioned in Section 1, because of the large number of recourse problems, the speed of algorithm for solving these problems and obtaining the gradients of smooth approximated recourse functions is of fundamental importance; and as mentioned in Section 3, its gradient depends on its normal solution. In this work, augmented Lagrangian method is used for solving the recourse problems and obtaining their normal solutions. The advantage of this method is solving an unconstrained problem, which accelerates obtaining the normal solution for the recourse problems.

The proposed algorithm was applied to solve some recourse problems. Table 1 compares this algorithm with CPLEX v. 12.1 solver for quadratic convex programming problems (6). As is evident from Table (1), most of recourse problems could be solved more successful by the algorithm is based on augmented Lagrangian method and generalized Newton method (ALGN) than CPLEX package. This algorithm give us high accuracy and the solution with minimum norm in minimum time. Also we can find that CPLEX is better than the algorithm proposed for some recourse problems that the matrix are approximately square (Ex. line 4–8).

The test generator generates recourse problems. These problems are generated using the following MATLAB code:

```
%Sgen: Generate random solvable recourse problems:
%Input: m,n,d(ensity); Output: W,q,ξ;
m=input('Enter n2:')
n=input('Enter m2:')
d=input('Enter d:')
pl=inline('(abs(x)+x)/2')
W=sprand(n2,m2,d);W=100*(W-0.5*spones(W));
z=sparse(10*pl(rand(m2,1)));
q=W*z;
y=spdiags((sign(pl(rand(n2,1)-rand(n2,1)))),0,n2,n2)
*5*((rand(n2,1)-rand(n2,1)));
ξ=W'*y-10*spdiags((ones(m2,1)-sign(z)),0,m2,m2)*ones(m2,1));
format short e; nnz(W)/prod(size(W))
```

The algorithm considered for solving several recourse problems was run on a computer with 2.5 dual-core CPU and 4 GB memory in MATLAB 7.8 programming environment. Also, in the generated problems, recourse matrix $W$ is the Sparse matrix ($n_2 \times m_2$) with the density $d$. The constants $\beta$ and $\delta$ in the above algorithm in (14) were selected 1 and $10^{-8}$, respectively.

