



Journal of Engineering for Development, Vol. 13(1), October (2021) pp.38-49

**FUZZY (TRAPEZOIDAL) LINEAR PROGRAMMING PROBLEMS
A COMPUTERIZED SOLUTION**

¹Ebojoh, E., ²Okusuwha, J.E., ³Irabor, J.O., ⁴Nwanbunwanne, E.

¹Department of Production Engineering, University of Benin, P.M.B. 1154, Benin City,
Edo State, Nigeria.

¹**Email:** voke.ebojoh@uniben.edu

ABSTRACT

This paper considers a Rectangular (trapezoidal) fuzzy linear programming problem and develops computerized solutions to these problems. The computerized process which is known as the software was programmed using the Microsoft visual basic.Net programming language. The method used is the simplex method which is normally used for solving linear programming problems in crisp environment. In conducting the analysis, the rectangular fuzzy linear programming problems involving fully rectangular (trapezoidal) fuzzy linear programming problem, or just when its objective function variables are fuzzy, or when just its right-hand side constants are fuzzy were considered. Then the results of the manual and software solution were compared on the some bench mark questions. The software shows robustness and accurate.

Keywords: Objective Function, Variables, Constraints, Computer Aided Design (CAD), Graphic User Interface (GUI)

Article Information:

Received 25 August, 2021

Revised 13 September, 2021

Accepted 20 September, 2021

Available online 08 March, 2022

1. Introduction

Linear programming was first used in World War 2, during the war soldiers moved at the lowest cost possible while maximizing the number of soldiers and equipment. This posed a linear structure problem that required an optimized solution". In 1974, George Dantzig a member of the U.S. Air Force developed the simplex method of optimization in order to provide an efficient algorithm for solving programming problems having linear structures.

Fuzzy Linear Programming (FLP) problems allow working with imprecise data and constraints, leading to more realistic models. They have often been used for solving a wide variety of problems in sciences and engineering. Fuzzy set theory was originally introduced by Lotfi Zadeh in the 1960's. Zadeh (1965) proposed that it was specifically designed to mathematically represent uncertainty and vagueness and provide formalized tools for dealing with the imprecision intrinsic to many problems. The objective function characterizes the amount to be optimized, and the objective of linear programming is to obtain the values of the variables that maximize or minimize the objective function, while the constraint is an inequality that characterizes how the values of the variable in a problem are restricted. Vidhya (2017), propose that it assists with tackling some perplexing optimization problems like the travelling salesman problem, Cook et al (1997), also said that the problem of limiting this movement incorporates the Euclidean travelling salesman problem as an extraordinary case.

Fuzzy number is mainly a generalization of a regular, real number in the sense that it does not refer to a single value but rather to a connected set of possible values, where each possible value has its own weights, the weight is usually called the membership function Dijkman et al (1983); fuzzy numbers is therefore an extension of real numbers. The calculations of fuzzy numbers are implemented by using fuzzy arithmetic operations, fuzzy is an uncertain theory, it consists of grey systems, granular computing, which involves computing with words, decision making, and other sciences and engineering branches, the first authors on fuzzy arithmetic were Dubois and Prade.

Linear programming has important applications in these applications, since the real-world problems are very complex, the parameters of linear programming are usually represented by fuzzy numbers, there for many researchers has shown interest in this area of fuzzy linear programming. Recently fuzzy set theory has been applied in many research regions, since fuzzy set theory is effective to solve the decision-making problems with imprecise data. The idea of fuzzy mathematical programming on an overall level was first proposed by Tanaka et al, (1974), in the frame work of Bellman and Zadeh subsequently numerous authors,

Tanaka et al (2000) concentrated on getting fuzzy decision making problems utilizing probability circulations of fuzzy decision variables. The formulation of fuzzy linear programming was first presented by Zimmermann (1987). Rommelfanger (1996), has also proposed a general model for the fuzzy linear programming problems and the main difference compared with that of Delgado et al is that here parameters of the objective function are also fuzzy numbers. Mahdavi-Amiri et al (2006) develop a new dual algorithm for solving the fuzzy linear programming problem directly. They proposed a fuzzy linear programming model where a linear ranking function was used to rank trapezoidal fuzzy numbers. They established the

dual problem of the linear programming problem with trapezoidal fuzzy variables and deduced some duality results to solve the fuzzy linear programming problem directly with the primal simplex tableau. Ebojoh and Akpobi (2020) considered integer linear programming (ILP) and focused on all integer linear programming (AILP), Mixed Integer linear programming (MILP) and Fuzzy Integer linear Programming (FILP). Equations that served as guides for developing the software are presented in this work.

