7 Fully Fuzzified Linear Programming I

7.1 Introduction

We first discuss the general fully fuzzified linear program in the next section. Then we study a max problem, the product mix problem, in Section 7.3. We have previously obtained an approximate fuzzy solution to this problem using an evolutionary algorithm ([1],[2]). In Section 7.4 we will apply our fuzzy Monte Carlo method to the problem to generate another approximate solution and then compare these new results to the evolutionary algorithm method.

7.2 Fully Fuzzified Linear Programming

Fuzzy linear programming has long been an area of application of fuzzy sets. Consider the classical linear program

$$\begin{aligned} & \max / \min Z = c_1 x_1 + \cdots + c_n x_n \\ \text{subject to:} & \\ & a_{i1} x_1 + \cdots + a_{in} x_n \le b_i, \quad 1 \le i \le m \\ & x_i \ge 0, \text{for all } i. \end{aligned} \tag{7.1}$$

We need to have values for all the parameters c_i, a_{ij} and b_i to completely specify the optimization problem. Many of these must be estimated and are therefore uncertain. It is then natural to model these uncertain parameters using fuzzy numbers. The problem then becomes a fuzzy linear programming problem.

We are going to allow all the parameters to be fuzzy and we obtain what we have called the fully fuzzified linear programming problem. Many researchers (see the references to Chapter 6) have looked at parts of this problem: (1) the a_{ij} and b_i can be fuzzy; or (but not both) (2) the c_i can be fuzzy. The fully fuzzified (max) linear program is

J.J. Buckley et al.: Monte Carlo Meth. in Fuzzy Optimization, STUDFUZZ 222, pp. 67–73, 2008.
springerlink.com

$$\begin{aligned} max\overline{Z} &= \overline{C}_1\overline{X}_1 + \cdots + \overline{C}_n\overline{X}_n \\ &\text{subject to:} \\ \overline{A}_{i1}\overline{X}_1 + \cdots + \overline{A}_{in}\overline{X}_n &\leq \overline{B}_i,\ 1 \leq i \leq m, \\ \overline{X}_i &\geq 0, \text{for all i.} \end{aligned} \tag{7.2}$$

where the $\overline{C}_i$, $\overline{A}_{ij}$ and $\overline{B}_i$ can all be triangular fuzzy numbers. Not every single parameter need be fuzzy; but we shall assume that some of the $\overline{C}_i$ are fuzzy and some of the $\overline{A}_{ij}$ and $\overline{B}_i$ are fuzzy. Since the parameters are fuzzy, the variables $\overline{X}_i$ will be triangular shaped fuzzy numbers.

We are now concerned with solving the optimization problem in equation (7.2). But first we must do two things: (1) explain what we mean by max/min $\overline{Z}$ since $\overline{Z}$ will also be a triangular shaped fuzzy number; and (2) decide on how we will evaluate the inequality ($\leq$) between fuzzy numbers $\overline{E}_i \leq \overline{B}_i$, where $\overline{E}_i = \overline{A}_{i1}\ \overline{X}_1 + \cdots + \overline{A}_{in}\ \overline{X}_n$.

In our previous research on this topic we handled max/min $\overline{Z}$ as discussed in Section 2.5 of Chapter 2. Also, in those publications we used both Kerre's Method (Section 2.6.2) and Chen's Method (Section 2.6.3) to evaluate $\leq$ between fuzzy numbers. In our fuzzy Monte Carlo method we will use both Kerre's Method and Chen's Method to evaluate $\leq$ and $<$ between fuzzy numbers.

7.3 Product Mix Problem

A company produces three products P_1, P_2 and P_3 each of which must be processed through three departments D_1, D_2 and D_3. The approximate time, in hours, each P_i spends in each D_j is given in Table 7.1.

Table 7.1. Approximate Times Product P_i is in Department D_j

		Department		
		D_1	D_2	D_3
Product	P_1	6	12	2
	P_2	8	8	4
	P_3	3	6	1

Each department has only so much time available each week. These times can vary slightly from week to week so the following numbers are estimates of the maximum time available per week, in hours, for each department: (1) for D_1 288 hours; (2) 312 hours for D_2; and (3) D_3 has 124 hours. Finally, the selling price for each product can vary a little due to small discounts to certain customers but we have the following average selling prices: (1) \$6 per unit for P_1; (2) \$8 per unit for P_2, and (3) \$6 per unit for P_3. The company wants to determine the number of units to produce for each product per week to maximize its revenue.

