Sumit Kumar Maiti, Sankar Kumar Roy

3 A probabilistic two-level programming in noncooperative game under fuzzy environment: comparative study

Abstract: This chapter develops a computational algorithm to solve a stochastic two-level programming in a noncooperative game using a fuzzy optimization technique. It also includes all parameters such as fuzzy variables except the constraints on the right side that follow an extreme value distribution. Mellin transformation is considered to convert the fuzzy numbers into the crisp value. Based on stochastic programming, probabilistic constraints are converted into the deterministic form. Furthermore, as the fuzzy programming is used for the degree of acceptance, we present a comparative study of nonlinear and linear membership functions for the acceptance corresponding to LINGO 15.0 iterative scheme and genetic algorithm to see its impact on optimization. A numerical experiment is considered in order to present the applicability and feasibility of the proposed algorithm. Finally, conclusion about the findings and outlook are presented.

Keywords: Bilevel programming, fuzzy number, GA, membership function, Stackelberg game, stochastic programming, satisfactory solution

3.1 Introduction

Two-level programming problem (TLPP) consists of two levels in any hierarchical decision process, and large numbers of research works have been investigated on TLPP in the past few decades ([19],[20],[24]). In TLPP, each decision maker (DM) optimizes his/her own objectives without considering the other one. Stackelberg game was first introduced by an economist H. von Stackelberg to solve noncooperative decision problems. In Stackelberg game, the leader has the capability to enforce his/her decision on the followers. Many researchers have carried out investigations on the Stackelberg game (cf [1–4]) Ref. [17]. In real-life-based applications, when the input data of any mathematical model are inexistent or scarce, data are considered as uncertain. To tackle the uncertainty problem, two principal resources say, fuzziness and randomness, are considered. Fuzzy programming (cf

Sumit Kumar Maiti, School of Applied Sciences and Humanities, Haldia Institute of Technology, West Bengal, India

Sankar Kumar Roy, Department of Applied Mathematics with Oceanology and Computer Programming, Vidyasagar University, West Bengal, India

[5, 6]) developed by characterizing the input data as fuzzy sets, and stochastic programming is also developed to express the input data as random variables. Using the Mellin transformation, all fuzzy numbers (triangular or trapezoidal) are reduced to a crisp form. The constraint parameters on the right side follow an extreme value distribution Ref. [22]. Then, using a stochastic programming, these constraints are converted to the deterministic form and the equivalent crisp problems are solved by LINGO 15.0 iterative scheme and the results are compared. This chapter includes the following:

- Mellin transformation is used for the fuzzy numbers (triangular or trapezoidal) into crisp value.
- The constraint parameters on the right side are considered to be an extreme value distribution.
- A computational algorithm is developed to solve a noncooperative game.

We develop a computational algorithm for solving a stochastic TLPP in a noncooperative game involving an extreme value distribution by fuzzy programming. Our aim is to study the impacts of different types of membership functions. For that instance, we consider a linear function and a nonlinear function. The research works of several authors related to this area are shown in Table 3.1.

Table 3.1: Research works of several authors related to this area.

Author(s)	Real number	Fuzzy number	Fuzzy random	Multichoice
Anandalingam and Apprey [1]	Yes			
Anandalingam and White [7]				Yes
Hamidi and Mishmast [8]		Yes		
Maity and Roy [2]		Yes		Yes
Roy and Maiti [4]			Yes	Yes
Sakawa and Matsui [9]			Yes	
Singh and Chakraborty [10]			Yes	
Yeh et al. [11]			Yes	
Youness et al. [12]		Yes		
Zheng et al. [13]	Yes			
Our chapter		Yes	Yes	

The sequence of the chapter is given as follows: Section 3.2 introduces the basic knowledge of fuzzy set. In Section 3.3, model formulation of stochastic TLPP in a noncooperative game involving an extreme value distribution is presented. The

computation algorithm of fuzzy optimization is discussed in Section 3.4. A numerical example is presented in Section 3.5. Results are reflected in Section 3.6. The conclusion and an outlook of future research directions are described in Section 3.7.

3.2 Preliminaries

Here, we recall the useful definitions of fuzzy set and related theorems that are to be required for our subsequent developments.

Definition 3.1 ([11]). Let $\tilde{A}$ be a fuzzy set defined on X , the universal set. Then $\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) : x \in X\}$, where $\mu_{\tilde{A}} : X \rightarrow [0, 1]$ is the membership function such that $\mu_{\tilde{A}}(x) = 0$ if $x \notin \tilde{A}$, $\mu_{\tilde{A}}(x) = 1$ if $x \in \tilde{A}$.

Definition 3.2 ([18]). Let $\tilde{A}$ be a fuzzy set defined on $\mathbb{R}$, the set of real numbers, which is said to be a fuzzy number if its membership function satisfies the following conditions:

- (a) $\mu_{\tilde{A}}(x) : \mathbb{R} \rightarrow [0, 1]$ is continuous,
- (b) $\mu_{\tilde{A}}(x) = 0 \quad \forall x \in (-\infty, c] \cup [b, \infty)$,
- (c) $\mu_{\tilde{A}}(x)$ is strictly increasing on $[c, d]$ and strictly decreasing on $[a, b]$, ($c < d < a < b$),
- (d) $\mu_{\tilde{A}}(x) = \rho \quad \forall x \in [d, a]$, where $0 < \rho \leq 1$.