From the forgoing this research is to ease the computational digits and to avoid errors when solving when solving fully fuzzy linear trapezoidal problems, by developing a computer program that is user friendly to handle and display results accurately.

2.0 Design of Software Model

In designing the relevant algorithm for this software, standard design methodologies well treated in a number of texts were followed.

2.1 Definitions and Notations

Any linear programming model representing real world situations involves a lot of parameters whose values are assigned by experts. However, both experts and decision makers frequently do not precisely know the value of those parameters. Therefore, it is useful to consider the knowledge of experts about the parameters as fuzzy data introduced by Zadeh (1965)

2.1.1 Definition

If X is a collection of objects denoted generically by x , then fuzzy set A in X is a set of ordered pairs:

$$A = \{ \langle x, \mu_A(x) \rangle \mid x \in X \} \quad (1)$$

$\mu_A(x)$ is called the membership function or grade of membership (also degree of compatibility or degree of truth) of x in A that maps X to the membership space M (When M contains only the two points 0 and 1, A is non-fuzzy and $\mu_A(x)$ is identical to the characteristic function of a non-fuzzy set). The range of the membership function is a subset of the nonnegative real numbers whose supremum is finite. Elements with a zero degree of membership are normally not listed.

Definition 2.1.2

The support of a fuzzy set A , $S(1)$, is the crisp set of all $x \in X$ such that $\mu_A(x) > 0$.

Definition 2.1.3

The (crisp) set of elements that belong to the fuzzy set at least to the degree a is called the a -level set:

$$A_a = \{ x \in X \mid \mu_A(x) \geq a \} \quad (2)$$

$A'_a = \{ x \in X \mid \mu_A(x) > a \}$ is called "strong α -level set" or "strong α -cut."

Definition 2.1.4

A fuzzy set A is *convex* if $\mu_A(\lambda x_1 + (1 - \lambda) x_2) \geq \min \{ \mu_A(x_1), \mu_A(x_2) \}$, $x_1, x_2 \in X$, $\lambda \in [0, 1]$. Alternatively, a fuzzy set is convex if all a -level sets are convex.

Definition 2.1.5

For a finite fuzzy set A, the cardinality |A| is defined as:

$$|\tilde{A}| = \sum_{x \in X} \mu_{\tilde{A}}(x) \tag{3}$$

$\|\tilde{A}\| = \frac{|A|}{|X|}$ is called the relative cardinality of A.

Obviously, the relative cardinality of a fuzzy set depends on the cardinality of the universe. So, you have to choose the same universe if you want to compare fuzzy sets by their relative cardinality.

Definition 2.1.6

The fuzzy number A is a fuzzy set whose membership function $\mu_A x$ satisfies the following conditions

- a) $\mu_{\tilde{A}} x$ is a piecewise continuous
- b) $\mu_{\tilde{A}} x$ is a convex (4)
- c) $\mu_{\tilde{A}} x$ is normal (i.e.) $\mu_{\tilde{A}} x0 = 1$.

Definition 2.1.7 A fuzzy number \tilde{A} defined on the universal set of real numbers R, denoted as $\tilde{A} = (a, b, c, d)$, is said to be a trapezoidal fuzzy number if its membership function, $\mu_{\tilde{A}}(x)$ is given by:

$$\mu_{\tilde{A}}(x) = \begin{cases} \frac{(x-a)}{(b-a)} & a \leq x < b \\ 1 & b \leq x \leq c \\ \frac{(x-d)}{(c-d)} & c \leq x \leq d \\ 0 & otherwise \end{cases} \tag{5}$$

Definition 2.1.8 Let $\tilde{A} = (a, b, c, d)$ be a trapezoidal fuzzy number. Then, its λ -cut A^λ is defined as follows: $A^\lambda = [a + (b - a)\lambda, d - (d - c)\lambda], 0 \leq \lambda \leq 1$ (6)

