Solving Fuzzy Two-Stage Programming Problem with Discrete Fuzzy Vector

Caili Zhou#1, Shenghua Wang#2

#1 College of Mathematics and Computer Science, Hebei University, Baoding 071002 China
#2 Department of Mathematics and Physics, North China Electric Power University, Baoding 071002 China

Abstract—The fuzzy two-stage programming problem with discrete fuzzy vector is hard to solve. In this paper, in order to solve this class of model, we design a algorithm by which we solve its deterministic equivalent programming to obtain optimal solution. Finally, two numerical examples are provided for showing the effectiveness of this algorithm.

Keywords—Fuzzy programming, fuzzy variable; two-stage, expected value.

I. INTRODUCTION

Since real-world situations are often not deterministic, conventional mathematical programming models are incapable to tackle all practical decision problems. Since Zadeh's pioneering work\cite{Zadeh}, there are much research on possibility theory\cite{Dubois, Klir,Yazenin}. Fuzzy decision making models have provided an important aspect in dealing with practical decision making problems\cite{Buckley, Inuiguchi, Tanaka, Liu1, Liu2,Liu6,Liu7}.

Based on the credibility theory\cite{BLiuYKLiu}, some new fuzzy programming were presented\cite{Gao, Xiaoxiahuang, YKLiu}. In \cite{YKLiu}, the author gave a class of fuzzy programming called fuzzy programming with recourse or fuzzy two-stage programming. The fuzzy two-stage programming is used to solve the problem in which first a decision, called the first-stage decision, is given, after the uncertain information formulated by fuzzy variables is realized, the second-stage decision is taken. Therefore, this fuzzy two-stage programming is a dynamic one. For solving the model presented, the author designed a hybrid intelligent algorithm combining fuzzy simulation, neural network and genetic algorithm (GA). However, the algorithm designed only obtains the approximate optimal solution of the model.

To the best of the author's knowledge, there is no research on fuzzy programming problem with discrete fuzzy variables. The author tries to do something in this area. Under this consideration, to our purpose, in this paper we consider the fuzzy two-stage programming with discrete fuzzy variables. Since the computational complexity of the expected value of the fuzzy variable, this class of fuzzy two-stage programming is very hard to solve. In order to solve it, we design a algorithm that can obtain its optimal solution by finding a so-called \{textit{optimal realized value array.}\}

This paper is organized as follows. In Section 2, we recall some necessary preliminaries on the fuzzy variable and the fuzzy expected value operator. Then, we give a theorem that is a basis on which the algorithm is designed in Section 3. Finally, two numerical examples are given to show the effectiveness of the algorithm is available.
II. PRELIMINARIES

Let $\xi$ be a normalized discrete fuzzy variable with following possibility distribution function

$$
\mu(r) = \begin{cases} 
\mu_1, & \text{if } r = a_1 \\
\mu_2, & \text{if } r = a_2 \\
\vdots \\
\mu_n, & \text{if } r = a_n,
\end{cases}
$$

then the expected value of $\xi$ is computed by the following formula\cite{BLiuYKLiu}

$$E[\xi] = \sum_{i=1}^{n} a_i p_i,$$

where $a_i$s are assumed to satisfy $a_1 \leq a_2 \leq \cdots \leq a_n$, and the weights are given by

$$p_i = \frac{1}{2} (\max_{j=1}^{i} \mu_j - \max_{j=0}^{i-1} \mu_j) + \frac{1}{2} (\max_{j=1}^{n} \mu_j - \max_{j=i+1}^{n} \mu_j).$$

($\mu_0 = 0, \mu_{n+1} = 0$) for $i = 1, 2, \cdots, n$.

The fuzzy variables $\xi_1, \xi_2, \cdots, \xi_n$ are said to be independent if and only if

$$\text{Pos}\{\xi_i \in B_i, i = 1, 2, \cdots, n\} = \min_{i \leq i \leq n} \text{Pos}\{\xi \in B_i\}$$

for any sets $B_1, B_2, \cdots, B_n$ of $\mathcal{F}\cite{Liu5}$. The vector $(\xi_1, \cdots, \xi_n)$ is called a fuzzy vector if and only if $\xi_1, \xi_2, \cdots, \xi_n$ are fuzzy variables \cite{Liu5}. If all of the fuzzy variables $\xi_1, \xi_2, \cdots, \xi_n$ are the discrete ones, then fuzzy vector $(\xi_1, \cdots, \xi_n)$ is called a discrete one.

