Tabu Search Solution for Fuzzy Linear Programming

M. Hadi Mashinchi[1], M. Reza Mashinchi[2], M. Mashinchi[3],
[1,2] *Faculty of Computer Sciences and Information Systems, University Technology of Malaysia, Skudai, Johor, Malaysia,*
[3]*Faculty of Mathematics and Computer Sciences, Shahid Bahonar University of Kerman, Kerman, Iran.*
E-mail:h_mashinchi@yahoo.com[1], r_mashinchi@yahoo.com[2], mashinchi@mail.uk.ac.ir[3]

Abstract

In this paper, a tabu search (TS) method is proposed to find a global solution for fuzzy linear programming (FLP). Two additional factors, distribution factor (DF) and similarity factor (SF), are also introduced for TS to enhance the searching performance. The application of DF and SF makes the TS to surf in a larger search space and reduces the possibility of re-searching the visited areas. A simulation is done to show the efficiency of the proposed study.

1. Introduction

In the past decades, the conventional optimization models only dealt with crisp and exact information. However, many realistic problems could not be modeled by exact constraints or variables. A famous problem in this area is FLP, where the objective function that needs to be maximized (minimized) is comprised of fuzzy coefficients and subjects to some constraints. Furthermore, there is a possibility that the constraints are not exactly defined due to the type of problem, lack of information, or for generalization purposes. FLP is a useful tool for complex systems whenever the problem is tied up with human understanding of the problem or the exact information is not available. It is worth mentioning that FLP has been applied in many real applications such as energy optimizations [1], water allocation [2], risk investment analysis [3], and classification [4].

TS is a meta-heuristic method originally developed for combinatorial optimizations by Glover [5, 6]. It basically works based on generating random solutions and a list to prevent the vicious cycle while searching for the global solution. Generally speaking, TS requires some memory, known as tabu list (TL), to trace the recent states which have been investigated. The next potential solution is chosen from neighbors of current solution. The solution can be accepted as the next move if it is not tabued, or if it does so, it should satisfy the aspiration criterion. The existence of the TL makes TS a powerful technique to conduct global search rather than searching on a small portion of the search space.

Recently there has been a great tendency for application of soft-computing and evolutionary techniques in the field of fuzzy optimization problems. A simulated annealing method has been proposed for FLP [7] to conduct global search. Also, level set based neural networks have been introduced as solution for FLP [8, 9]. Furthermore, evolutionary algorithms have been widely used in this context. Firstly, Buckley and Hayashi initiated the application of evolutionary algorithms for FLP [10]. A more accurate two-phased genetic based method was then proposed in [11]. In a more recent approach [12], fuzzy membership functions are distributed into partitions. So instead of dealing with fuzzy membership functions, the genetic algorithms (GA) find the proper membership grades. As a result, the fuzzy computations in GAs do not need to undergo the extension principle or fuzzy arithmetic [12]. Apart from all the benefits of the evolutionary methods, unfortunately they are very slow to find a tender estimation [13]. Although all of the above mentioned methods have some advantages and have shown good performance, but none of them tries to reduce the possibility of cyclic search. Cyclic search is a very time consuming task and makes the searching process not necessarily to terminate with a global solution. This issue motivates us to apply an algorithm that can keep the traced states and also be able to search globally rather than locally.

978-0-7695-3131-1/08 $25.00 © 2008 IEEE
DOI 10.1109/ICIS.2008.105

In this paper, the application of TS solution with two additional factors is studied for FLP. TS has the ability to search for a global solution, while escaping from the cyclic search. Therefore, a larger search space is surfed. Moreover, the solution is more trustable since there is less possibility that a local minimum solution is returned.

2. Fuzzy linear programming

Traditional optimization problems dealt with maximizing (minimizing) a crisp function under some precise conditions. To be able to consider cases where the information is incomplete, fuzzy optimization was introduced [14]. There are different types of FLP depending on their type of constraints and objective functions. Table 1, which has been exactly taken from [7], lists various models of FLPs.