Definition 2.1.9 Trapezoidal fuzzy numbers: There are various types of fuzzy numbers, but the triangular and trapezoidal are the most important fuzzy memberships. In this research we use the trapezoidal fuzzy numbers. In fact, the fuzzy number is defined by its corresponding membership function as shown in Figure 1. The trapezoidal membership function for fuzzy number \tilde{A} is as follows: A fuzzy number with membership function of the form below is called a Trapezoidal fuzzy number $\tilde{A} = (a, b, c, d)$

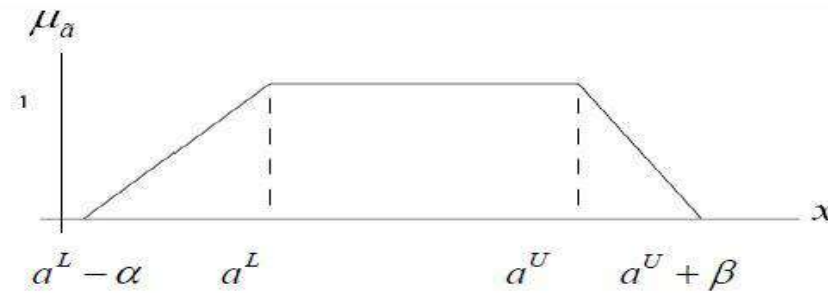


Figure 1: Trapezoidal fuzzy numbers function

The technique used for solving fully fuzzy linear programming for this study is the simplex method.

2.2 The Simplex Method

In solving linear programming with fuzzy number, the simplex method can be used. The Revised Tsao method is one of the most efficient methods for ranking fuzzy numbers and solving such problems. In his paper solving fuzzy linear programming by using revised tsao method, Nasser and Sohrabi (2010), suggested using this method in the fuzzy primal simplex where we need to evaluate the optimality condition in the iteration. The mentioned approach will be useful when one have to use the fuzzy dual simplex too Nasser et al (2005).

2.2.1 Converting the Constraints to Equation

The first step of the simplex method requires each inequality constraint in an LP formulation is converted into an equation. Less-than-or-equal-to constraints (\leq) can be converted to equations by adding slack variables, which represent the amount of an unused resource.

2.2.2 Setting up the First Simplex Tableau

To simplify handling the equations and objective function in an LP problem, all of the coefficients put into a tabular form. Variables in the solution mix, which is often called the **basis** in LP terminology, are referred to as **basic variables**. In this example, the basic variables are S_1 and S_2 . Variables not in the solution mix or basis (x_1 and x_2 , in this case) are called **non basic variables**.

2.2.3 Simplex Solution Procedures

Once initial tableau is completed, a series of five steps are followed to compute all of the numbers needed for the next tableau. The calculations are not difficult, but they are sufficiently complex that the smallest arithmetic error can produce a very wrong answer.

1. Determine which variable to enter into the solution mix next. Identify the column hence the variable with the largest positive number in the $C_j - Z_j$ row of the previous tableau. This step means that we will now be producing some of the product contributing the greatest additional profit per unit.
2. Determine which variable to replace. Because we have just chosen a new variable to enter into the solution mix, we must decide which variable currently in the solution to remove to make room for it. To do so, we divide each amount in the quantity column by the corresponding number in the column selected in step 1. The row with the smallest

nonnegative number calculated in this fashion will be replaced in the next tableau (this smallest number, by the way, gives the maximum number of units of the variable that we may place in the solution). This row is often referred to as the pivot row, and the column identified in step 1 is called the pivot column. The number at the intersection of the pivot row and pivot column is the pivot number.

3. Compute new values for the pivot row. To find them, we simply divide every number in the row by the pivot number.
4. Compute new values for each remaining row. (In our sample problems there have been only two rows in the LP tableau, but larger problems have many more rows.) All remaining row(s) are calculated as follows:
(New row numbers) = (numbers in old rows) – [(number in old row above or below X (corresponding number in the new row. the row replaced in step 3)]
5. Compute the Z_j and $C_j - Z_j$ rows, as demonstrated in the initial tableau. If all numbers in the $C_j - Z_j$ row are zero or negative, we have found an optimal solution. If this is not the case, we must return to **step 1**.