Since all the numbers given are uncertain, we will model the problem as a fully fuzzified linear program. We substitute a triangular fuzzy number for each value where the peak of the fuzzy number is at the number given. So, we have the following fully fuzzified linear program

$$\max \overline{Z} = (5.8/6/6.2)\overline{X}_1 + (7.5/8/8.5)\overline{X}_2 + (5.6/6/6.4)\overline{X}_3 \tag{7.3}$$

$$\text{subject to:} \tag{7.4}$$

$$(5.6/6/6.4)\overline{X}_1 + (7.5/8/8.5)\overline{X}_2 + (2.8/3/3.2)\overline{X}_3 \leq (283/288/293),$$
$$(11.4/12/12.6)\overline{X}_1 + (7.6/8/8.4)\overline{X}_2 + (5.7/6/6.3)\overline{X}_3 \leq (306/312/318),$$
$$(1.8/2/2.2)\overline{X}_1 + (3.8/4/4.2)\overline{X}_2 + (0.9/1/1.1)\overline{X}_3 \leq (121/124/127),$$
$$\overline{X}_1, \overline{X}_2, \overline{X}_3 \geq 0,$$

where the $\overline{X}_i$ are triangular shaped fuzzy numbers for the amount to produce for P_i per week.

7.4 Fuzzy Monte Carlo Method

We need to find intervals $I_i = [0, M_i]$, $i = 1, 2, 3$, as explained in Section 6.3 in Chapter 6, for the $\overline{X}_i$. Since we form fuzzy vector $\overline{V} = (\overline{X}_1, \overline{X}_2, \overline{X}_3)$, and each $\overline{X}_i$ consumes 5 crisp numbers, we choose our stream of Sobol quasi-random numbers which had been generated 15 at a time to get three sets of 5. We randomly generate $\overline{X}_i \in [0, M_i]$, $i = 1, 2, 3$, and form the random fuzzy vector $\overline{V} = (\overline{X}_1, \overline{X}_2, \overline{X}_3)$. We test to see if $\overline{V}$ is feasible, or the $\overline{X}_i$ satisfy the constraints. Assuming that $\overline{V}$ is feasible we compute $\overline{Z}_0 = \overline{C}_1\overline{X}_1 + ... + \overline{C}_3\overline{X}_3$. If $\overline{Z}^*$ is the current best (max) value of $\overline{Z}$ then we replace $\overline{Z}^*$ with $\overline{Z}_0$ if $\overline{Z}^* < \overline{Z}_0$, otherwise we discard $\overline{Z}_0$.

We also need to solve the fuzzy max problem twice. First we use Kerre's Method to evaluate $\overline{E}_i \leq \overline{B}_i$ in the constraints and $\overline{Z}^* < \overline{Z}_0$ in the objective function. Then we use Chen's Method.

To get an idea for the intervals I_i for the $\overline{X}_i$, $i = 1, 2, 3$, we studied the solutions to this problem from the evolutionary algorithm, and we studied our constraint equations. Since our constraint equations are '$\leq$' inequalities, we may set any two of our $\overline{X}_i$ to be zero to determine the maximum possible value for the third fuzzy variable. Then we use the minimum of these maximums to determine a possible support interval for feasible sets. For example, set $\overline{X}_2 = \overline{X}_3 = 0$, and let $\overline{X}_1 \approx (x_{11}/x_{12}/x_{13})$. After multiplying by the fuzzy coefficients we solve each $6x_{12} \leq 293$, $12x_{12} \leq 318$, $2x_{12} \leq 127$ for x_{12} and take the minimum. The result is 26.5 and we take the interval $[0, 26.5]$ for $\overline{X}_1$. Similarly we obtain $[0, 31.8]$ for $\overline{X}_2$ and $[0, 53]$ for $\overline{X}_3$.

Now we follow the procedure outlined in Sections 6.3.1 and 6.3.2.

