Risk Assessment of Software Projects Using Fuzzy Inference System

Seyed Hossein Iranmanesh[1], Seyed Behrouz Khodadadi[2], Shakib Taheri[3]

[1]Industrial Engineering Department, College of Engineering, University of Tehran, Tehran, Iran, P.O. Box: 11155/4563 (Corresponding author: hiranmanesh@ut.ac.ir)
[2]University of Tehran, Iran (bkhodadadi@ut.ac.ir)
[3]University of Tehran, Iran (shakibtaheri@ut.ac.ir)

ABSTRACT

Risk management in software projects plays a vital role in the success of the project. Various risk factors in such projects make it difficult to make reliable and quick decisions in order to accept, mitigate, transfer or reject these risks and obtain an overall view of the whole project. In this paper it is introduced a fuzzy expert system which includes expertise to evaluate risk of software projects in all respects. Fuzzy inference has been used because of its capability in dealing with ambiguity and linguistic variables. Risk factors, the probability of failure and the severity of impact, are very close to fuzzy theory concepts. To develop our fuzzy expert system we deal with a rule base with about 17 million rules. Instead of constructing the whole rule base, a heuristic programming was created to infer the inputs without losing any rules. The output of the model is numerical values which present state of risk for each factor as well as the risk of project called the total risk. The results show better performance compared with traditional risk analysis system. The proposed tool can be used as a decision support system for top management to compare different projects or better risk mitigation in these projects.

Keywords: Risk assessment, Fuzzy inference system, Software projects, Expert systems, Fuzzy rule-based system

1. INTRODUCTION

One of the most important factors in making decisions to accept or reject a software project is considering the risk of it. Risk is a combination of probability or frequency of occurrence of an event and the severity or consequences of the hazard. Risk addresses the condition that is out of control of the project team and if it is not neutralized, it will cause adverse influence on the success of a project. Successful project managers try to solve the potential problems before they occur by using risk management tools [1].

Risk analysis is a review of the uncertainty associated with the research, development, and production of a product or a service [2] - in our case, software. Figure 1 shows the risk management process. As it is shown, before analyzing risk and using it in decision making we should identify hazard factors by gathering proper data. In this connection, we used a list of common software projects risk factors which has been developed by Schmidt [3] (Table 1).
In order to solve a problem, first we consider regular methods. In case they don't prove efficient then we should turn to different methods such as expert systems. Expert systems are information systems for solving a specific problem which provides an expertise similar to those of experts in the problem area [4]. One of the uses of expert systems is employing them in decision making. Therefore the subject of risk assessment due to its poor structure and being in need of expertness can have a good position for implementing these systems. A fuzzy expert system is an expert system that uses fuzzy logic. Zhiwei [5] has developed a fuzzy expert system to support risk assessment in the very early phase of the software life cycle and compared it with traditional risk assessment methods. Hadjimichael [6] And Dikmen [7] have used fuzzy expert system to assess risk in aviation and international construction projects respectively. Some recent research have also used fuzzy logic and fuzzy rule based modeling in their risk assessment methods in different fields such as petroleum projects [8], human health [9] and safety [10].

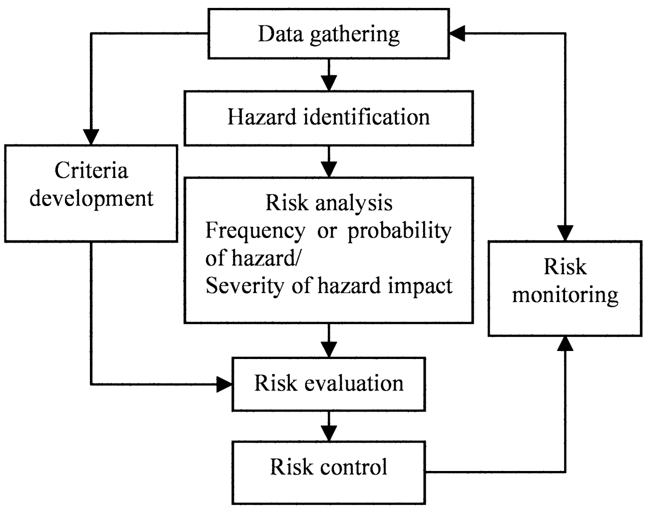

Figure 1 Risk management process

We have developed a two-layer fuzzy expert system to evaluate categorized risk factors and the total risk of software projects as a decision support tool. Experts usually use linguistic statements to describe variables such as severity of a hazard or probability of hazard occurrence. Risk matrix is also a linguistic decision making tool. As it known, fuzzy models are interactive models and obviously are more flexible than probability models with inflexible doctrine. Accordingly we chose fuzzy inference and developed fuzzy rule bases in each layer.