3.0 Writing the Programme Code

The program code is written in line with the following algorithm.

Start

Enter number of constraints and number of variables

Select type of fuzzy problem

Problem type 1: Only objective function as fuzzy numbers

Problem type 2: Only RHS as fuzzy numbers

Problem type 3: Fully fuzzy

Load and create input tables, column width based on problem type

Input known parameters

Solve

If problem type == Type 1 (Only objective function as fuzzy numbers) Then

Determine the rank of each objective function fuzzy number

Obtain the first tableau

Check for optimality

If result is optimal Then

Display results

Else If result is not optimal Then

If cycles count > 100 Then

End procedure

Else

Compute new tableau using simplex method and fuzzy arithmetic

goto: Check for optimality

End if

End if

End if

If problem type == Only RHS as fuzzy numbers Then

Determine the rank of each RHS fuzzy number

```
Obtain the first tableau
Check for optimality
If result is optimal Then
    Display results
Else If result is not optimal Then
    If cycles count > 100 Then
        End procedure
    Else
        Compute new tableau using simplex method and fuzzy arithmetic
        goto: Check for optimality
    End if
End if
End if
If problem type == fully fuzzy Then
    Determine the rank of each of the fuzzy numbers
    Obtain the first tableau
    Check for optimality
    If result is optimal Then
        Display results
    Else If result is not optimal Then

        If cycles count > 100 Then
            End procedure
        Else
            Compute new tableau using simplex method and fuzzy arithmetic
            goto: Check for optimality
        End if
    End if
End if
```

4.0 Flow Chart

The schematic representation of the flowchart is shown in figure 2.