Definition 3.3 ([14]). Let $\tilde{B} = (b_1, b_2, b_3)$ be a fuzzy number. Then $\tilde{B}$ is said to be a triangular fuzzy number if $\mu_{\tilde{B}}(x)$ is given by

$$\mu_{\tilde{B}}(x) = \begin{cases} \left(\frac{x-b_1}{b_2-b_1}\right), & \text{if } b_1 \leq x \leq b_2 \\ \left(\frac{b_3-x}{b_3-b_2}\right), & \text{if } b_2 \leq x \leq b_3 \\ 0, & \text{if otherwise} \end{cases} \quad (3.1)$$

Definition 3.4 ([14]). Let $\tilde{C} = (c_1, c_2, c_3, c_4)$ be a fuzzy number. Then $\tilde{C}$ is said to be a trapezoidal fuzzy number if $\mu_{\tilde{C}}(x)$ is given by

$$\mu_{\tilde{C}}(x) = \begin{cases} \left(\frac{x-c_1}{c_2-c_1}\right), & \text{if } c_1 \leq x \leq c_2 \\ 1, & \text{if } c_2 \leq x \leq c_3 \\ \left(\frac{c_4-x}{c_4-c_3}\right), & \text{if } c_3 \leq x \leq c_4 \\ 0, & \text{if otherwise} \end{cases} \quad (3.2)$$

Definition 3.5 ([15]). Let $g(x)$ be a probability density function (pdf). Then Mellin transform $M_X(r)$ of $g(x)$ is given by

$$M_X(r) = \int_0^{\infty} x^{r-1} g(x) dx$$

provided the integral exists.

3.3 Model formulation

This chapter discusses on a stochastic TLPP for a noncooperative game in which cost coefficients of both level objectives are either triangular or trapezoidal fuzzy numbers. Generally, the constraint coefficients in a TLPP are considered as crisp. But in real-life situations, the constraint parameters are imprecise. So, we consider that the constraint parameters are of two different natures. The left side of the constraints is a fuzzy number, whereas the right side follows an extreme value distribution. The following assumptions are considered to design the model:

- $\tilde{c}_{1j}$: fuzzy number (triangular or trapezoidal) for leader objectives, $j = 1, 2, \dots, n$;
- $\tilde{c}_{2j}$: fuzzy number (triangular or trapezoidal) for follower objectives, $j = 1, 2, \dots, n$;
- $\tilde{a}_{ij}$: a $m \times n$ matrix with fuzzy numbers;
- $\tilde{b}_i$: random variable follows an extreme value distribution.

Therefore, the model on a stochastic TLPP for a noncooperative game involving extreme value distribution is defined as follows:

P1

$$\begin{aligned}
 &\text{maximize} && Z_1(x) = \sum_{j=1}^n \tilde{c}_{1j} x_j \\
 &\text{maximize} && Z_2(x) = \sum_{j=1}^n \tilde{c}_{2j} x_j \\
 &\text{subject to} && \text{pr} \left(\sum_{j=1}^n \tilde{A}_{ij} x_j \leq \tilde{B}_i \right) \geq 1 - \gamma_i, \quad i = 1, 2, \dots, m \\
 &&& x_j \geq 0 \quad \forall j \quad \text{and} \quad 0 < \gamma_i < 1 \quad \forall i
 \end{aligned} \tag{3.3}$$

3.3.1 Deterministic form of a fuzzy number

Here, we present a defuzzification procedure of a triangular or trapezoidal fuzzy number.

3.3.1.1 Triangular fuzzy number

Let $\tilde{a} = (a_1, a_2, a_3)$ be the triangular fuzzy number. Then the crisp value is obtained by finding the expected value using the pdf corresponding to the membership function.

Therefore, the proportional pdf corresponding to $\tilde{a}$ is defined as follows:

$f_{\tilde{a}}(x) = d\mu_{\tilde{a}}(x)$, where $\mu_{\tilde{a}}(x)$ is given by eq. (3.1).

To find d such that $\int_{-\infty}^{\infty} f_{\tilde{a}}(x) dx = 1$, we get $d = \frac{2}{a_3 - a_1}$.

Then the proportional pdf corresponding to $\tilde{a}$ is given by

$$f_{\tilde{a}}(x) = \begin{cases} \frac{2(x-a_1)}{(a_2-a_1)(a_3-a_1)}, & \text{if } a_1 \leq x \leq a_2 \\ \frac{2(a_3-x)}{(a_3-a_2)(a_3-a_1)}, & \text{if } a_2 \leq x \leq a_3 \\ 0, & \text{if otherwise} \end{cases}$$

According to Definition 3.5, we obtain

$$M_{\tilde{a}}(r) = \int_0^{\infty} x^{r-1} f_{\tilde{a}}(x) dx = \int_{a_1}^{a_2} x^{r-1} \frac{2(x-a_1)}{(a_2-a_1)(a_3-a_1)} dx + \int_{a_2}^{a_3} x^{r-1} \frac{2(a_3-x)}{(a_3-a_2)(a_3-a_1)} dx$$

On integration, we obtain

$$M_{\tilde{a}}(r) = \frac{2}{(a_3-a_1)r(r+1)} \left[\frac{a_3(a_3^r - a_2^r)}{a_3-a_2} - \frac{a_1(a_2^r - a_1^r)}{a_2-a_1} \right]$$

Therefore, the mean ($E[\tilde{a}]$) and variance ($\sigma_{\tilde{a}}^2$) corresponding to $\tilde{a}$ are obtained as follows:

$$E[\tilde{a}] = M_{\tilde{a}}(2) = \frac{a_1 + a_2 + a_3}{3}$$

$$\sigma_{\tilde{a}}^2 = M_{\tilde{a}}(3) - [M_{\tilde{a}}(2)]^2 = \frac{a_1^2 + a_2^2 + a_3^2 - a_1a_2 - a_2a_3 - a_3a_1}{18}$$

Then the crisp value of $\tilde{a}$ is given by

$$\text{Crisp}(\tilde{a}) = a' = \frac{a_1 + a_2 + a_3}{3} \quad (3.4)$$

3.3.1.2 Trapezoidal fuzzy number

Let $\tilde{b} = (b_1, b_2, b_3)$ be the trapezoidal fuzzy number, and the proportional pdf corresponding to $\tilde{b}$ is defined as follows: $f_{\tilde{b}}(x) = c\mu_{\tilde{b}}(x)$, where $\mu_{\tilde{b}}(x)$ is given by eq. (3.2) and c is given by relation

$$\int_{-\infty}^{\infty} f_{\tilde{b}}(x) dx = 1 \quad (3.5)$$

From eq. (3.5), we obtain $c = \frac{2}{b_4 + b_3 - b_2 - b_1}$.