7.4.1 Kerre's Method

We randomly generate vectors $\overline{V} = (\overline{X}_1, \overline{X}_2, \overline{X}_3)$, where the $\overline{X}_{ik}$ are Bézier (quadratic) fuzzy numbers (QBGFNs in Chapter 4), and check to see if they

Table 7.2. Monte Carlo Solution to the Fuzzy Linear Program, Kerre's $\leq$, N=100,000

$max\overline{Z}$	$\overline{X}_i$ (QBGFNs from Sobol)
$\approx (168.02/260.40/591.80)$	$\overline{X}_1 = (0.10, 2.75, 12.41, 0.57, -10.55)$
	$\overline{X}_2 = (22.21, 24.30, 26.98, 0.92, 0.80)$
	$\overline{X}_3 = (0.15, 8.25, 44.61, -5.30, -31.23)$

satisfy constraint equations (7.4) using Kerre's $\leq$. We wish to solve the optimization problem given in equation (7.3). If these equations are satisfied, then $\overline{V}$ is feasible and we evaluate $\overline{Z}$, the fuzzy objective function in equation (7.3). Let the previous best (max) value of $\overline{Z}$ be $\overline{Z}^*$ and the current value of $\overline{Z} = \overline{Z}_0$ from the recent feasible $\overline{V}$. If $\overline{Z}^* < \overline{Z}_0$ using Kerre's $<$, then set $\overline{Z}^*$ to be $\overline{Z}_0$, otherwise discard $\overline{Z}_0$ and generate the next random $\overline{V}$.

With $N = 100,000$ the results of the fuzzy Monte Carlo method are shown in Table 7.2, and Figures 7.1 & 7.2. All the fuzzy numbers in Table 7.2 are triangular shaped fuzzy numbers. $\overline{X}_1$, $\overline{X}_2$ and $\overline{X}_3$ are QBGFNs. The notation we use for these fuzzy numbers was explained in Section 4.3.2. and also in Section 6.3.1. Since $max\overline{Z}$ is not necessarily or likely to be a QBGFN, we only give the support and core for $max\overline{Z}$. Our approximate solution to this fuzzy linear program are the fuzzy numbers determined by this Monte Carlo program.

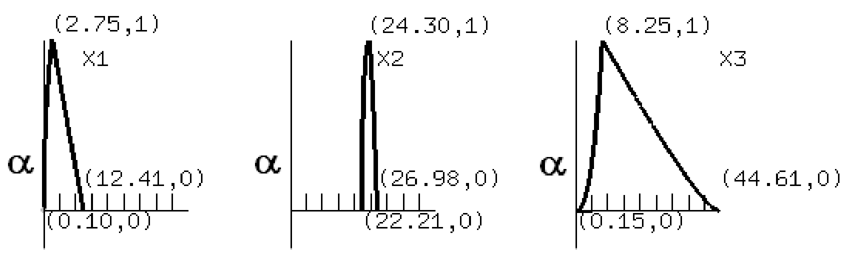

Fig. 7.1. $\overline{X}_1, \overline{X}_2, \overline{X}_3$ Solution using Kerre's $\leq$, Product Mix Problem

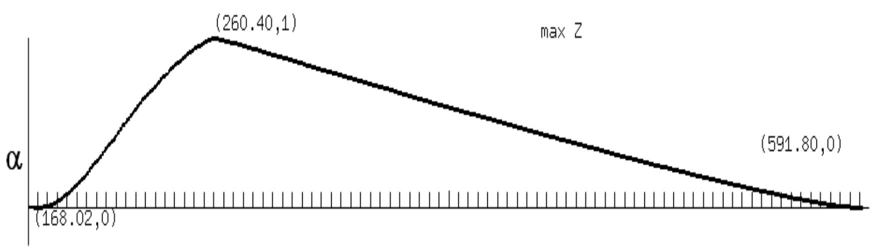

Fig. 7.2. $max\overline{Z}$ using Kerre's $\leq$, Product Mix Problem

Figure 7.1 show the "optimal" $\overline{X}_i$. Figure 7.2 shows the value of the objective function $\overline{Z}$, corresponding to the values of the $\overline{X}_i$ given in Figure 7.1.

7.4.2 Chen's Method

Now we randomly generate vectors $\overline{V} = (\overline{X}_1, \overline{X}_2, \overline{X}_3)$, where the $\overline{X}_{ik}$ are Bézier (quadratic) fuzzy numbers (QBGFNs in Chapter 4), and check to see if they satisfy constraint equations (7.4) using Chen's $\leq$. We wish to solve the optimization problem given in equation (7.3). If these equations are satisfied, then $\overline{V}$ is feasible and we evaluate $\overline{Z}$, the fuzzy objective function in equation (7.3). Let the previous best (max) value of $\overline{Z}$ be $\overline{Z}^*$ and the current value of $\overline{Z} = \overline{Z}_0$ from the recent feasible $\overline{V}$. If $\overline{Z}^* < \overline{Z}_0$ using Chen's $<$, then set $\overline{Z}^*$ to be $\overline{Z}_0$, otherwise discard $\overline{Z}_0$ and generate the next random $\overline{V}$.