III. FUZZY TWO-STAGE MODEL

A. Fuzzy two-stage model with discrete fuzzy vector

In \cite{YKLiu}, a new class of fuzzy programming called fuzzy two-stage programming was presented as follows

$$
\begin{aligned}
\min_{x} \quad & c^T x + E_\xi[\min_{y} q(\xi^T) y] \\
\text{s.t.} \quad & Ax = b \\
& W(\xi) y + T(\xi) x = h(\xi) \\
& x \geq 0, y \geq 0
\end{aligned}
$$

(1)

where $x \in \mathbb{R}^{n_1}$ is the first-stage decision, $y \in \mathbb{R}^{m_1}$ is the second-stage decision. Corresponding to $x$ are the first-stage vector and matrices $c$, $b$ and $A$, of sizes $n_1 \times 1$, $m_1 \times 1$ and $m_1 \times n_1$, respectively. For a given realization $\xi$ of fuzzy vector $\xi$, the second-stage problem data $q(\xi), h(\xi), T(\xi)$ and $W(\xi)$ become known, where $q(\xi)$ is $n_2 \times 1$, $h(\xi)$ is $m_2 \times 1$, $T(\xi)$ is $m_2 \times n_2$, and $W(\xi)$ is $m_2 \times n_2$.

For each fixed first-stage decision $x$ and realization $\xi$, the second-stage decision $y^*$ is the optimal solution to the following model

$$
\begin{aligned}
\min_{y} \quad & q(\xi^T) y \\
\text{s.t.} \quad & W(\xi) y + T(\xi) x = h(\xi) \\
& y \geq 0.
\end{aligned}
$$

Let $Q(x) = E_\xi[Q(x, \xi)]$ and $Q(x, \xi) = \min_{y} \{q(\xi^T) y | W(\xi) y + T(\xi) x = h(\xi), y \geq 0\}$, then the model (1) is equivalent to the following programming

$$
\begin{aligned}
\min_{x} \quad & c^T x + Q(x) \\
\text{s.t.} \quad & Ax = b \\
& x > 0.
\end{aligned}
$$

On the fuzzy two-stage programming problem, the interested readers may refer to \cite{YKLiu}. 

ISSN: 2231-5373  http://www.ijmttjournal.org  Page 115
For the sake of the demand of the research, in the rest of this paper, we assume that 
\( \xi = (\xi_1, \cdots, \xi_m) \) in the model (1) is a discrete fuzzy vector with the following possibility distribution

\[
\mu(r) = \begin{cases} 
\mu_1, & \text{if } r = \hat{\xi}_1 \\
\mu_2, & \text{if } r = \hat{\xi}_2 \\
\vdots \\
\mu_s, & \text{if } r = \hat{\xi}_s 
\end{cases}
\]

where \( \xi_i \) are independent discrete variables for \( i = 1, \cdots, m \).

Note that the model (1) with the above discrete fuzzy vector is not equivalent to the following model:

\[
\begin{align*}
\min_{x,y} & \quad c^T x + \sum_{j=1}^{s} p_j q(\hat{\xi}_j)^T y_j \\
\text{s.t.} & \quad Ax = b \\
& \quad W(\hat{\xi}_j) y_j + T(\hat{\xi}_j) x = h(\hat{\xi}_j), \\
& \quad j = 1, \cdots, s \\
& \quad x, y_1, \cdots, y_j \geq 0
\end{align*}
\]

where

\[
p_k = \frac{1}{2} \left( \max_{i=1}^{k} \mu_i - \max_{i=0}^{k-1} \mu_i \right) + \frac{1}{2} \left( \max_{i=k}^{s} \mu_i - \max_{i=k+1}^{s+1} \mu_i \right)
\]

(\( \mu_0 = \mu_{s+1} = 0 \)) for \( k = 1, \cdots, s \).

Next we give a theorem by which a algorithm in the next subsection is designed to solve this class of fuzzy two-stage programming problem with the discrete fuzzy vector.

Theorem 1. Let \( x^* \) and \( y_1^*, \cdots, y_s^* \) be the optimal solutions of the following model

\[
\begin{align*}
\min_{x,y} & \quad c^T x + \sum_{j=1}^{s} p_j q(\hat{\xi}_1)^T y_1 + \cdots + p_s q(\hat{\xi}_s)^T y_s \\
\text{s.t.} & \quad Ax = b \\
& \quad W(\hat{\xi}_j) y_j + T(\hat{\xi}_j) x = h(\hat{\xi}_j), \\
& \quad j = 1, \cdots, s \\
& \quad x, y_1, \cdots, y_s \geq 0
\end{align*}
\]

where

\[
p_k = \frac{1}{2} \left( \max_{i=1}^{k} \mu_i - \max_{i=0}^{k-1} \mu_i \right) + \frac{1}{2} \left( \max_{i=k}^{s} \mu_i - \max_{i=k+1}^{s+1} \mu_i \right)
\]

(\( \mu_0 = \mu_{s+1} = 0 \)) for \( k = 1, \cdots, s \).