Therefore, the proportional pdf corresponding to $\tilde{B}$ is given by

$$f_{\tilde{b}}(x) = \begin{cases} \frac{2(x-b_1)}{(b_2-b_1)(b_4+b_3-b_2-b_1)}, & \text{if } b_1 \leq x \leq b_2 \\ \frac{2}{(b_4+b_3-b_2-b_1)}, & \text{if } b_2 \leq x \leq b_3 \\ \frac{2(b_4-x)}{(b_4-b_3)(b_4+b_3-b_2-b_1)}, & \text{if } b_3 \leq x \leq b_4 \\ 0, & \text{if otherwise} \end{cases}$$

According to Definition 3.5, we obtain

$$\begin{aligned} M_{\tilde{b}}(r) &= \int_0^{\infty} x^{r-1} f_{\tilde{b}}(x) dx \\ &= \int_{b_1}^{b_2} x^{r-1} \frac{2(x-b_1)}{(b_2-b_1)(b_4+b_3-b_2-b_1)} dx + \int_{b_2}^{b_3} x^{r-1} \frac{2}{(b_4+b_3-b_2-b_1)} dx \\ &\quad + \int_{b_3}^{b_4} x^{r-1} \frac{2(b_4-x)}{(b_4-b_3)(b_4+b_3-b_2-b_1)} dx \end{aligned}$$

On integration, we obtain

$$M_{\tilde{b}}(r) = \frac{2}{(b_4+b_3-b_2-b_1)r(r+1)} \left[\frac{(b_4^{r+1}-b_3^{r+1})}{b_4-b_3} - \frac{(b_2^{r+1}-b_1^{r+1})}{b_2-b_1} \right]$$

Then, the mean $(E[\tilde{b}])$ and variance $(\sigma_{\tilde{b}}^2)$ corresponding to $\tilde{b}$ are obtained as follows:

$$\begin{aligned} E[\tilde{b}] &= M_{\tilde{b}}(2) = \frac{1}{3} \left[(b_1+b_2+b_3+b_4) + \frac{(b_1b_2-b_3b_4)}{(b_4+b_3-b_2-b_1)} \right] \\ \sigma_{\tilde{b}}^2 &= M_{\tilde{b}}(3) - [M_{\tilde{b}}(2)]^2 = \frac{1}{6} \left[(b_1^2+b_2^2+b_3^2+b_4^2) + \frac{(b_1+b_2)(b_3^2+b_4^2) - (b_3+b_4)(b_1^2+b_2^2)}{(b_4+b_3-b_2-b_1)} \right] \end{aligned}$$

Then the crisp value of $\tilde{b}$ is given by

$$\text{Crisp}(\tilde{b}) = b' = \frac{1}{3} \left[(b_1+b_2+b_3+b_4) + \frac{(b_1b_2-b_3b_4)}{(b_4+b_3-b_2-b_1)} \right] \quad (3.6)$$

Using eq. (3.4) (or eq. (3.6)), model 1 is transformed into the following model with the probabilistic constraint as follows:

P2

$$\begin{aligned}
 &\text{maximize} & Z_1(x) &= \sum_{j=1}^n c'_{1j} x_j \\
 &\text{maximize} & Z_2(x) &= \sum_{j=1}^n c'_{2j} x_j \\
 &\text{subject to} & \text{pr} \left(\sum_{j=1}^n \tilde{A}_{ij} x_j \leq \bar{B}_i \right) &\geq 1 - \gamma_i, \quad i = 1, 2, \dots, m \\
 & & x_j &\geq 0 \quad \forall j \quad \text{and} \quad 0 < \gamma_i < 1 \quad \forall i
 \end{aligned}$$

3.3.2 Deterministic form of probabilistic constraint

Here, $\bar{B}_i \quad \forall i$ are independent random variables that follow an extreme value distribution with δ_i and β_i as location and scale parameters, respectively, where the aspiration level is γ_i , $0 < \gamma_i < 1$.

Now, the pdf of $\bar{B}_i \quad \forall i$ is defined as follows:

$$\begin{aligned}
 f(\bar{B}_i) &= \frac{1}{\beta_i} \exp \left(- \left(\frac{\bar{B}_i - \delta_i}{\beta_i} \right) \right) \exp \left(- \left(\exp \left(- \left(\frac{\bar{B}_i - \delta_i}{\beta_i} \right) \right) \right) \right), \\
 &\quad -\infty < \bar{B}_i < \infty \quad \text{and} \quad \delta_i > 0, \beta_i > 0 \quad \forall i
 \end{aligned} \tag{3.7}$$

Therefore, eq. (3.3) can be written as follows:

$$\int_{\sum_{j=1}^n A'_{ij} x_j}^{\infty} f(\bar{B}_i) d\bar{B}_i \geq 1 - \gamma_i \tag{3.8}$$

Using eq. (3.7), the above eq. (3.8) can be written as

$$\int_{\sum_{j=1}^n A'_{ij} x_j}^{\infty} \frac{1}{\beta_i} \exp \left(- \left(\frac{\bar{B}_i - \delta_i}{\beta_i} \right) \right) \exp \left(- \left(\exp \left(- \left(\frac{\bar{B}_i - \delta_i}{\beta_i} \right) \right) \right) \right) d\bar{B}_i \geq 1 - \gamma_i \tag{3.9}$$