Table 1. Different modes of imprecision in fuzzy optimization [7]

Case 1	Imprecision in the constraints boundaries. This implies the fuzzification of the inequalities limits.
Case 2	A fuzzy goal is imposed on the objective function. Essentially this implies fuzzifying the utility function by considering a limiting goal (similar to goal programming).
Case 3	Compound imprecision. This implies combinations of the above sources of imprecision.
Case 4	The parameters (coefficients) of the variables of the constraints are not known precisely. This means that the coefficients are fuzzy numbers.
Case 5	The coefficients of the variables in the objective function are not known precisely. This means that coefficients are fuzzy numbers.
Case 6	All possible combinations of uncertainties e.g. [10].

Since to maximize (minimize) an objective function, the vagueness aspect is inevitable in realistic environments, the FLP attracts many attentions [15]. In this paper, we are interested in finding a solution for the Case 6 in Table 1 where usually was solved by either probabilistic programming or multiobjective programming methods [16, 17]. The objective functions are given in (1) or (3) and the constraints are defined in (2) or (4), respectively.

$$\max \quad \tilde{Z} = \tilde{C}_1 X_2 + \tilde{C}_2 X_2 + \cdots + \tilde{C}_n X_n \qquad (1)$$

s.t.

$$\tilde{A}_{i,1} X_1 + \tilde{A}_{i,2} X_2 + \cdots + \tilde{A}_{i,n} X_n \leq \tilde{B}_i \quad 1 \leq i \leq m \qquad (2)$$
$$X_i \geq 0$$

or,

$$\min \quad \tilde{Z} = \tilde{C}_1 X_2 + \tilde{C}_2 X_2 + \cdots + \tilde{C}_n X_n \qquad (3)$$

s.t.

$$\tilde{A}_{i,1} X_1 + \tilde{A}_{i,2} X_2 + \cdots + \tilde{A}_{i,n} X_n \geq \tilde{B}_i \quad 1 \leq i \leq m \qquad (4)$$
$$X_i \geq 0$$

To check the fuzzy inequality in (2) and (4), the same method used in [10] is applied. In (2) or (4), a vector of solution $\vec{X} = (X_1, X_2, \cdots, X_n)$ is either satisfies the conditions or does not. We say $\tilde{N} \leq \tilde{M}$, if (5) is true [10].

$$d(\tilde{N}, \tilde{\max}(\tilde{M}, \tilde{N})) \leq d(\tilde{M}, \tilde{\max}(\tilde{M}, \tilde{N})), \qquad (5)$$

where in (5) by $d(\tilde{M}, \tilde{N})$, we mean the Hamming distance between two fuzzy numbers $\tilde{M}$ and $\tilde{N}$, and is defined as (6).

$$d(\tilde{M}, \tilde{N}) = \int_{-\infty}^{+\infty} \left| \tilde{M}(x) - \tilde{N}(x) \right| dx. \qquad (6)$$

The maximum of two fuzzy numbers [10] is defined in (7).

$$\tilde{\max}(\tilde{M}, \tilde{N}) = \sup\left\{ \min(\tilde{M}(x), \tilde{N}(y)) \mid \max(x, y) = z \right\}.$$

3. Tabu-based FLP

TS is a very useful method whenever there exist many local minimums in the search domain. In contrast to local optimizer algorithms, TS is able to search in a large search space. The primary TS was intended for non-continues problems, while with some modifications it has been applied for global optimization in continues problems [18]. The reason of its capability to search globally is that it can keep the visited area in a limited memory, known as TL. The size of the memory has direct effect on the performance of the optimization problems. Although a very large size of TL broadens the search space, but slows down the process. Conversely, if the TL is very small, it can not effectively keep the traced states and avoid the cycles among the visited areas. Having the record of recent surfed states, the algorithm does not allow traversing any states close to the visited ones, unless they satisfy the aspiration condition [18]. The aspiration condition is set to activate the tabued states in the TL where they are potential to have some good global neighbors [19].

Furthermore, the way of distributing the neighbors around the current solution to find the next solution is very important. In case a large DF is considered, the algorithm is unlikely to find a very good approximation of solution. On the other hand, small DF makes TS to act like local optimizer methods. The "next solution" vector is generated as the following (8).

$$next\_solution_i = current\_solution + DF \times rand_i(), \quad i = 1,2,\cdots,n, \tag{8}$$

where the $rand()$ function produces a random value in the close interval of [-1,1], and n is the number of neighbors.