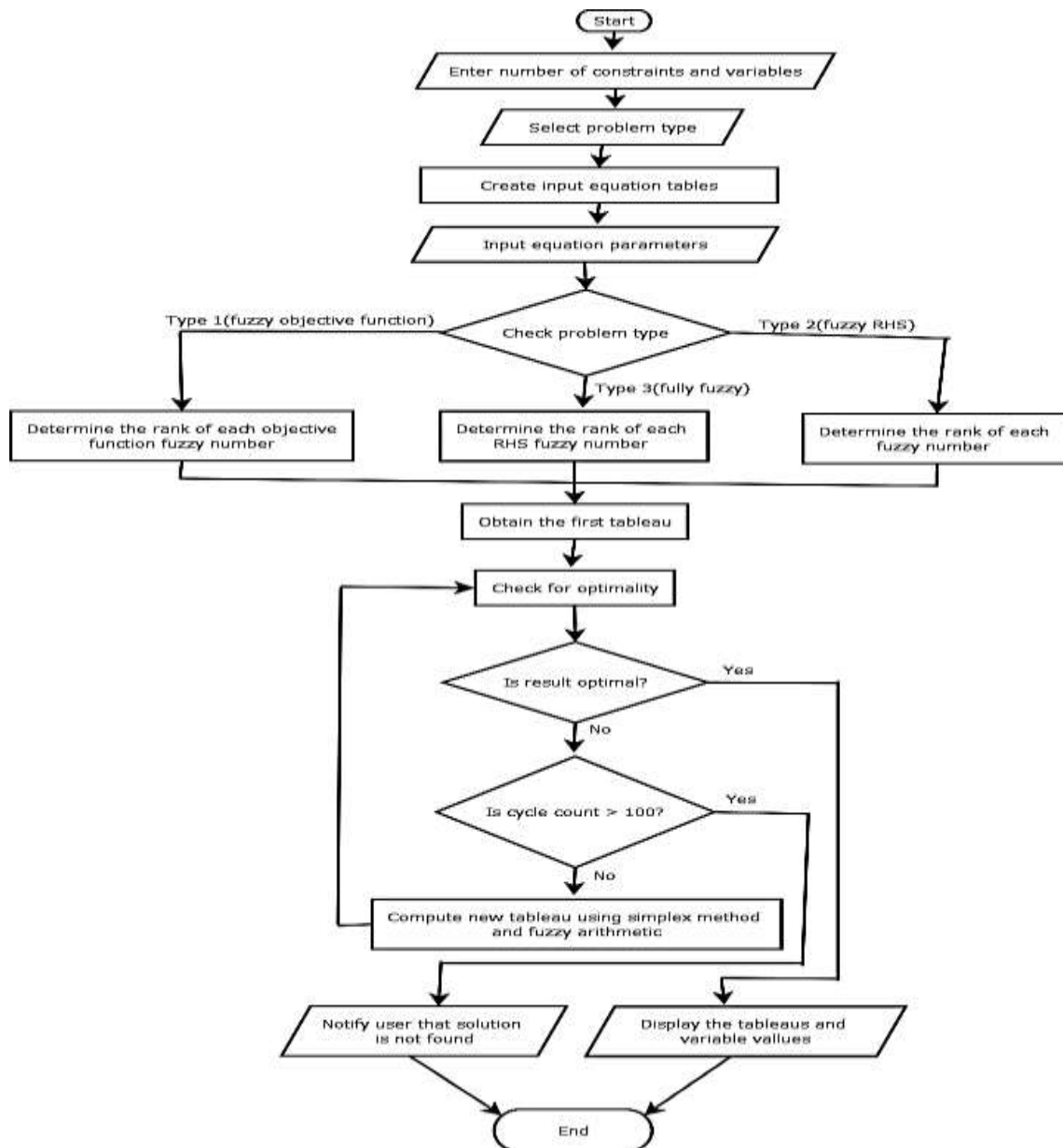


Figure 2: Flow chart for Software Development

5.0 Results and Discussion

To test the efficiency, accuracy and robustness of the software, the following benchmark problems were solved using the software.

Example 1:

$$\begin{aligned} \text{Max } Z &= [5, 8, 2, 5]X_1 + [6, 10, 2, 6]X_2 \\ \text{St } 2X_1 + 3X_2 &\leq 6 \\ 5X_1 + 4X_2 &\leq 10 \\ X_1, X_2 &\geq 0 \end{aligned}$$

Solution

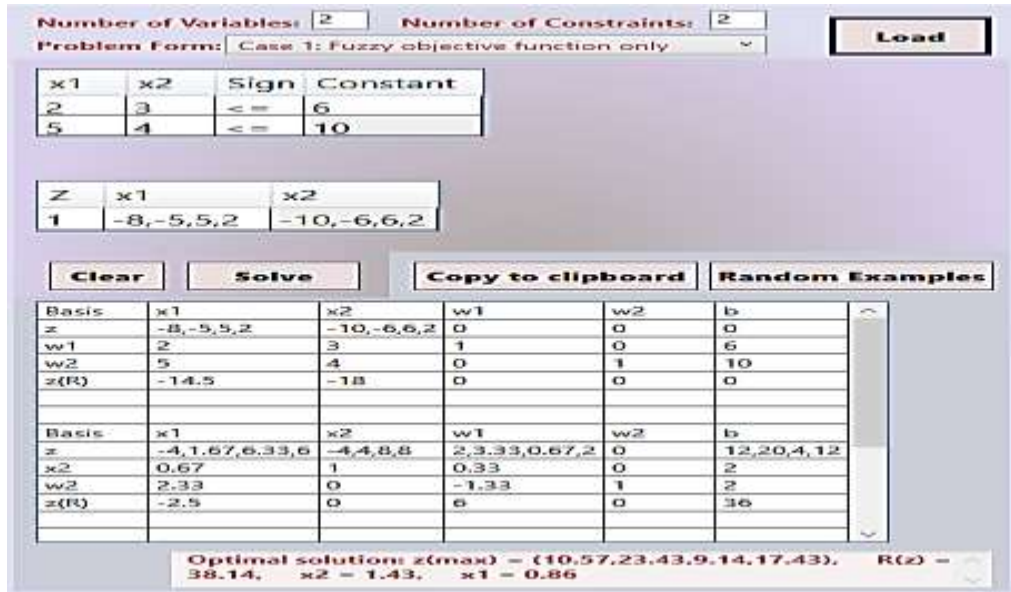


Figure 3: Snapshot of Answer

Example 2:

$$\begin{aligned} \text{Max } Z &= [4, 5, 6, 7]X_1 + [8, 9, 10, 11]X_2 \\ \text{St } [2, 3, 4, 5]X_1 + [1, 2, 3, 4]X_2 &\leq [12, 15, 18, 19] \\ [4, 8, 9, 10]X_1 + [3, 5, 7, 9]X_2 &\leq [20, 25, 28, 30] \\ X_1, X_2 &\geq 0 \end{aligned}$$