Let us take $\exp \left(- \left(\frac{\bar{B}_i - \delta_i}{\beta_i} \right) \right) = q$. Then eq. (3.9) reduces to

$$\exp \left(- \left(\frac{\sum_{j=1}^n A'_{ij} x_j - \delta_i}{\beta_i} \right) \right) \Big|_0^{e^{-q}} \geq 1 - \gamma_i.$$

On integration, we get

$$\exp \left(- \exp \left(- \left(\frac{\sum_{j=1}^n A'_{ij} x_j - \delta_i}{\beta_i} \right) \right) \right) \leq 1 - \gamma_i$$

Taking logarithm of both sides two times and rearranging, we get

$$\sum_{j=1}^n A'_{ij} x_j \leq \delta_i - \beta_i [\ln(-\ln(\gamma_i))] \quad (3.10)$$

Then we obtained the following deterministic model:

P3

$$\begin{aligned} & \text{maximize} \quad Z_1(x) = \sum_{j=1}^n c'_{1j} x_j \\ & \text{maximize} \quad Z_2(x) = \sum_{j=1}^n c'_{2j} x_j \\ & \text{subject to} \quad \sum_{j=1}^n A'_{ij} x_j \leq \delta_i - \beta_i [\ln(-\ln(\gamma_i))] \\ & \quad \quad \quad x_j \geq 0 \quad \forall j \text{ and } 0 < \gamma_i < 1, \delta_i > 0, \beta_i > 0 \quad \forall i \end{aligned}$$

3.4 Computational algorithm

After getting the deterministic form of P1, we develop the following two algorithms corresponding to the satisfactory solution using fuzzy approach. The following steps of fuzzy programming are given:

Algorithm 3.1 (For linear membership function).

Step 4.1: Independently solve the leader and follower objectives with respect to the constraints and assign the best (Z_s^0) and worst (Z_s^1) solutions for each objective function, $s = 1, 2$.

Step 4.2: A linear membership function is considered for each objective function as follows:

$$\mu_s(Z_s(x)) = \begin{cases} 1, & \text{if } Z_s(x) \leq Z_s^0 \\ \left(\frac{Z_s(x) - Z_s^1}{Z_s^0 - Z_s^1} \right) & \text{if } Z_s^1 \leq Z_s(x) \leq Z_s^0 \\ 0, & \text{if } Z_s(x) \geq Z_s^1 \end{cases}$$

Again, considering a linear membership function for the upper-level decision variable (x_1) with f tolerance value is defined by

$$\mu_{x_1}(x_1) = \begin{cases} \left(\frac{x_1 - x_1^t + f}{f} \right), & \text{if } x_1^t - f \leq x_1 \leq x_1^t \\ \left(\frac{x_1^t + f - x_1}{f} \right), & \text{if } x_1^t \leq x_1 \leq x_1^t + f \\ 0, & \text{if otherwise} \end{cases}$$

where x_1^t is the solution set of the top-level DM.

Step 4.3: According to Zimmerman approach [16], solve the following problem:

P4

$$\begin{aligned} & \text{maximize} \quad \lambda \\ & \text{subject to} \quad \mu_s(Z_s(x)) \geq \lambda \quad \forall s \\ & \quad \mu_{x_1}(x_1) \geq \lambda, \\ & \quad \text{eq. (3.10),} \\ & \quad x \geq 0 \text{ and } \lambda \in [0, 1] \end{aligned}$$

If the top-level DM is satisfied with the obtained optimal result, then the solutions become a satisfactory solution and go to Step 4.4. Otherwise, top-level DM updated his/her membership function and then go to Step 4.2.

Step 4.4: Stop.

Algorithm 3.2 (For nonlinear membership function).

Repeat Step 4.1 and follow the steps.

Step 4.2: Consider a nonlinear membership for each objective, and the top-level decision variable (x_1) with tolerance value f is given by

$$\mu_s(Z_s(x)) = \begin{cases} 1, & \text{if } Z_s(x) > Z_s^0 \\ \left(\frac{e^{-r \left(\frac{Z_s(x) - Z_s^1}{Z_s^0 - Z_s^1} \right)} - e^{-r}}{1 - e^{-r}} \right), & \text{if } Z_s^1 \leq Z_s(x) \leq Z_s^0 \\ 0, & \text{if } Z_s(x) \leq Z_s^1 \end{cases}$$

$$\mu_{x_1}(x_1) = \begin{cases} \frac{e^{-r \left(\frac{x_1 - x_1^t + f}{f} \right)} - e^{-r}}{1 - e^{-r}}, & \text{if } x_1^t - f \leq x_1 \leq x_1^t \\ \frac{e^{-r \left(\frac{x_1^t + f - x_1}{f} \right)} - e^{-r}}{1 - e^{-r}}, & \text{if } x_1^t \leq x_1 \leq x_1^t + f \\ 0, & \text{if otherwise} \end{cases}$$

where x_1^t is the solution of the top-level DM. After that repeat Step 4.3 and Step 4.4.

3.5 Numerical experiment

Here, we present two numerical examples corresponding to triangular and trapezoidal fuzzy numbers and one practical application.

3.5.1 Example 1

Let us consider the following example:

P5

$$\begin{aligned}
 &\text{maximize} && Z_1(x) = (200, 302, 388)x_1 + (234, 280, 347)x_2 + (187, 199, 223)x_3 + (207, 289, 301)x_4 \\
 &\text{maximize} && Z_2(x) = (89, 107, 112)x_1 + (106, 127, 132)x_2 + (67, 80, 93)x_3 + (73, 79, 104)x_4 \\
 &\text{subject to} && (34, 42, 49)x_1 + (27, 37, 52)x_2 + (30, 47, 53)x_3 + (18, 22, 28)x_4 \leq b_1 \\
 &&& (32, 37, 42)x_1 + (18, 22, 25)x_2 + (25, 27, 29)x_3 + (30, 32, 36)x_4 \leq b_2 \\
 &&& (26, 36, 42)x_1 + (16, 20, 27)x_2 + (41, 44, 49)x_3 + (51, 55, 59)x_4 \geq b_3 \\
 &&& (38, 43, 47)x_1 + (52, 56, 57)x_2 + (61, 65, 68)x_3 + (43, 45, 49)x_4 \geq b_4 \\
 &&& x_j \geq 0 \quad \forall j
 \end{aligned}$$

The location and scale parameters and specified probability levels of b_1, b_2, b_3, b_4 are given in Table 3.2.