In continues problems, a range should be define to check whether a generated solution is close (similar) to any tabued solution or not. The similarity between two solutions is defined by below equation (9).

$$(next\_solution_i - SF < tabued\_solution_t) \;\&\; (tabued\_solution_t < next\_solution_i + SF), \quad i = 1,2,\cdots,n, \quad t = 1,2,\cdots,m, \tag{9}$$

where m is the maximum number of recent solutions that TL can keep.

If a generated solution (next_solution) for one or more states in TL satisfies the inequalities in (9) then it is considered as a tabued case.

The flowchart of the proposed TS solution for FLP is outlined in Figure 1.

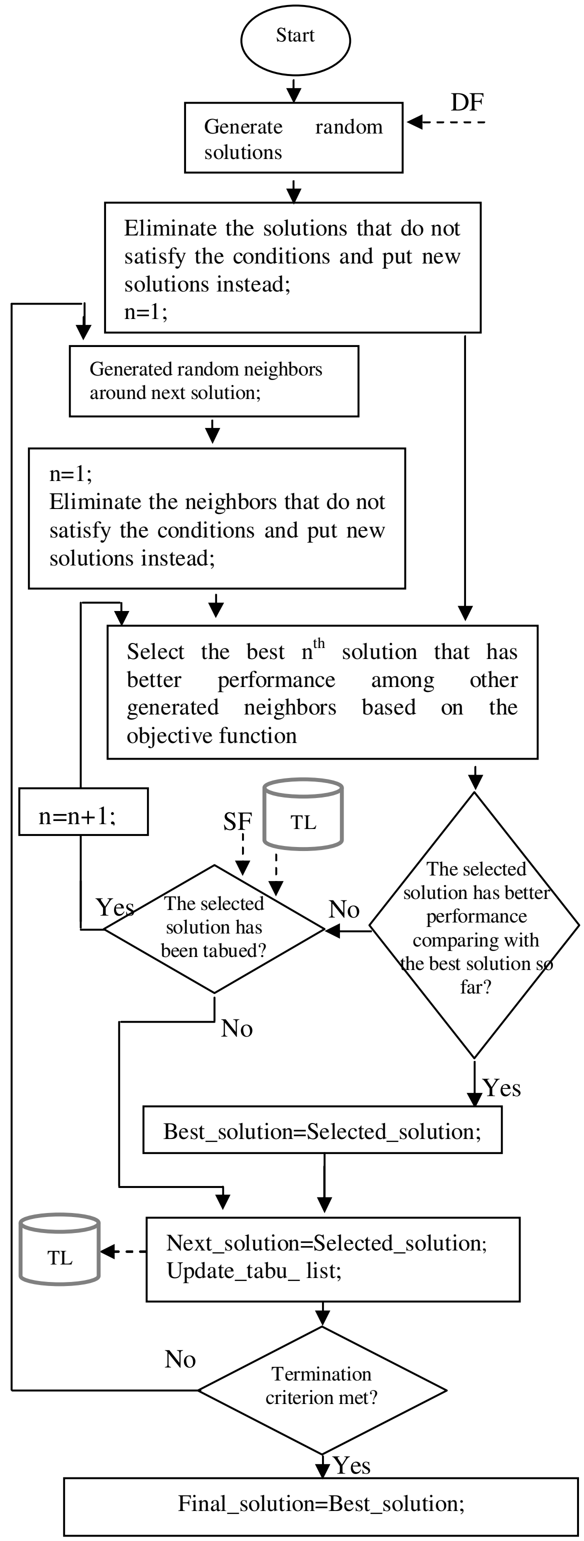

Figure 1. The flowchart of the TS solution for FLP

Although the TS is a potential tool for fuzzy optimization problems [20], but it has not yet been comprehensively studied for FLP. In Figure 1, to find the best global solution set which can satisfy the conditions, TS initially generates random solutions for the FLP. The random solutions are then checked to ensure if they satisfy the conditions or not. The unsatisfactory solutions are eliminated. Then the TS selects the best solution set as the next starting solution. New generation of the solution sets are randomly made around the current solution set according to the defined DF (8). Then the best non-tabued solution based on SF (9) is captured as the next solution. The aspiration condition guarantees that the good solutions, prohibited to choose, have the chance of consideration. These processes are executed till the termination criteria are met.

4. Computer experiment

For simulation purpose, an example is taken from [10] with minor revisions to be suitable for our study. The scenario is as following.