2. DEVELOPEMENT PROCESS

Fuzzy logic was presented by Zadeh [11] with the purpose of modeling uncertainty in natural language and it is used as logic of fuzzy expert systems. A complete review of fuzzy logic has been done by Lai and Hwang [12]. To develop a fuzzy expert system the following five steps are common:

1. Defining linguistic variables
2. Determining fuzzy sets and membership functions
3. Constructing fuzzy rules
4. Coding fuzzy inference
5. Tuning the system

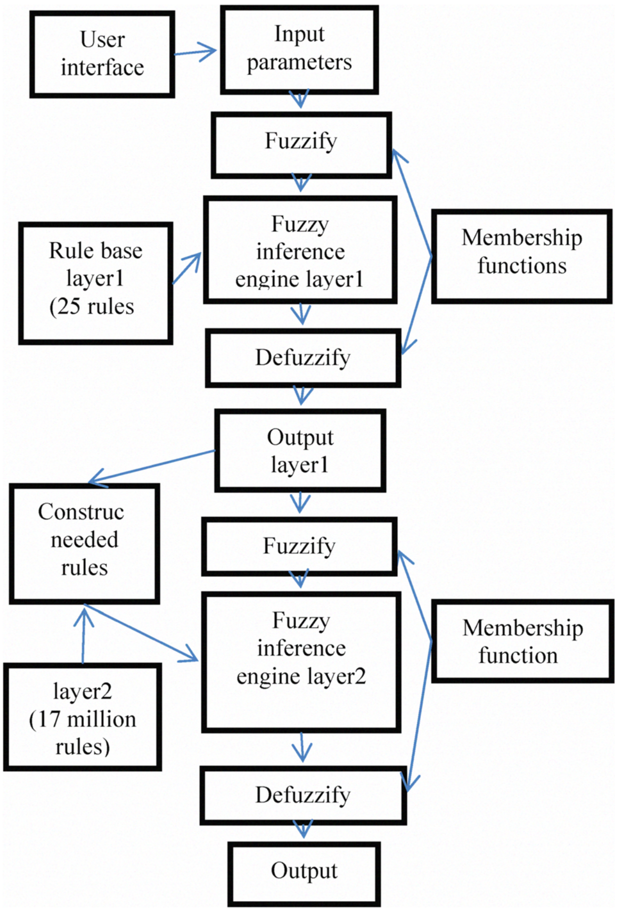

Figure 2 Function of proposed model

These steps have been implemented to develop the risk assessment fuzzy expert system. A brief view of the proposed model is depicted in Figure 2.

2.1 LINGUISTIC VARIABLES

In this paper we have considered software risk factors introduced by Schmidt [3] and summarized them in 12 major factors shown in Table 1.

Table 1 List of software projects risk factors

1.	Corporate Environment
2.	Sponsorship/Ownership
3.	Relationship Management
4.	Project Management
5.	Scope
6.	Requirements
7.	Funding
8.	Scheduling & Planning
9.	Development Process
10.	Personnel & Staffing
11.	Technology
12.	External Dependencies

For each factor, as it is usual, we consider two linguistic variables: probability and severity which we described earlier. We present probability of failure in each risk factor with five linguistic values i.e. very likely, likely, even, unlikely and very unlikely. To describe the severity of impact on the project we use these five linguistic values: very little, little, medium, high and catastrophic. The outputs of first layer of the model are the risks of each factor separately, which are presented by low, medium, significant and high. These outputs are the inputs of the second layer, and the output of second layer, that is, the total risk of the project is presented with linguistic values as well as risk of each factor, which are presented by low, medium, significant and high. Support ranges of fuzzy sets are shown in Table 2.

Table 2 Range of linguistic variables

Linguistic value	Numerical range
Linguistic variable: probability	
Very unlikely	[0,0.25]
Unlikely	[0.05, 0.5]
Even	[0.25, 0.75]
Likely	[0.5, 0.95]
Very likely	[0.75, 1]
Linguistic variable: severity	
Very little	[0, 2.5]
Little	[0.5, 5]
Medium	[2.5, 7.5]
High	[5, 9.5]
Catastrophic	[7.5, 10]
Linguistic variable: risk	
Low	[0, 0.33]
Medium	[0.05, 0.66]
Significant	[0.33, 0.95]
High	[0.66, 1]

2.2 MEMBERSHIP FUNCTIONS

The membership functions for probability and severity of factor 1 are shown in Figure 3 and Figure 4 respectively. The membership functions for risk of factor 1 are shown in Figure 5. We considered triangular membership functions for middle membership functions as it is usual. But we considered trapezium membership function for the side values except for the first value of probability membership functions, very unlikely value, which we presented with triangular function in order to be more careful about very small probabilities of failure.