Table 3.2: Input parameters for $b_i, i = 1, 2, 3, 4$.

Random parameters b_i	Location parameter (δ_i)	Scale parameter (β_i)	Specified probability levels (γ_i)
b_1	800	200	0.5
b_2	600	100	0.6
b_3	150	75	0.7
b_4	86	25	0.8

Using eqs. (3.4) and (3.10), P5 can be written in the following model:

P6

$$\begin{aligned}
 &\text{maximize} && Z_1(x) = 296.67x_1 + 287x_2 + 203x_3 + 265.67x_4 \\
 &\text{maximize} && Z_2(x) = 102.67x_1 + 121.67x_2 + 80x_3 + 85.33x_4 \\
 &\text{subject to} && 41.67x_1 + 38.67x_2 + 43.33x_3 + 22.67x_4 \leq 873.3 && (3.11)
 \end{aligned}$$

$$37x_1 + 21.67x_2 + 27x_3 + 32.67x_4 \leq 667.17 \quad (3.12)$$

$$34.67x_1 + 21x_2 + 44.67x_3 + 55x_4 \geq 136.08 \quad (3.13)$$

$$42.67x_1 + 55x_2 + 64.67x_3 + 45.67x_4 \geq 74.10 \quad (3.14)$$

$$x_j \geq 0 \quad \forall j$$

Using Algorithm 3.1 of computational algorithm, P6 is converted to a linear programming problem, that is, P7.

P7

$$\begin{aligned} & \text{maximize} && \lambda \\ & \text{subject to} && \left(\frac{Z_1(x) - 618.407}{6730.487} \right) \geq \lambda \\ & && \left(\frac{Z_2(x) - 211.122}{2661.279} \right) \geq \lambda \\ & && \left(\frac{x_1 + x_2 - 17.363 + f}{f} \right) \geq \lambda \\ & && \text{constraints (3.11) – (3.14)} \\ & && x_j \geq 0 \quad \forall j \end{aligned}$$

Now, P7 is solved for $f = 0.1$ by using LINGO 15.0 iterative scheme and genetic algorithm (GA), and the optimum results are presented in Table 3.3.

Table 3.3: Optimal results for Algorithm 3.1.

Method	λ	Z_1	Z_2	x_1	x_2	x_3	x_4
LINGO	0.998	7,335.433	2,867.078	0.127	17.297	0.0	8.783
GA	0.899	7,338.571	2,848.670	0.216	16.549	0.0	9.504

Also, using Algorithm 3.2 of computational algorithm, P6 is transformed into the following model:

P8

$$\begin{aligned} & \text{maximize} && \lambda \\ & \text{subject to} && \frac{e^{-r} \left(\frac{Z_1(x) - 618.407}{6730.487} \right) - e^{-r}}{1 - e^{-r}} \geq \lambda \\ & && \frac{e^{-r} \left(\frac{Z_2(x) - 211.122}{2661.279} \right) - e^{-r}}{1 - e^{-r}} \geq \lambda \\ & && \frac{e^{-r} \left(\frac{x_1 + x_2 - 17.363 + f}{f} \right) - e^{-r}}{1 - e^{-r}} \geq \lambda \\ & && \text{constraints (3.11) – (3.14)} \\ & && x_j \geq 0 \quad \forall j \end{aligned}$$

Using LINGO 15.0 iterative scheme and GA, P8 is solved for $r = 0.5$ and $f = 0.1$, and the results are presented in Table 3.4.

Table 3.4: Optimal results for Algorithm 3.2.

Method	λ	Z_1	Z_2	x_1	x_2	x_3	x_4
LINGO	0.99999	2,717.578	1,035.560	0.00001	3.70111	3.61948	3.46520
GA	0.99956	2,718.753	1,037.992	0.00005	3.7811	3.5989	3.3989

3.5.2 Example 2

Again, let us take another example:

P9

$$\begin{aligned}
 &\text{maximize } Z_1(x) = (200, 302, 388)x_1 + (234, 280, 347)x_2 + (187, 199, 223)x_3 + (207, 289, 301)x_4 \\
 &\text{maximize } Z_2(x) = (89, 107, 112)x_1 + (106, 127, 132)x_2 + (67, 80, 93)x_3 + (73, 79, 104)x_4 \\
 &\text{subject to } (34, 42, 49)x_1 + (27, 37, 52)x_2 + (30, 47, 53)x_3 + (18, 22, 28)x_4 \leq b_1 \\
 &\quad (32, 37, 42)x_1 + (18, 22, 25)x_2 + (25, 27, 29)x_3 + (30, 32, 36)x_4 \leq b_2 \\
 &\quad (26, 36, 42)x_1 + (16, 20, 27)x_2 + (41, 44, 49)x_3 + (51, 55, 59)x_4 \geq b_3 \\
 &\quad (38, 43, 47)x_1 + (52, 56, 57)x_2 + (61, 65, 68)x_3 + (43, 45, 49)x_4 \geq b_4 \\
 &\quad x_j \geq 0 \quad \forall j
 \end{aligned}$$

The location and scale parameters and specified probability levels of b_1, b_2, b_3, b_4 are given in Table 3.5.

Table 3.5: Input parameters for $b_i, i = 1, 2, 3, 4$ corresponding to Example 2.