| recourse problem $\mathbf{n_2 \times m_2 \times d}$ | solver | $\|\mathbf{Wz} - \mathbf{q}\|_\infty$ | $\|\hat{\mathbf{Q}}(\xi) - \xi^{\mathbf{T}}\mathbf{z}\|$ | $\|\mathbf{z}\|$ | time |
|---|---|---|---|---|---|
| $100 \times 500 \times 0.1$ | **ALGN** | 8.3230e-011 | 5.8208e-011 | 6.5162e+001 | 1.9374e-002 |
| | **CPLEX** | 9.1927e-010 | 1.0801e-008 | 6.5162e+001 | 1.5094e-001 |
| $100 \times 1000 \times 0.01$ | **ALGN** | 1.2629e-011 | 7.6398e-011 | 6.7950e+001 | 1.1986e-002 |
| | **CPLEX** | 1.0232e-012 | 9.0949e-013 | 6.7950e+001 | 1.6749e-001 |
| $400 \times 1400 \times 0.1$ | **ALGN** | 7.3631e-010 | 2.6193e-010 | 1.3352e+002 | 1.1299e+001 |
| | **CPLEX** | 3.1957e-009 | 8.0618e-008 | 1.3352e+002 | 2.7906e-001 |
| $900 \times 1000 \times 0.1$ | **ALGN** | 1.1282e-008 | 4.9477e-010 | 8.8432e+001 | 6.7239e+000 |
| | **CPLEX** | 8.0490e-011 | 7.1850e-011 | 8.8429e+001 | 5.9187e-001 |
| $1000 \times 3000 \times 0.06$ | **ALGN** | 2.6132e-009 | 1.0128e-008 | 2.0147e+002 | 3.0346e+000 |
| | **CPLEX** | 1.0914e-011 | 3.7835e-010 | 2.0147e+002 | 9.6612e-001 |
| $2000 \times 3000 \times 0.08$ | **ALGN** | 6.9426e-007 | 3.9482e-006 | 2.6776e+002 | 4.4673e+001 |
| | **CPLEX** | 9.8368e-011 | 6.4028e-010 | 2.6764e+002 | 4.9726e+000 |
| $2000 \times 2800 \times 0.008$ | **ALGN** | 3.9826e-010 | 1.4552e-011 | 1.3296e+002 | 3.0546e+001 |
| | **CPLEX** | 6.8212e-013 | 2.7285e-012 | 1.3292e+002 | 3.4133e+000 |
| $2500 \times 3200 \times 0.08$ | **ALGN** | 5.6676e-008 | 3.5730e-008 | 1.4959e+002 | 7.3602e+001 |
| | **CPLEX** | 8.1855e-012 | 7.2760e-012 | 1.4958e+002 | 8.6231e+000 |
| $10 \times 1e4 \times 0.01$ | **ALGN** | 1.1928e-011 | 9.5497e-012 | 9.2708e+000 | 2.5644e-002 |
| | **CPLEX** | 3.4106e-013 | 4.5475e-013 | 9.2747e+000 | 1.8510e-001 |
| $10 \times 1e5 \times 0.01$ | **ALGN** | 7.2608e-011 | 4.6384e-011 | 1.2633e+001 | 2.6799e-001 |
| | **CPLEX** | 2.7875e-003 | 9.1892e-004 | 1.2633e+001 | 3.2820e-001 |
| $10 \times 1e6 \times 0.01$ | **ALGN** | 2.3571e-010 | 5.9117e-011 | 1.2860e+001 | 3.9303e+000 |
| | **CPLEX** | 1.8526e-002 | 1.2776e-003 | 1.2995e+001 | 2.4763e+000 |
| $1000 \times 10000 \times 0.03$ | **ALGN** | 1.9374e-009 | 2.2555e-009 | 8.5676e+001 | 2.4149e+000 |
| | **CPLEX** | 4.0927e-012 | 1.4552e-011 | 8.5677e+001 | 1.3033e+000 |
| $2000 \times 8000 \times 0.01$ | **ALGN** | 5.3298e-009 | 1.8084e-008 | 1.1423e+002 | 1.3925e+001 |
| | **CPLEX** | 9.0949e-013 | 6.5484e-011 | 1.1423e+002 | 4.4492e+000 |
| $1000 \times 2e5 \times 0.01$ | **ALGN** | 1.0538e-008 | 2.0198e-008 | 9.8036e+001 | 6.3172e+000 |
| | **CPLEX** | 2.4921e-004 | 1.0984e-003 | 9.8051e+001 | 6.5342e+000 |
| $100 \times 1e5 \times 0.002$ | **ALGN** | 6.6768e-010 | 2.9468e-010 | 3.6251e+001 | 2.3436e-001 |
| | **CPLEX** | 5.7040e-004 | 1.5700e-003 | 3.6252e+001 | 3.8267e-001 |
| $100 \times 1e5 \times 0.0005$ | **ALGN** | 1.2057e-011 | 9.5497e-012 | 3.0085e+001 | 3.3800e-001 |
| | **CPLEX** | 1.7345e-004 | 1.8042e-004 | 3.0085e+001 | 3.3691e-001 |
| $1e3 \times 1e5 \times 0.0002$ | **ALGN** | 3.1754e-008 | 2.7171e-010 | 1.0944e+002 | 3.6869e-001 |
| | **CPLEX** | 7.4164e-005 | 1.3418e-004 | 1.0944e+002 | 5.0654e-001 |
| $1e3 \times 2e3 \times 0.0001$ | **ALGN** | 5.0690e-013 | 5.6843e-014 | 3.7841e+001 | 8.1997e-002 |
| | **CPLEX** | 5.6843e-014 | 1.4211e-014 | 3.7841e+001 | 1.3760e-001 |
| $1e3 \times 1.5e3 \times 0.00006$ | **ALGN** | 4 8.4058e-014 | 3.5527e-015 | 8.3717e+000 | 6.9437e-003 |
| | **CPLEX** | 0 | 3.5527e-015 | 8.3717e+000 | 1.3759e-001 |
| $1000 \times 1100 \times 0.00002$ | **ALGN** | 5.5655e-013 | 1.7053e-013 | 3.8148e+001 | 2.2785e-002 |
| | **CPLEX** | 1.1369e-013 | 1.1369e-013 | 3.8148e+001 | 1.4130e-001 |
| $1e3 \times 1e5 \times 0.1$ | **ALGN** | 3.3084e-008 | 2.9919e-007 | 4.6576e+000 | 4.9411e+001 |
| | **CPLEX** | 4.9943e-006 | 4.6461e-006 | 4.6579e+000 | 4.2090e+001 |
| $1e3 \times 1e6 \times 0.001$ | **ALGN** | 2.9583e-010 | 1.5280e-010 | 1.2741e+002 | 4.9994e+000 |
| | **CPLEX** | 4.9070e-005 | 2.5601e-006 | 1.2763e+002 | 8.2551e+000 |
| $1e3 \times 1e6 \times 0.005$ | **ALGN** | 1.4971e-008 | 9.1677e-010 | 1.2700e+002 | 1.5840e+001 |
| | **CPLEX** | 5.8208e-011 | 5.4206e-010 | 1.2777e+002 | 1.9866e+001 |
| $1e3 \times 1e6 \times 0.00008$ | **ALGN** | 2.5196e-010 | 3.6925e-010 | 1.1885e+002 | 2.8980e+000 |
| | **CPLEX** | 1.2800e-005 | 3.4102e-006 | 1.1904e+002 | 2.6511e+000 |
| $1e3 \times 1e6 \times 0.000006$ | **ALGN** | 2.8258e-011 | 2.3192e-011 | 1.0308e+002 | 3.3927e+000 |
| | **CPLEX** | 3.4106e-009 | 8.1586e-009 | 1.0313e+002 | 2.1842e+000 |