With $N = 100,000$ the results of the fuzzy Monte Carlo method are shown in Table 7.3, and Figures 7.3 & 7.4. All the fuzzy numbers in Table 7.3 are triangular shaped fuzzy numbers. $\overline{X}_1$, $\overline{X}_2$ and $\overline{X}_3$ are QBGFNs. The notation we use for these fuzzy numbers was explained in Section 4.3.2. and also in Section 6.3.1. Since $max\overline{Z}$ is not necessarily or likely to be a QBGFN, we only give the support and core for $max\overline{Z}$. Our approximate solution to this fuzzy linear program are the fuzzy numbers determined by this Monte Carlo program.

Table 7.3. Monte Carlo Solution to the Fuzzy Linear Program, Chen's $\leq$, N=100,000

$max\overline{Z}$	$\overline{X}_i$ (QBGFN from Sobol)
$\approx (285.30/286.20/287.10)$	$\overline{X}_1 = (0.00, 0.00, 0.00, -1.00, -0.00)$
	$\overline{X}_2 = (15.90, 15.90, 15.90, -0.00, -0.00)$
	$\overline{X}_3 = (26.50, 26.50, 26.50, -0.00, -0.00)$

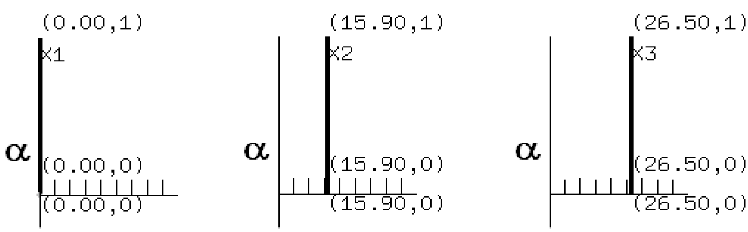

Fig. 7.3. $\overline{X}_1, \overline{X}_2, \overline{X}_3$ Solution using Chen's $\leq$, Product Mix Problem

Figure 7.3 show the "optimal" $\overline{X}_i$. Please note that this is not a crisp solution. They appear to be crisp when we round the results to two decimal places. Each left and right support is not coincident with its vertex. Figure 7.4 shows the value of the objective function $\overline{Z}$, corresponding to the values of the $\overline{X}_i$ given in Figure 7.3.

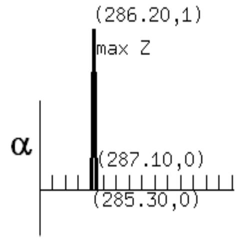

Fig. 7.4. $max\overline{Z}$ using Chen's $\leq$, Product Mix Problem

7.5 Comparison of Solutions

For another comparison the solution to the crisp problem is $x_1 = 0$, $x_2 = 27$, $x_3 = 16$ with max $z = 312$. The crisp linear program is the one obtained using the core values of all the fuzzy numbers.

All of these software efforts were performed on Windows-based PCs. These two fuzzy Monte Carlo optimizations were executed on a Lenovo T60 (T2400), 1.83GHz, 1.49GB RAM.

The Kerre comparison method found $8,749$ feasible sets in a stream of 100,000 ($\overline{X}_{1k}$,$\overline{X}_{2k}$,$\overline{X}_{3k}$); 38 of them had triggered new maximums. Elapsed time for the run was 14:38:08; the last maximum was found nearly at the end of the stream ($97,018^{th}$ vector triplet).

The Chen comparison method found $4,987$ feasible sets in the same stream of 100,000 ($\overline{X}_{1k}$,$\overline{X}_{2k}$,$\overline{X}_{3k}$); 7 of them had triggered new maximums. Elapsed time for the run was 14:14:35, but the last maximum was found after 04:44:17 at vector triplet $33,335$. The Chen solution was found with the $1,687^{th}$ feasible set. The solution using Chen's comparison method is not too fuzzy and is strangely not like the crisp solution reported above. We review the crisp solution with respect to the fuzzy constraint equations (7.4) and find that the second constraint equation evaluates to (311.3/312/312.7) which does not satisfy the constraint by either Kerre's or Chen's $\leq$. We note that a crisp solution to a crisp linear programming problem might not satisfy a fuzzy linear programming problem.