Solution

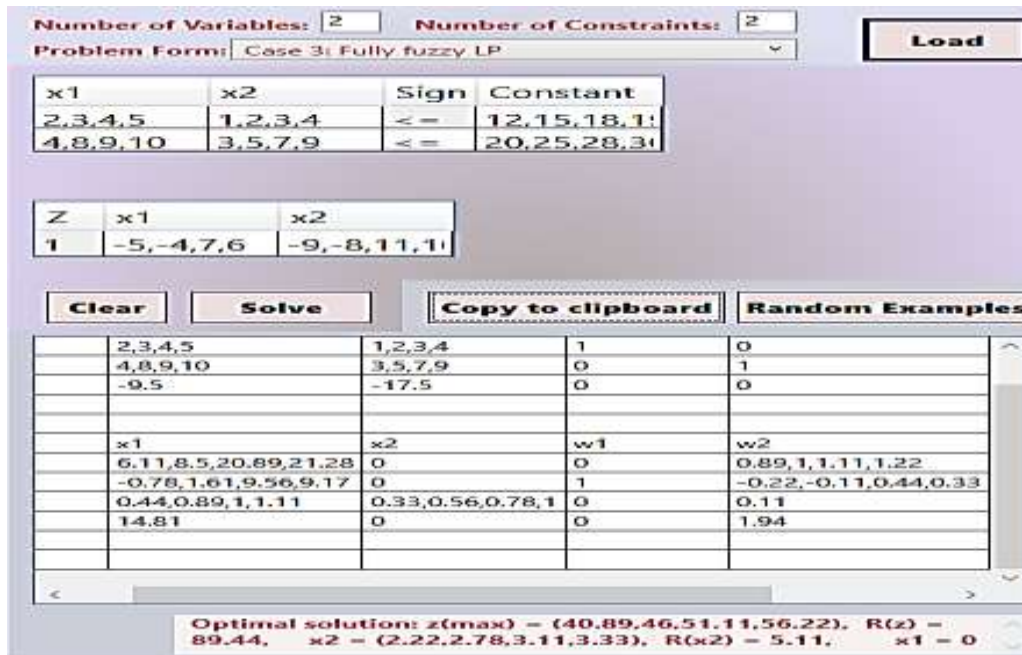


Figure 4: Snapshot of Answer

Example 3:

$$\begin{aligned} \text{Max } Z &= (4, 6, 2, 3)X_1 + (5, 8, 2, 4)X_2 \\ \text{St } 3X_1 + 6X_2 &\leq 18 \\ 5X_1 + 4X_2 &\leq 20 \\ X_1, X_2 &\geq 0 \end{aligned}$$

Solution:

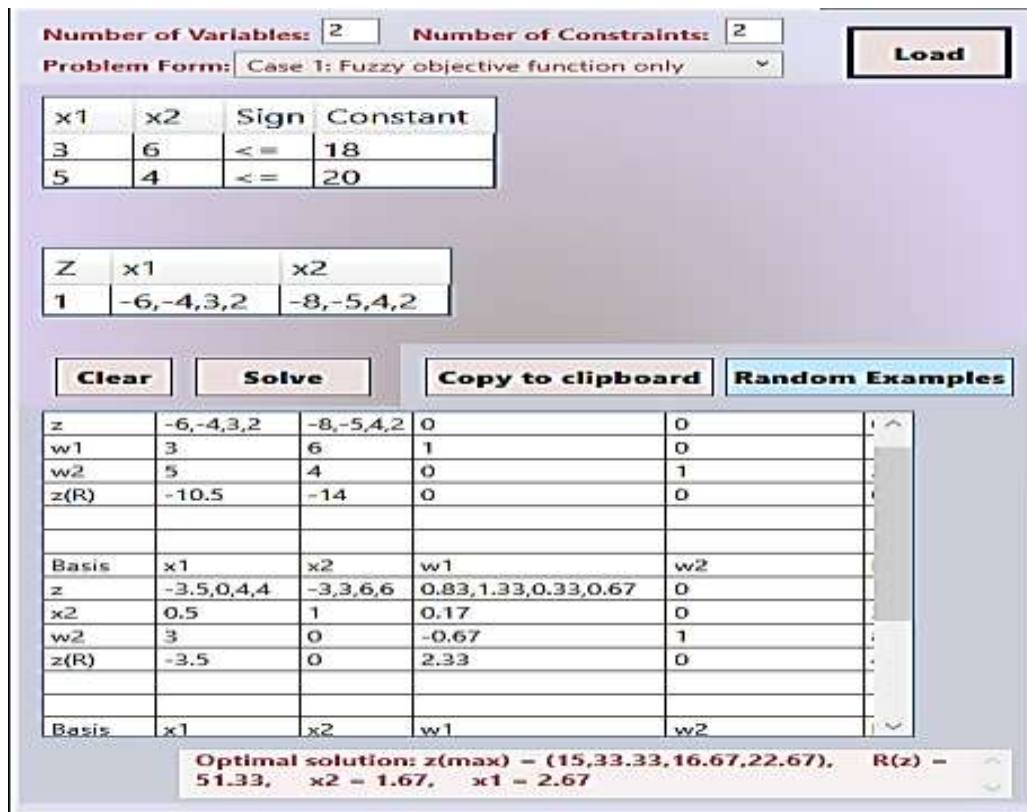


Figure 5: Snapshot of Answer

Example 4:

$$\begin{aligned} \text{Max } Z &= 3X_1 + 4X_2 \\ \text{St } 3X_1 + X_2 &\leq (2,4,1,3) \\ 2X_1 - 3X_2 &\leq (3,5,2,1) \\ X_1, X_2 &\geq 0 \end{aligned}$$

Example 5:

$$\begin{aligned} \text{Max } Z &= (5, -4, 1, 2)X_1 + (6, -7, 3, -4)X_2 + (4, -5, -3, 2)X_3 \\ &+ (-7, -5, 4, 3)X_4 + (4, 3, 2, -1)X_5 \\ \text{St } 4X_1 + 5X_2 + X_3 + 2X_4 + 3X_5 &\leq 35 \\ 2X_1 + 15X_2 - 2X_3 + X_4 - 4X_5 &\leq 23 \\ 2X_1 - 2X_2 + 15X_3 + 2X_4 - X_5 &\leq 28 \\ X_1, X_2, X_3, X_4, X_5 &\geq 0 \end{aligned}$$

The software functionality was tried on a number of practical problems already solved as shown in figure 3 to figure 5 above and the solution obtained both from manual computation and software computation was carefully entered in the Table 1, having tried the software on a number of problem, the software accuracy is evident in the consistency of results with that obtained by manual computation available in established texts, we were able to ascertain the percentage proximity in terms of accuracy of the software computation to the manual computation evaluated as percentage error. The percentage error was obtained as shown in Table 1.