There are three different available products P1, P2 and P3 where a farmer can buy and feed his pigs. Also the pigs need certain daily amount of substances S1 and S2. Each needs nearly 54 and 60 units of S1 and S2 respectively. The term "nearly" is considered since the nutrition requirement of each pig is different from the other. To generalize the value to all pigs, we consider the value by a fuzzy triangular number. The amount of S1 and S2 that exists in each gram of P1, P2 and P3 are defined by the producers in Table 2.

Table 2. Approximate units of food Sj in each gram of the product Pj

Product	Substance	
	S1	S2
P1	2.5	5
P2	4.5	3
P3	5	10

The producers have provided all the given information. Although the information is measured according to some standards and the producers aim to stick to these values, but it is inevitable for all of products to provide exactly the given values of S1 and S2. Therefore, they are considered them with some fuzziness. The costs of P1, P2 and P3 are also various from day to day but the average costs are: (1) 8 cents per gram for P1; (2) 9 cents per gram for P2; and (3) 10 cents per gram for P3. The inexactness of the prices is considered by fuzzy triangular number too.

The owner of the pigs wants to satisfy the daily nutrition requirement of his pigs while spending the least possible amount of money. He therefore needs to know how many grams of each food should be bought. So the objective function is to find weights X_1, X_2 and X_3, of products P1, P2 and P3 respectively in which the daily cost gets minimized. The problem can be modeled as following (10).

$$\min \quad \tilde{Z} = (7,8,9)X_1 + (8,9,10)X_2 + (9,10,11)X_3 \qquad (10)$$

subject to:

$$(2,2.5,3)X_1 + (4,4.5,5)X_2 + (4.5,5,5.5)X_3 \geq (50,54,58),$$
$$(4.5,5,5.5)X_1 + (2.5,3,3.5)X_2 + (9,10,11)X_3 \geq (56,60,64),$$

and

$$X_1, X_2, X_3 \geq 0.$$

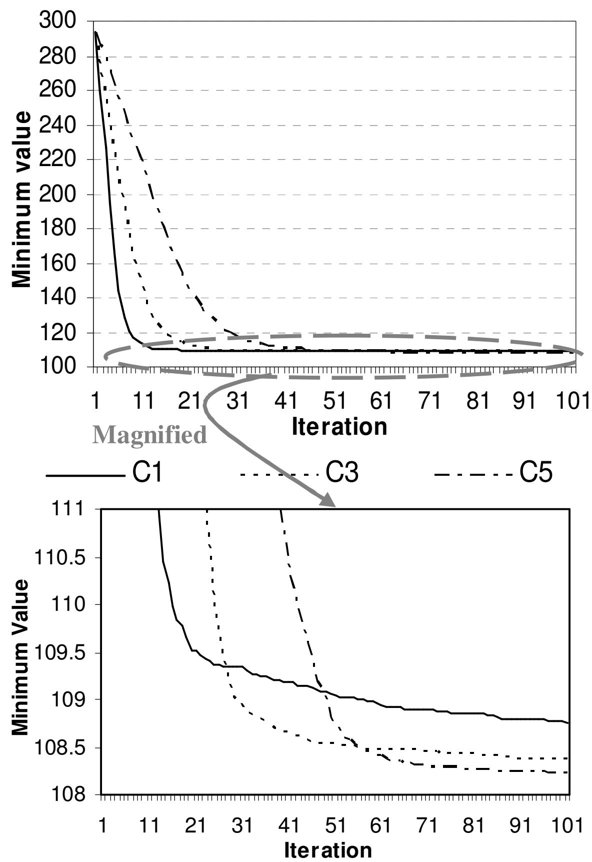

Figure 2. The convergence rate of modes C1, C3 and C5

Modes		C1	C2	C3	C4	C5	C6
Properties	TLS	10	30	10	30	10	30
	DF	2	2	1	1	0.50	0.50
	SF	1	1	0.45	0.45	0.20	0.20
The minimum obtained weights (in 60 trials)	X_1	0.0074	0.0121	0.0072	0.0009	0.0036	0.0013
	X_2	5.7946	4.8869	0.5824	0.7894	2.1098	3.8372
	X_3	5.5890	6.4023	10.2764	10.0964	8.9032	7.3483
	Minimum value	108.1006	108.1023	108.0638	108.1247	108.0763	108.0738
Average of tabued states for 60 trials with 100 iterations		237.38	290.83	158.87	206.98	92.35	128.17
Average of the minimized $\tilde{Z}$ for 60 trails with 100 iterations		**108.76**	**108.79**	**108.36**	**108.49**	**108.23**	**108.27**

Table 3. The simulation results of the given problem

The TS algorithm given in Figure 1 is applied to find the optimize value of X_1, X_2 and X_3 in which the minimum of $\tilde{Z}$ can be obtained.