**Tab. 1.** Comparative between augmented Lagranigian method (ALGN) and CPLEX solver.

In the following table, the first column indicates the size and density of matrix $W$, the third column indicates the feasibility of the primal problem (5) and the last column demonstrates time duration for solving this problem.[1]

## 6. CONCLUSION

The stochastic linear programming problems include many linear sub-problems. Therefore, the speed of solving inner sub-problems and obtaining their normal solution is of prime importance. In this paper, generalized Newton method based on augmented Lagrangian algorithm was proposed and used for solving linear sub-problems. As the numerical results show, this algorithm has appropriate speed in most of the problems and, specifically this can be observed in recourse problems with a rectangular matrix of coefficients ($W^T$) in which the number of constraints ($m_2$) is noticeably more than the number of variables ($n_2$). The more challenging is solving the problems which their coefficient matrix is square (the numbers of constraints and variables get closer to each other) and more time is needed by the algorithm for solving the problem. In some large square problems, the volume of calculation becomes more than the memory (the last row of the table); this issue and the high number of recourse problems in stochastic linear programming justify the requirement of an attempt for parallelizing the algorithm and its parallel application on the computers.

(Received November 30, 2011)

## REFERENCES

[1] L. Armijoo: Minimazation of functions havinig Lipschitz-continus first partial derivetives. Pacific J. Math. *16* (1966), 1–3.

[2] J. F. Benders: Partitioning procedures for solving mixed-variables programming problems. Numer. Math. *4* (1962), 238–252.

[3] J. R. Birge, X. Chen, and L. Qi: Newton Method for Stochastic Quadratic Programs with Recourse. AMR, School of Mathematics, UNSW, Sydney 1994.

[4] J. R. Birge and F. Louveaux: Introduction to stochastic programming. Springer Series in Operation Research, Springer-Verlag, New York 1997.

[5] M. Branda: Chance constrained problems: penalty reformulation and performance of sample approximation technique. Kybernetika *48* (2012), 105–122.

[6] X. Chen: A parallel BFGS-SQP method for stochastic linear programs. In: Computational Techniques and Applications, World Scientific 1995, pp. 67–74.