Random parameters b_i	Location parameter (δ_i)	Scale parameter (β_i)	Specified probability levels (γ_i)
b_1	1,200	100	0.02
b_2	1,400	200	0.03
b_3	200	140	0.05
b_4	500	120	0.07

Using eqs. (3.6) and (3.10), P9 is reduced to the following model:

P10

$$\begin{aligned}
 &\text{maximize} && Z_1(x) = 15.29x_1 + 24x_2 + 30.55x_3 + 44x_4 \\
 &\text{maximize} && Z_2(x) = 8.7x_1 + 15.19x_2 + 6.42x_3 + 5.43x_4 \\
 &\text{subject to} && 16.29x_1 + 34.09x_2 + 7.8x_3 + 34.21x_4 \leq 1,063.59 \\
 &&& 14x_1 + 24.4x_2 + 57.22x_3 + 54.43x_4 \leq 1,149.07 \\
 &&& 25.62x_1 + 29.3x_2 + 49.64x_3 + 54.3x_4 \geq 615.83 \\
 &&& 4x_1 + 25.81x_2 + 67.29x_3 + 26.81x_4 \geq 814.78 \\
 &&& x_j \geq 0 \quad \forall j
 \end{aligned}$$

Using Algorithms 3.1 and 3.2 of computational algorithm, optimal results are shown in Tables 3.6 and 3.7, respectively.

Table 3.6: Optimal results for Algorithm 3.1.

Method	λ	F	Z_1	Z_2	x_1	x_2	x_3	x_4
LINGO	0.998	4.4	1,045.933	549.124	40.598	10.503	5.667	0.0
GA	0.977	4.4	1,045.983	559.914	39.985	11.897	4.88	0.0

Table 3.7: Optimal results for Algorithm 3.2.

Method	λ	F	r	Z_1	Z_2	x_1	x_2	x_3	x_4
LINGO	0.999	4.4	0.3	379.686	80.292	0.091	0.0	12.382	0.0
GA	0.995	4.4	0.3	380.318	80.329	0.074	0.0	12.412	0.0

3.5.3 Practical application (production planning problem)

India bought crude oil from different cities of the World. That crude oil cannot be directly used in our daily life. So it should be refined. Under different circumstances, the crude oil produces various by-products from different process units. India has eight oil refineries in different locations such as Haldia, Gujarat, Panipath, Paradip, Mathura, Guwahati, Barauni, Bongaigaon. Each oil refinery has different process units and are discussed as follows:

1. FOB (fuel oil block): It has two atmospheric distribution units: CDU-I and II along with the naphtha pretreating unit, catalytic reforming unit, and kero hydrodesulfurization unit.

2. DHDS (diesel-hydrodesulfurization): It produces Euro-III quality eco-friendly diesel, which reduced sulfur dioxide in vehicular emissions and as a result pollution-free environment.
3. FCCU (fluidized catalytic cracking unit): It processes heavier feedstock, including a proportion of residue, and produces value-added products and contributes to improve the distillate yield from the refinery.
4. MSQIU (motor sprit quality improvement unit): It produces EURO-III petrol.
5. OHCU (once-through hydrocracker): It comprises hydrogen generation unit, sulfur recovery units, hydrogen storage facility, air compressor, heat recovery steam generation, effluent treatment plant, and feed preparation unit.
6. LOB (lube oil block): It consists of vacuum distillation unit, propane de-asphalting unit, lube hydrofinishing unit, vis-breaking unit and catalytic ISO dewaxing unit.
7. DHDT (diesel hydrotreating unit): It produces EURO-IV-grade MS and HSD (High Speed Diesel).
8. GT (gas turbines): It generates electricity by using gas turbine.

All the process units are not available in all refineries. Due to fluctuation of global market and plant equipments, the capacity of the process units in each refinery is not fixed. So considering these capacities are fuzzy numbers (triangular or trapezoidal). We mainly discuss here six types of oil refineries and their capacities that are supplied in Tables 3.8 and 3.9, respectively.

Table 3.8: Input data for FOB, DHDS, FCCU, and MSQIU units.

Refineries	Process units			
	FOB	DHDS	FCCU	MSQIU
Haldia	(5,5.5,7)	(6,6.3,6.5)	(10,12,12.3)	(11,11.2,11.4)
Gujarat	(1.1,1.4,1.5)	(2.1,2.3,2.6)	(2.4,2.41,2.43)	(2.13,2.16,2.17)
Panipath	(1.9,2.3,2.7)	–	(1.2,1.3,1.7)	–
Paradip	(0.47,0.49,1)	(0.57,1,1.1)	–	(0.39,0.45,0.5)
Mathura	(5.6,5.86,5.89)	(6.2,6.5,7.2)	(6.15,6.19,6.34)	(5.1,5.15,5.24)
Guwahati	(2.12,2.15,2.17)	–	(2.18,2.19,2.21)	(1.89,1.92,1.95)

Let us consider that $x_1, x_2, x_3, x_4, x_5, x_6, x_7$, and x_8 be the decision variables (quantity) corresponding to the process units FOB, DHDS, FCCU, MSQIU, OHCU, LOB, DHDT, and GT, respectively. So the problem gives a TLPP where the top-level DM maximizes the profit and the lower one also maximizes the products. The cost

Table 3.9: Input data for OHCU, LOB, DHDT, and GT units.

Refineries	Process units			
	OHCU	LOB	DHDT	GT
Haldia	(13,13.2,13.5)	(12,12.6,14)	(7.2,7.5,7.9)	(5.5,5.9,6.3)
Gujarat	(1.8,2.1,2.3)	(1.12,1.14,1.17)	(1.43,1.45,1.48)	(2.6,2.9,3.1)
Panipath	(2.1,3.3,3.4)	(3.21,3.42,3.46)	–	(3.42,3.45,4)
Paradip	–	(0.31,0.38,0.47)	(0.67,0.69,0.72)	–
Mathura	(4.3,4.4,4.49)	–	(4.18,4.24,4.38)	(4.67,4.75,4.86)
Guwahati	–	(2.15,2.19,2.27)	(1.73,1.79,1.83)	–

coefficients of both level objective functions and location, shape, and specified probability levels of oil refineries are shown in Tables 3.10 and 3.11, respectively. Using Tables 3.8–3.11, model P3 is transformed into the following model:

Table 3.10: Input data for cost coefficients.