If the inequality

\[
q(\hat{\xi}_1)^T y_1^* \leq \cdots \leq q(\hat{\xi}_s)^T y_s^*,
\]

holds, then \( x^* \) is the optimal solution of the model (1).

Proof. Obviously, \( y_j^* \) is the optimal solution to the following model

\[
\begin{align*}
\min & \quad q(\hat{\xi}_j)^T y_j \\
\text{s.t.} & \quad W(\hat{\xi}_j) y_j + T(\hat{\xi}_j) x^* = h(\hat{\xi}_j) \\
& \quad y_j \geq 0,
\end{align*}
\]

otherwise, \( y_j^* \) can not be the optimal solution to the model (2) for \( j = 1, \cdots, s \).
Since \( y_1^*, \ldots, y_s^* \) satisfy the condition (4), and \( x^* \) is the feasible solution to the model (1), we have

\[
Q(x^*) = p_1q(\xi_1)^T y_1^* + \cdots + p_sq(\xi_s)^T y_s^*
\]

where \( p_j \) is computed by the formula (3) for \( j = 1, \ldots, s \). Noting that \( x^* \) is the optimal solution of the model (2), we have that \( x^* \) is also the optimal solution of the model (1). The proof is complete.

B. Solving of the model

According to Theorem 1, in order to solve the model (1), we need to look for a array \((r_1^*, \ldots, r_s^*)\) where \( r_i^* \in \{\xi_1, \ldots, \xi_s\} \) for \( i = 1, \ldots, s \), called the \emph{optimal realized value array}, such that \( y_1^*, \ldots, y_s^* \) satisfy the condition

\[
q(r_i^*)^T y_i^* \leq \cdots \leq q(r_s^*)^T y_s^*,
\]

where \( y_1^*, \ldots, y_s^* \) are the optimal solution of the following model

\[
\begin{align*}
\min_{x,y} & \quad c^T x + p_1q(r_1^*)^T y_1 + \cdots + p_sq(r_s^*)^T y_s \\
\text{s.t.} & \quad Ax = b \\
& \quad W(r_j^*)y_j + T(r_j^*)y_j = h(r_j^*), \\
& \quad j = 1, \ldots, s \\
& \quad x, y_1, \ldots, y_s \geq 0
\end{align*}
\]

and \( p_k \) are computed by the formula

\[
p_k = \frac{1}{k} (\max_{i=1}^k \mu_i - \max_{i=0}^{k-1} \mu_i) + \frac{1}{2} (\max_{i=k}^s \mu_i - \max_{i=k+1}^{s+1} \mu_i)
\]

\((\mu_0 = \mu_{s+1} = 0)\) where \( \mu_k \) is the possibility of realized value \( r_k^* \) of the fuzzy vector \( \xi \) for \( k = 1, \ldots, s \).

For finding the \emph{optimal realized value array} \((r_1^*, \ldots, r_s^*)\) to obtain the optimal solution of the model (1), we give the algorithm as follows.

\textbf{Algorithm}

\begin{enumerate}
  \item \textbf{Step 0.} Set \( t = 0 \). Let \((r_1^*, \ldots, r_s^*)\) be the current array. Randomly initialize the array \((r_1, \ldots, r_s)\) where \( r_k \in \{\xi_1, \ldots, \xi_s\} \) for \( k = 1, \ldots, s \). Let \( r \in \mathbb{R}^m \).
  \item \textbf{Step 1.} Let \((r_1^*, \ldots, r_s^*) = (r_1, \ldots, r_s)\).
  \item \textbf{Step 2.} In the model (6), let \( r_k^* = r_k^* \) for \( k = 1, \ldots, s \). After computing \( p_k \) according to the formula (7) for \( k = 1, \ldots, s \), solve the model (6) to obtain the optimal solution \( x^*, y_1^*, \ldots, y_s^* \).
  \item If \( y_1^*, \ldots, y_s^* \) satisfy the condition (5), then \((r_1^*, \ldots, r_s^*)\) is the \emph{optimal realized value array} and \( x^* \) is the optimal solution of the model (1), the algorithm stops. Otherwise, go to the next step.
  \item \textbf{Step 3.} Randomly take two number \( i \) and \( j \) from the set \( \{1, \ldots, s\} \) such that \( i < j \). Let \( r = r_i^* \), \( r_i^* = r_j^* \), and \( r_j^* = r \). Set \((r_1, \ldots, r_s) = (r_1^*, \ldots, r_s^*)\) and \( t = t + 1 \). Return step 1.
\end{enumerate}
IV. Numerical Examples