The two methods reported two identical new maximums. At the 492^{nd} vector triplet they reported a new maximum of $\approx$(155.06/275.57/346.00). Then again at vector triplet $33,335$ they found the same new maximum. Chen's method found no more new maximums.

For our Monte Carlo solution using Sobol quasi-random numbers, we compare Kerre's method results and Chen's method results to find that the $max\overline{Z}$ from Kerre's method is less than the $max\overline{Z}$ from Chen's method solution if Buckley's method or Kerre's method is used to compare the maximums. If Chen's method is used to compare the maximums, the $max\overline{Z}$ from Chen's method is less than the $max\overline{Z}$ from Kerre's method solution.

Yet another interesting solution is that given in [1], obtained by using an Evolutionary Algorithm to arrive at a solution. The results of that Evolutionary Algorithm method, using Kerre's method, are shown in Figures 7.5 and 7.6.

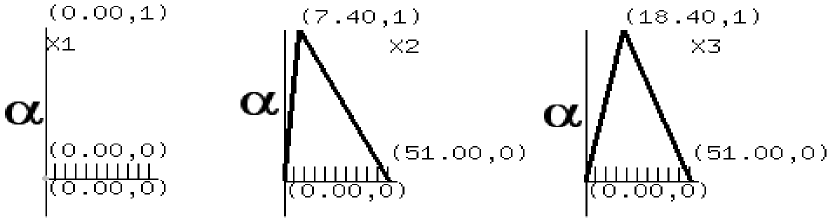

Fig. 7.5. Evolutionary Algorithm, $\overline{X}_1, \overline{X}_2, \overline{X}_3$ using Kerre's $\leq$ Product Mix Problem

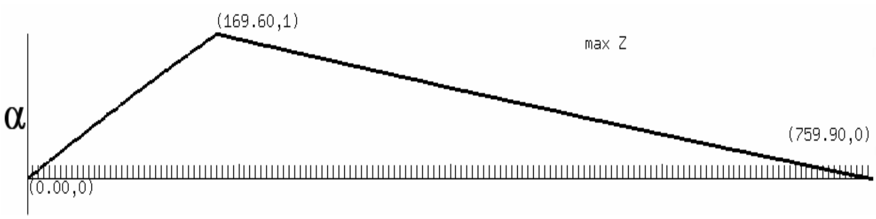

Fig. 7.6. Evolutionary Algorithm, $max\overline{Z}$ using Kerre's $\leq$, Product Mix Problem

All the fuzzy numbers in Figure 7.5 are triangular fuzzy numbers but $max\overline{Z}$ in Figure 7.6 is actually a triangular shaped fuzzy number. One will need to review [1] to understand their evolutionary algorithm method. We compare our Monte Carlo solution using Sobol quasi-random numbers and Kerre's method with the Evolutionary Algorithm solution from [1] and find that the $max\overline{Z}$ from our Monte Carlo solution is greater than the $max\overline{Z}$ from the Evolutionary Algorithm solution (regardless which of Buckley's method, Kerre's method, or Chen's method is used to compare the maximums).

Next we attempt to compare our Monte Carlo results using Chen's method with an Evolutionary Algorithm solution given in [1]. Unfortunately, we determined a discrepancy in those Evolutionary Algorithm results. That Evolutionary Algorithm solution satisfies neither the constraint equations nor the objective function. Thus we were not able to use them.

We see by comparing these fuzzy Monte Carlo solutions with a crisp solution that we have a solution not inconsistent with the crisp solution. Additionally, compared with an Evolutionary Algorithm solution, the fuzzy Monte Carlo Solution finds a greater fuzzy maximum.

References

1. Buckley, J.J., Feuring, T.: Evolutionary Algorithm Solution to Fuzzy Problems: Fuzzy Linear Programming. Fuzzy Sets and Systems 109, 35–53 (2000)
2. Buckley, J.J., Eslami, E., Feuring, T.: Fuzzy Mathematics in Economics and Engineering. Physica-Verlag, Heidelberg (2002)