$c_{11} = (800, 820, 850)$	$c_{12} = (380, 420, 435)$	$c_{13} = (305, 310, 325)$
$c_{14} = (210, 225, 240)$	$c_{15} = (180, 197, 213)$	$c_{16} = (170, 175, 180)$
$c_{17} = (150, 156, 159)$	$c_{18} = (140, 148, 153)$	$c_{21} = (80, 85, 89)$
$c_{22} = (52, 53, 55)$	$c_{23} = (46, 48, 49)$	$c_{24} = (32, 35, 38)$
$c_{25} = (27, 27.6, 28)$	$c_{26} = (17, 17.6, 17.9)$	$c_{27} = (12, 12.9, 13)$
$c_{28} = (8, 8.7, 8.9)$		

Table 3.11: Input parameters for location, shape, and probability levels.

Refineries	Location parameter (δ_i)	Scale parameter (β_i)	Specified probability levels (γ_i)
Haldia	200	20	0.05
Gujarat	64	10	0.06
Panipath	19	9	0.07
Paradip	6	3	0.07
Mathura	5	3	0.08
Guwahati	7	2	0.09

P11

$$\begin{aligned}
&\text{maximize } Z_1(x) = 823.33x_1 + 411.67x_2 + 313.33x_3 + 225x_4 + 196.67x_5 + 175x_6 + 155x_7 + 147x_8 \\
&\text{maximize } Z_2(x) = 84.67x_1 + 53.33x_2 + 47.67x_3 + 35x_4 + 27.53x_5 + 17.5x_6 + 12.63x_7 + 8.53x_8 \\
&\text{subject to } 5.83x_1 + 6.27x_2 + 11.43x_3 + 11.2x_4 + 13.23x_5 + 12.87x_6 + 7.53x_7 + 5.9x_8 \leq 178.06 \\
&\quad 1.33x_1 + 2.33x_2 + 2.41x_3 + 2.15x_4 + 2.07x_5 + 1.14x_6 + 1.45x_7 + 2.87x_8 \leq 53.66 \\
&\quad 2.3x_1 + 1.4x_3 + 2.93x_5 + 3.36x_6 + 3.62x_8 \leq 10.20 \\
&\quad 0.65x_1 + 0.89x_2 + 0.45x_4 + 0.39x_6 + 0.69x_7 \geq 13.87 \\
&\quad 5.78x_1 + 6.63x_2 + 6.23x_3 + 5.14x_4 + 4.4x_5 + 4.27x_7 + 4.76x_8 \geq 12.45 \\
&\quad 2.15x_1 + 2.19x_3 + 1.92x_4 + 2.2x_6 + 1.78x_7 \geq 11.72 \\
&\quad x_j \geq 0 \quad \forall j
\end{aligned}$$

Using Algorithms 3.1 and 3.2 of computational algorithm, optimal solutions are obtained from P11 by LINGO 15.0 iterative scheme and GA and put into Tables 3.12 and 3.13, respectively.

Table 3.12: Optimal results for Algorithm 3.1.

Method	Z_1	Z_2	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
LINGO	11,936.45	1,448.545	4.435	19.574	0.0	0.64	0.0	0.0	0.537	0.0
GA	12,141.87	1,448.597	5.568	17.774	0.0	0.58	0.0	0.0	0.71	0.0

Table 3.13: Optimal results for Algorithm 3.2.

Method	Z_1	Z_2	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
LINGO	3,447.215	304.995	0.0	0.0	0.0	1.542	0.0	0.322	18.914	0.764
GA	3,447.324	306.866	0.0	0.0	0.0	1.627	0.0	0.413	18.728	0.722

3.6 Discussion

In our chapter, we solved a stochastic TLPP for a noncooperative game under fuzzy environment using linear and nonlinear membership functions. For Example 1, from Tables 3.3 and 3.4, it has been observed that the optimal solutions have been derived using LINGO 15.0 iterative scheme and GA, respectively. In case of linear membership function under GA, we have the best solution for $\lambda = 0.899$ and $d = 0.1$ corresponding

to both level DMs $Z_1 = 7338.571$, $Z_2 = 2848.670$. However, for the nonlinear membership function under GA, the results are $Z_1 = 2718.753$, $Z_2 = 1037.992$ with $\lambda = 0.99998$, $t = 0.5$ and $d = 0.1$, respectively. For Example 2, from Tables 3.6 and 3.7, it is concluded that the optimal solutions are reached only for the linear membership function under GA $Z_1 = 1045.983$, $Z_2 = 559.914$ with $\lambda = 0.977$ and $d = 4.4$. Again, for the production planning problem, from Tables 3.12 and 3.13, we have observed that the optimal results have been obtained for both levels $Z_1 = 12141.87$, $Z_2 = 1448.597$, $d = 4.4$, $\lambda = 0.999$ corresponding to the linear membership function, whereas $Z_1 = 3447.324$, $Z_2 = 306.866$, $d = 1.1$, $t = 0.2$, and $\lambda = 0.9567$ corresponding to the nonlinear membership function. So the best solution occurred corresponds to the linear membership function under GA. Finally, we have concluded that the three numerical experiments that attained the best solution correspond to only consider the linear membership function under GA.