Example 1. Consider Example 13 in [13] as follows

\[
\min \quad 2x_1 + 3x_2 + E_\xi[\min 2y_1 + y_2]
\]
\[s.t. \quad x_1 \leq 1 \]
\[x_2 \leq 1 \]
\[y_1 + y_2 \geq 1 - x_1 \]
\[y_1 \geq x_1 - x_2 \]
\[x_1, x_2, y_1, y_2 \geq 0 \]

where \(\xi\) is a fuzzy variable taking the values 0, 1, and 2 with possibility \(\frac{1}{2}, 1\), and \(\frac{1}{2}\), respectively.

In [13], the above problem is converted to its deterministic equivalent programming

\[
\min \quad 2x_1 + 3x_2 + Q(x_1, x_2)
\]
\[s.t. \quad x_1 \leq 1 \]
\[x_2 \leq 1 \]
\[x_1, x_2 > 0 \]

where

\[
Q(x_1, x_2) = \begin{cases} 
-\frac{4}{3}x_1 - \frac{2}{3}x_2 + 2 & \text{if } x_1 + x_2 > 1 \\
-\frac{3}{2}x_1 - \frac{1}{2}x_2 + \frac{1}{2} & \text{if } x_1 + x_2 \leq 1.
\end{cases}
\]

By calculating the above deterministic programming, we obtain that the optimal solution of the original problem is \((x_1^*, x_2^*) = (0, 0)\), and the optimal value is \(\frac{5}{6}\).

For showing the effectiveness of the algorithm designed in subsection B, we apply it to solve Example 1.

Consider the following model

\[
\min_{x, y} \quad 2x_1 + 3x_2 + \sum_{j=1}^{3} p_j(2y_{j1} + y_{j2})
\]
\[s.t. \quad x_1 \leq 1 \]
\[x_2 \leq 1 \]
\[y_{j1} + y_{j2} \geq 1 - x_2, \quad j = 1, 2, 3 \]
\[y_{j1} \geq \tilde{\xi}_j - x_1 - x_2, \quad j = 1, 2, 3 \]
\[x_1, x_2, y_{j1}, y_{j2} \geq 0, \quad j = 1, 2, 3 \]

where \(\tilde{\xi}_j\) is the realized value of the fuzzy variable \(\xi\) with the possibility \(\mu_j\) for \(j = 1, 2, 3\), and \(p_j\) is computed by the formula

\[
p_j = \frac{1}{2}(\max_{i=1}^{j} \mu_i - \min_{i=0}^{j-1} \mu_i) + \frac{1}{2}(\max_{i=j}^{3} \mu_i - \min_{i=j+1}^{4} \mu_i) \quad (9)
\]

\((\mu_0 = \mu_4 = 0)\) for \(j = 1, 2, 3\).

Next we apply the algorithm to solve the example 1. In the first iteration we assume that \(\xi_1 = 2, \xi_2 = 0\) and \(\xi_3 = 1\) in the model (8), compute \(p_1 = \frac{1}{3}, p_2 = 0\) and \(p_3 = \frac{2}{3}\) according to the formula (9), then by solving the model (8) obtain the optimal solution

\[
(x_1^*, x_2^*, y_{11}^*, y_{12}^*, y_{21}^*, y_{22}^*, y_{31}^*, y_{32}^*) = (0.1812, 0, 1.8188, 0, 0.110.7978, 94.9196, 0.8188, 0).
\]

Since the inequality \(2y_{11}^* + y_{12}^* \leq 2y_{12}^* + y_{22}^* \leq 2y_{13}^* + y_{13}^*\) does not hold, \((x_1^*, x_2^*) = (0.1812, 0)\) is not the optimal solution of Example 1.