Twenty random neighbors are considered for next solutions in each iteration. Different modes based on TL size, DFs and SFs are investigated and their prosperities are given in Table 3. To show the exact value that the owner should spend, the fuzzy number $\tilde{Z}$ was defuzzified by the centriod method. The average of the minimum values of $\tilde{Z}$ for 60 trials and other details are given in Table 3. The convergence rate to the minimum value for modes C1, C3 and C5 are illustrated in Figure 2. Apparently, the mode C1 has higher convergence rate at the beginning. On the other hand, mode C5 is potential to find a tender estimation since it can find better solution comparing to C1 at final iterations.

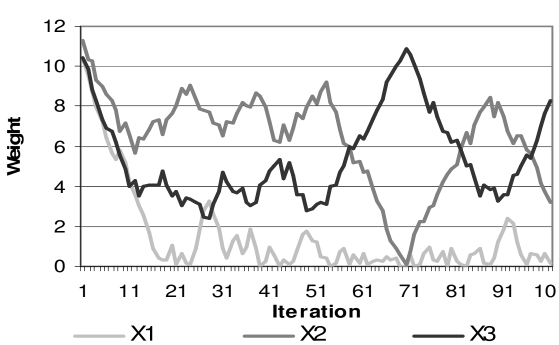

Figure 3. The searched X_1, X_2 and X_3

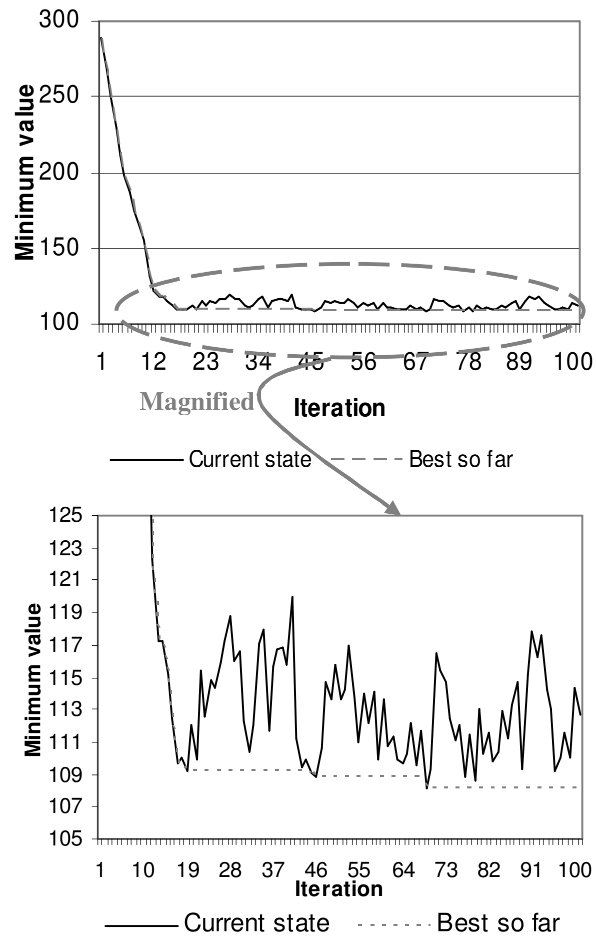

Figure 4. The convergence rate in best case for mode C3

To show more details of the studied TS algorithm, From Table 3 the best solution obtained in modes 3 is considered. The history of searched weights (X_1, X_2 and X_3) and the convergence rate are shown in Figures 3 and 4 respectively.

5. Conclusions

In this paper, the TS was applied to find a global optimized solution for FLP. Different modes of studied method were investigated according to their DF, SF and TL size. The results of these modes were shown in Table 3 and Figure 2. Apparently, DF, SF and TL size have great effect on the final results. According to the conducted simulation, if a fast and good solution is needed, then large DF and SF should be applied. On the other hand, if time is not a matter of interest, then a small DF and SF can be used to find a much optimized solution in a longer period of time. Also it was observed that the large TL size can decrease the goodness of final results.