3.7 Concluding remarks and future research

In this chapter we considered a TLPP for a noncooperative game where the cost parameters and the constraint parameters except on the right-hand side are either triangular or trapezoidal fuzzy numbers, whereas the constraints on the right side follow an extreme value distribution. Using Mellin transformation, fuzzy numbers are transformed into crisp value. Thereafter, the constraints in probabilistic nature have been converted into a crisp form by using the stochastic programming procedure. So it becomes a nonlinear programming problem. Then based on the computational algorithm, reduced problem is solved by LINGO 15.0 iterative scheme and GA, respectively and then compared the results.

The following aspects are treated as new features of our chapter.

- (a) For the first time, we have proposed a probabilistic TLPP in Stackelberg game under a fuzzy environment using linear and nonlinear membership functions.
- (b) Based on Mellin transformation, all types of fuzzy numbers (triangular or trapezoidal) are converted into crisp values.
- (c) Probabilistic constraints follow an extreme value distribution to be converted into a deterministic form using stochastic programming.
- (d) Comparative study about the results by LINGO 15.0 iterative scheme and GA correspond to nonlinear and linear memberships.

In future studies, this method can be considered in another type of nonlinear memberships such as hyperbolic, parabolic, and logarithm. This research work can be extended for multiobjective TLPP in noncooperative game, including managerial decision-making problem and transportation problem.

References

- [1] Anandalingam G. & Apprey V. (1991) Multi-level programming and conflict resolution, *European Journal of Operational Research*, 51(2), 233–247.
- [2] Maiti S.K. & Roy S.K. (2018) Bi-level programming for Stackelberg game with intuitionistic fuzzy number: a ranking approach, *Journal of the Operations Research Society of China*. <https://doi.org/10.1007/s40305-018-0234-2>.
- [3] Roy S.K. (2006) Fuzzy programming techniques for Stackelberg game, *Tamsui Oxford Journal of Management Sciences*, 22(3), 43–56.
- [4] Roy S.K. & Maiti S.K. (2017) Stochastic bi level programming with multi-choice for Stackelberg game via fuzzy programming, *International Journal of Operational Research*, 29(4), 508–530.
- [5] Roy S.K., Maity G., & Weber G.W. (2017) Multi-objective two-stage grey transportation problem using utility function with goals, *Central European Journal of Operations Research*, 25(2), 417–439.
- [6] Roy S.K. (2014) Multi-choice stochastic transportation problem involving Weibull distribution, *International Journal of Operational Research*, 21(1), 38–58.
- [7] Anandalingam G. & White D.J. (1990) A solution method for the linear static Stackelberg problem using penalty functions, *IEEE Transactions on Automatic Control*, 35(10), 1170–1173.
- [8] Hamidi F. & Mishmast N. (2013) Bi-level linear programming with fuzzy parameters, *Iranian Journal of Fuzzy Systems*, 10(4), 83–99.
- [9] Sakawa M., Katagiri H., & Matsui T. (2012) Fuzzy random bi-level linear programming through expectation optimization using possibility necessity, *International Journal of Machine Learning and Cybernetics*, 3(3), 183–192.
- [10] Singh V.P. & Chakraborty D. (2017) Solving bi-level programming problem with fuzzy random variable coefficients, *Journal of Intelligent and Fuzzy Systems*, 32(1), 521–528.
- [11] Yeh K., Whittaker C., Realffa M.J., & Lee J.H. (2015) Two stage stochastic bi-level programming model of a pre-established timber lands supply chain with biorefinery investment interests, *Computers and Chemical Engineering*, 73(2), 141–153.
- [12] Youness E.A., Emam O.E., & Hafez M.S. (2014) Fuzzy bi- level multi- objective fractional integer programming, *Applied Mathematics and Information Sciences*, 8(6), 2857–2863.
- [13] Zheng Y., Wana Z., & Wang G. (2011) A fuzzy interactive method for a class of bi-level multi-objective programming problem, *Expert Systems with Applications*, 38(8), 10384–10388.
- [14] Grzegorzewski P. (2002) Nearest interval approximation of a fuzzy number, *Fuzzy Sets and Systems*, 130(3), 321–330.
- [15] Debnath L. (1965) *Integral transforms and their applications*, CRC Press, New York.
- [16] Zimmermann H.J. (1978) Fuzzy programming and linear programming with several objective functions, *Fuzzy Sets and Systems*, 1(1), 45–55.
- [17] Roy S.K. (2007) Fuzzy programming approach to two person multi-criteria bi-matrix games, *The Journal of Fuzzy Mathematics*, 15(1), 141–153.
- [18] Zadeh L.A. (1965) Fuzzy sets, *Information and Control*, 8(3), 338–353.
- [19] Ren A.H. & Wang Y.P. (2014) Optimistic Stackelberg solutions to bi-level linear programming with fuzzy random variable coefficients, *Knowledge Based Systems*. <https://doi.org/10.1016/j.knosys.2014.05.010>.
- [20] Maiti S.K. & Roy S.K. (2016) Multi-choice stochastic bi-level programming problem in co-operative nature via fuzzy programming approach, *Journal of Industrial Engineering International*, 12(3), 287–298.
- [21] Lai Y.J. (1996) Hierarchical optimization: A satisfactory solution, *Fuzzy Sets and Systems*, 77(3), 321–335.

- [22] Mahapatra D.R., Roy S.K., & Biswal M.P. (2013) Multi-choice stochastic transportation problem involving extreme value distribution, *Applied Mathematical Modelling*, 37(4), 2230–2240.
- [23] Kiyota S. & Aiyishi E. (1981) A computational method for Stackelberg and min max problem use of a penalty method, *IEEE Transactions on Automatic Control*, 26(2), 460–466.
- [24] Sakawa M. & Matsui T. (2016) Bi-level linear programming with fuzzy random variables through absolute deviation minimization, *International Journal of Operational Research*, 25 (1), 1–27.
- [25] Sakawa M. & Yano H. (1989) An interactive fuzzy satisfying method of multi- objective nonlinear programming problems with fuzzy parameters, *Fuzzy Sets and Systems*, 30(3), 221–238.