[7] X. Chen: Newton-type methods for stochastic programming. Math. Comput. Model. *31* (2000), 89–98.

[8] X. Chen, L. Qi, and R. S. Womersley: Newton's method for quadratic stochastic programs with recourse. J. Comp. Appl. Math. *60* (1995), 29–46.

[9] X. Chen and R. S. Womersley: A parallel inexact Newton method for stochastic programs with recourse. Ann. Oper. Res. *64* (1995), 113–141.

[1] The Matlab code of this paper is available from the authors upon request.

[10] X. Chen and R. S. Womersley: Newton's method for quadratic stochastic programs with recourse. J. Comput. Appl. Math. *60* (1995), 29–46.

[11] X. Chen and R. S. Womersley: Random test problems and parallel methods for quadratic programs and quadratic stochastic programs. Optim. Method. Softw. *13* (2000), 275–306.

[12] G. B. Dantzig: Linear programming under uncertainty. Management Sci. *1* (1995), 197–206.

[13] G. B. Dantzig and P. W. Glynn: Parallel processors for planning under uncertainty. Ann. Oper. Res. *22* (1990), 1–21.

[14] G. B. Dantzig and P. Wolfe: Decomposition principle for linear programs. Oper. Res. *8* (1960), 101–111.

[15] Yu. G. Evtushenko, A. I. Golikov, and N. Mollaverdi: Augmented lagrangian method for large-scale linear programming problem. Optim. Method. Softw. *20* (2005), 515–524.

[16] J. L. Higle and S. Sen: Stochastic decomposition: an algorithm for two stage linear programs with recourse. Math. Oper. Res. *16* (1991), 650–669.

[17] J.-B. Hiriart-Urruty, J. J. Strodiot, and V. H. Nguyen: Generalized Hessian matrics and second-order optimality condisions for problems CL1 data. Appl. Math. Optim. *11* (1984), 43–56.

[18] G. Infanger: Monte carlo(Importance) sampling within a benders decomposition algorithm for stochastic linear programs. Ann. Oper. Res. *39* (1992), 69–95.

[19] S. Joe and I. H. Sloan: Imbedded lattice rules for multidimensional integration. SIAM J. Numer. Anal. *29* (1992), 1119–1135.

[20] P. Kall and S. W. Wallace: Stochastic Programming. John Wiley and Sons, 1994.

[21] C. Kanzow, H. Qi and L. Qi: On the minimum norm solution of linear programs. J. Optim. Theory Appl. *116* (2003), 333–345.

[22] S. Ketabchi and E. Ansari-Piri: On the solution set of convex problems and its numerical application. J. Comput. Appl. Math. *206* (2007), 288–292.

[23] O. L. Mangasarian: Normal solutions of linear programs. Math. Program. Stud. *22* (1984), 206–216.

[24] O. L. Mangasarian: A Newton method for linear programming. J. Optim. Theory Appl. *121* (2004), 1–18.

[25] A. Prekopa: Probabilistic programming. In: Stochastic programming (A. Ruszczynski and A. Shapiro, eds.), Handbook in Operations Research and Management Science *10* (2003), pp. 267–352, Elsevier, Amsterdam.

[26] R. T. Rockafellar: Convex Analysis. Princeton University Press, Princeton 1970.

[27] S. A. Tarim and I. Miguel: A hybrid Bender's decomposition method for solving stochastic constraint programs with linear recourse. Lect. Notes Comput. Sci. *3978* (2006), 133–148.

[28] R. M. Van Slyke and R. Wets: $L-$shaped linear programs with applications to optimal control and stochastic programming. SIAM J. Appl. Math. *17* (1969), 638–663.

*Saeed Ketabchi, Department of Applied Mathematics, Faculty of Mathematical Sciences, Guilan University, P. O. Box 416351914, Rasht. Iran.*
*e-mail: sketabchi@guilan.ac.ir*

*Malihe Behboodi-Kahoo, Department of Applied Mathematics, Faculty of Mathematical Sciences, Guilan University, P. O. Box 416351914, Rasht. Iran.*
*e-mail: behboodi.m@webmail.guilan.ac.ir*