In the second iteration, letting \(\xi_1 = 0, \xi_2 = 1\) and \(\xi_3 = 2\) in the model (8) and computing the weights \(p_1 = \frac{1}{6}, p_2 = \frac{1}{3}\) and \(p_3 = \frac{1}{3}\) according to the formula (9), then we solve the model (8) and obtain the optimal solution
Since \(2y_{11}^* + y_{12}^* = 1, 2y_{12}^* + y_{22}^* = 2\) and \(2y_{13}^* + y_{13}^* = 4\), by Theorem 1, we know that \((x_1^*, x_2^*) = (0, 0)\) is the optimal solution of Example 1, and the optimal value is
\[
\frac{2x_1^* + 3x_2^* + \frac{1}{6}(2y_{11}^* + y_{12}^*) + \frac{1}{2}(2y_{12}^* + y_{22}^*) + \frac{1}{3}(2y_{13}^* + y_{13}^*)}{15}.
\]

Example 2. Consider the following two-stage programming problem
\[
\begin{align*}
\min & \quad x_3 + E_{\xi} \left[ \min 2\xi_1 y_1 - y_2 \right] \\
\text{s.t.} & \quad x_1 + x_2 + x_3 \leq 10 \\
& \quad x_1 - 2x_2 + x_3 \leq 15 \\
& \quad -x_1 + 2x_2 + 2\xi_2 y_1 + y_2 \leq 7 \\
& \quad x_1 + x_3 + y_1 + y_2 \leq 25 \\
& \quad x_1, x_2, x_3, y_1, y_2 \geq 0.
\end{align*}
\]
where \(\xi_1\) and \(\xi_2\) are two independent discrete fuzzy variables taking values 1, 2, and 2, 3 with possibility 0.4, 1 and 0.2, respectively. By solving the following model using the algorithm in subsection 3.2
\[
\begin{align*}
\min & \quad x_3 + \sum_{j=1}^{4} p_j (2\hat{\xi}_{1j} y_{1j} - y_{2j}) \\
\text{s.t.} & \quad x_1 + x_2 + x_3 \leq 10 \\
& \quad x_1 - 2x_2 + x_3 \leq 15 \\
& \quad -x_1 + x_2 + 2\hat{\xi}_2 y_{1j} + y_{2j} \leq 7, \ j = 1, 2, 3, 4 \\
& \quad x_1 + x_3 + y_{1j} + y_{2j} \leq 25, \ j = 1, 2, 3, 4 \\
& \quad x_1, x_2, x_3, y_{1i}, y_{2i} \geq 0, \ j = 1, 2, 3, 4.
\end{align*}
\]
where \((\hat{\xi}_{11}, \hat{\xi}_{21}) = (1, 3), (\hat{\xi}_{12}, \hat{\xi}_{22}) = (1, 1), (\hat{\xi}_{13}, \hat{\xi}_{23}) = (2, 1)\) and \((\hat{\xi}_{14}, \hat{\xi}_{24}) = (2, 3)\) with possibility \(\mu_1 = 0.2, \mu_2 = 0.4, \mu_3 = 1\) and \(\mu_4 = 0.2\), respectively, and \(p_1 = 0.1, p_2 = 0.1, p_3 = 0.7, p_4 = 0.1\) by the formula
\[
p_j = \frac{1}{2}(\max_{i=1}^{j-1} \mu_i - \max_{i=0}^{j-1} \mu_i) + \frac{1}{2}(\max_{i=1}^{j+1} \mu_i - \max_{i=j}^{j+1} \mu_i)
\]
\((\mu_0 = \mu_5 = 0)\) for \(j = 1, 2, 3, 4\), we obtain the optimal solution
\[
(x_1^*, x_2^*, x_3^*, y_{11}^*, y_{12}^*, y_{21}^*, y_{22}^*, y_{13}^*, y_{23}^*, y_{14}^*, y_{24}^*)
\]
\[
= (9, 0, 0, 0, 16, 0, 16, 0, 16, 0, 16).
\]
Since \(2\hat{\xi}_{11} y_{11}^* - y_{21}^* \leq 2\hat{\xi}_{12} y_{12}^* - y_{22}^* \leq 2\hat{\xi}_{13} y_{13}^* - y_{23}^* \leq 2\hat{\xi}_{14} y_{14}^* - y_{24}^*\), we know that \((x_1^*, x_2^*, x_3^*) = (9, 0, 0)\) is the optimal solution of Example 2, and \((y_{1j}^*, y_{2j}^*)\) is the second-stage optimal solution for the realized value \((\xi_{1j}, \xi_{2j})\) of the fuzzy vector \((\xi_1, \xi_2)\) and the first-stage optimal solution \((x_1^*, x_2^*, x_3^*)\) for \(j = 1, 2, 3, 4\), respectively.
References