Table 1: Showing Results Obtained by Manual and Software Computation

S/N	TYPE PROBLEM	TYPE	SOFTWARE COMPUTATION	MANUAL COMUPUTATION	%ERROR
1	Fuzzy Objective Function only	Max	$Z_{\max} = [10.57, 23.43, 9.14, 17.43]$ $Z^{\circledast} = 38.14$ $X_2 = 1.43$ $X_1 = 0.86$	$Z_{\max} = [\frac{74}{7}, \frac{164}{7}, \frac{64}{7}, \frac{122}{7}]$ $Z^{\circledast} = \frac{267}{7}$ $X_2 = \frac{10}{7}$ $X_1 = \frac{6}{7}$	0.00
2	Fully Fuzzy	Max	$Z_{\max} = [40.89, 46, 51.11, 56.22]$ $Z^{\circledast} = 89.44$ $X_2 = [2.22, 2.78, 3.11, 3.33]$ $X_2^{\circledast} = 5.1$ $X_1 = 0$	$Z_{\max} = [\frac{368}{9}, \frac{414}{9}, \frac{460}{9}, \frac{506}{9}]$ $Z^{\circledast} = 89.44$ $X_2 = [\frac{20}{9}, \frac{25}{9}, \frac{28}{9}, \frac{30}{9}]$ $X_2^{\circledast} = 5.1$ $X_1 = 0$	0.000
3	Fuzzy Objective Function only	Max	$Z_{\max} = [15, 33.33, 16.67, 22.67]$ $Z^{\circledast} = 51.33$ $X_2 = 1.67$ $X_1 = 2.67$	$Z_{\max} = [15, \frac{100}{3}, \frac{50}{3}, \frac{68}{3}]$ $Z^{\circledast} = \frac{154}{3}$ $X_2 = \frac{5}{3}$ $X_1 = \frac{8}{3}$	0.000
4	Fuzzy RHS constant only	Max	$Z_{\max} = [16, 8, -12, -4]$ $Z^{\circledast} = 28$ $X_2 = [2, 4, 1, 3]$ $X_2^{\circledast} = 7$ $X_1 = 0$	$Z_{\max} = [16, 8, -12, -4]$ $Z^{\circledast} = 28$ $X_2 = [2, 4, 1, 3]$ $X_2^{\circledast} = 7$ $X_1 = 0$	0.000
5	Fuzzy Objective Function only	Max	$Z_{\max} = [78.03, 98.3, 35.84, 57.25]$ $Z^{\circledast} = 187.04$ $X_5 = 0.55$ $X_4 = 14.96$ $X_3 = 0$ $X_2 = 0.68$ $X_1 = 0$	$Z_{\max} = [78.03, 98.3, 35.84, 57.25]$ $Z^{\circledast} = 187.04$ $X_5 = 0.55$ $X_4 = 14.96$ $X_3 = 0$ $X_2 = 0.68$ $X_1 = 0$	0.000

6.0 Conclusion

A software that can solve and give the solution to problems relating to trapezoidal fuzzy numbers associated with simplex method has been developed. From the results for both manual and the software, calculations are in good agreement. Knowing fully well that manual

computation of problems would require some pretty time, the software was designed to solve speedily, so as to easily interpret the solutions to the problem since it is user friendly.

References

- Cook, W. J., Cunningham, W. H., Pulleyblank, W. R. furthermore, Schrijver, A. (1997). Combinatorial Optimization. Wiley Interscience, New York.
- Dijkman, J.G, Haeringen, H van, Lange, S.J de (1983). “Fuzzy numbers”. Journal of mathematics analysis and applications. 92 (2) 301- 341.
- Ebojoh E. and Akpobi J.A (2020), Integer Linear Programming an Object Oriented Approach. Journal of the Nigerian Association of Mathematical Physics, vol. 54, pp129-136
- Examination Vidhya, Introductory guide on straight programming for (hopeful) information researchers, February 28, 2017.
- Faculty of Computer Science and Information Systems, West Pomeranian University of Technology, Żołnierska 49, 71-210 Szczecin, Poland.
- Mahdavi-Amiri, N., Nasseri, S.H. (2006), Duality in fuzzy number linear programming by the use of a certain linear ranking function. Appl. Math. Comput. 180, 206–216.
- Rommelfanger, H. (1996), Fuzzy linear programming and applications. Eur. J. Oper. Res. 92, 512– 527.
- Tanaka, H., Guo, P., Zimmermann, H.J. (2000), Possibility distributions of fuzzy decision variables obtained from possibilistic linear programming problems. Fuzzy Sets Syst. 113, 323– 332.
- Zadeh, L.A. (1965), Fuzzy sets. Inf. Control 8, 338–353.
- Zimmermann, H.J. (1987), Fuzzy Sets. Decision Making and Expert Systems. Kluwer Academic Publishers, MA.
- Nasseri, S.H. & Sohrabi, M. (2010), “Solving fuzzy Linear Programming by using Revised Tsao’s Method”. Australian Journal of Basic and Applied Sciences, 4(10): pp. 4592 – 4600 (2010). ISSN 1991 – 8178
- Nasseri, S.H., Ardil, E. Y. A., Zaefarian, R. (2005), “Simplex Method for solving Linear Programming Problems with Fuzzy Numbers”. World academy of science, Engineering and Technology, pp. 10 (2005)