As a future research trend, we tend to find optimized values of DF, SF and TL size dynamically during the searching process. A Comprehensive study should be also held for the comparison with other methods of solving FLP.

6. Acknowledgement

The third author is partially supported by Fuzzy Systems and Applications Centre of Excellence, Shahid Bahonar University of Kerman, Kerman, Iran.

7. References

[1] M. Sadeghi, H. M. Hosseini, "Energy supply planning in Iran by using fuzzy linear programming approach (regarding uncertainties of investment costs)", *Energy Policy*, Vol. 34 , 2006, pp. 993–1003.

[2] G. Tsakiris, M. spiliotis, "Fuzzy linear programming for the problem of water allocation under uncertainty", *European Water*, 7(8), pp. 25-37, 2004.

[3] R. C. Silva, J. L. Verdegay, A. Yamakami, "Two-phase method to solve fuzzy quadratic programming problems", *IEEE International Fuzzy Systems Conference*, London, 2007, pp. 1-6.

[4] A. Li, Y. Shi, J. He, "A data classification method based on fuzzy linear programming", *Proceeding of Multi-criteria Decision Making*, Chania, Greece, 2006.

[5] F. Glover, "Tabu search, part I", *ORSA J. Comp.*, Vol. 2, 1989, pp. 190-206.

[6] F. Glover, "Tabu search, part II", *ORSA J. Comp.*, Vol. 2, 1990, pp. 4-32.

[7] R. A. Ribeiro, F. M. Pires, "Fuzzy linear programming via simulated annealing", *Kybernetika*, 35(1), R. Yager, H-J Zimmermann, J. Kacprzyk (Eds), 1990, pp. 57-67.

[8] J. J. Buckley, Y. Hayashi, "Neural net solutions to fuzzy linear Programming", *Proc. FUZZ-IEEE '96*, New Orleans, 1996, pp. 1563-1566.

[9] J. J. Buckley, T. Feuring, Y. Hayashi, "Neural net solutions to fuzzy linear programming", *Fuzzy Sets and Systems*, Vol. 106, 1999, pp. 99-111.

[10] J. J. Buckley, T. Feuring, "Evolutionary algorithm solution to fuzzy problems: fuzzy linear programming", *Fuzzy Sets and Systems*, Vol. 109, 2000, pp. 35-53.

[11] K. Niwa, I. Nishizaki, M. Sakawa, "Computational methods for two-level linear programming Problems with fuzzy parameters through genetic algorithms", *Proc. of FUZZ-IEEE 2001*, 2001, pp. 1211-1214.

[12] F. Lin, M. Lee, "Fuzzy constraints linear programming: a genetic algorithms approach", *International Journal of Innovative Computing, Information and Control*, 3(1), 2007, pp. 71-83, 2007.

[13] Y. Wang, Y. Lu, "The combination of neural networks and genetic algorithm for fast and flexible wideing in digital beamforming", *Proc. of 9th International Conference on Neural Information Processing*, Singapore, 2002, pp.782-786.

[14] R. E. Bellman, L. A. Zadeh, "Decision-making in a fuzzy environment", *Management Science*, 17(4), 1970, pp. 141-164.

[15] H. Rommelfanger, "Fuzzy linear programming and applications", *European Journal of Operational Research*, Vol. 92, 1996, pp. 512-527.

[16] H. R. Maleki, M. Tata, M. Mashinchi, "Linear programming with fuzzy variables", *Fuzzy Sets and Systems*, Vol. 109, Issue. 1, 2000, pp. 21-33.

[17] E. Yazdani, H. R. Maleki, M. Mashinchi, "A method for solving a fuzzy linear programming", *Korean J. Comput. & Appl. Math.*, Vol. 8, No. 2, 2000, pp. 347-356.

[18] R. Chelouah, P. Siarry, "Tabu Search applied to global optimization", *European Journal of Operational Research*, Vol. 123, 2000, pp. 256-270.

[19] D. Cvijovic, J. Klinowski, "Taboo search: an approach to the multiple minima problem", *Science*, 267 (3), 1995, pp. 664–666.

[20] C. Li, X. Liao, J. Yu, "Tabu search for fuzzy optimization and applications", *Information Sciences*, Vol. 158, 2004, pp. 3-13.