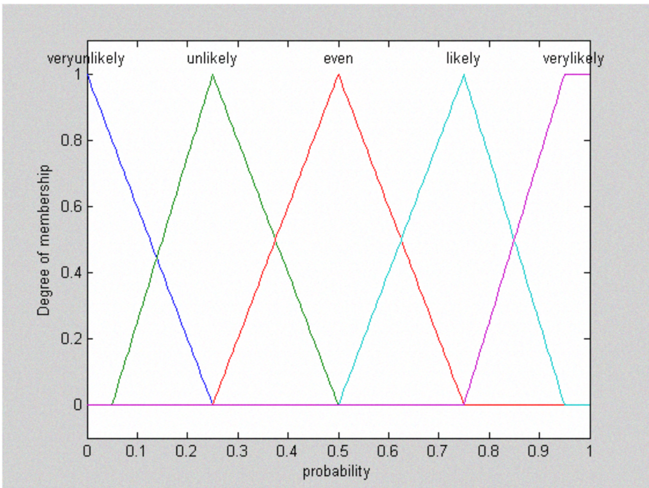

Figure 3 Probability membership function

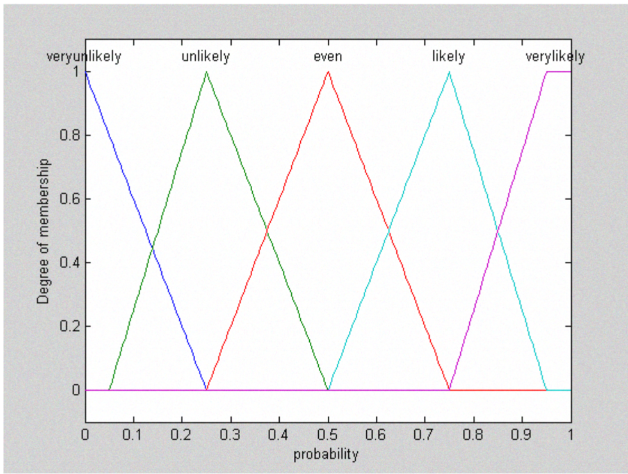

Figure 4 Severity membership function

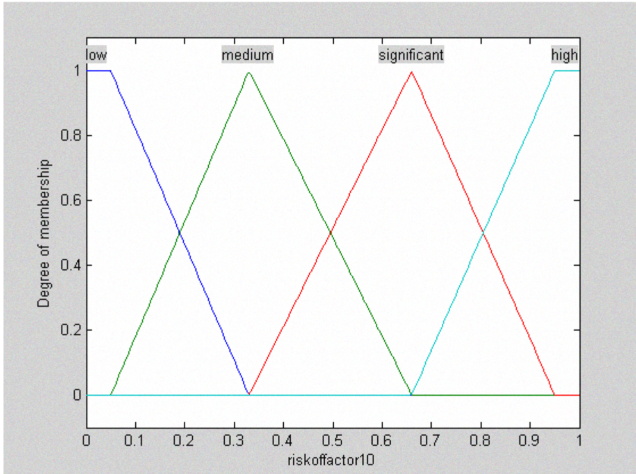

Figure 5 Risk membership function

Table 3 Risk matrix

Severity/ Probability	Very little	Little	Medium	High	Catastrophic
Very likely	Significant	Significant	High	High	High
Likely	Medium	Significant	Significant	High	High
Even	Low	Medium	Significant	High	High
Unlikely	Low	Low	Medium	Significant	High
Very Unlikely	Low	Low	Medium	Significant	Significant

2.3 RULE BASE

A rule consists of antecedents and consequents. Antecedents can be connected by different connectives such as AND, OR, NOT, NOR, ALSO. We might ask the expert to describe how the problem can be solved using the fuzzy linguistic variables defined previously. Required knowledge also can be collected from other sources such as books, computer databases, flow diagrams and observed human behaviour. To construct the rule base of first layer of the model we considered the risk matrix. Risk matrixes are useful in qualitative risk assessment. Table 3 shows a common risk matrix used in this study. We developed fuzzy risk matrix and it is where the difference between the old approach and our vision manifests itself. Therefore, we have 25 fuzzy rules, called fuzzy IF-THEN rules. As an example, one of the rules for the first factor is written below:

If probability of failure in factor 1 is related to VERY LIKELY and severity of its impact on the project is related to VERY LITTLE then the risk of factor 1 is related to SIGNIFICANT.

Graphical display of non-fuzzy risk matrix is shown in Figure 6 and would be comparable with proposed fuzzy risk matrix which is depicted in Figure 7. Figure 6 shows lots of jumps in degree of risk; obviously these jumps are not reasonable. In the proposed fuzzy rule base, there are no sharp boundaries for risk matrix as it shown in Figure 7.

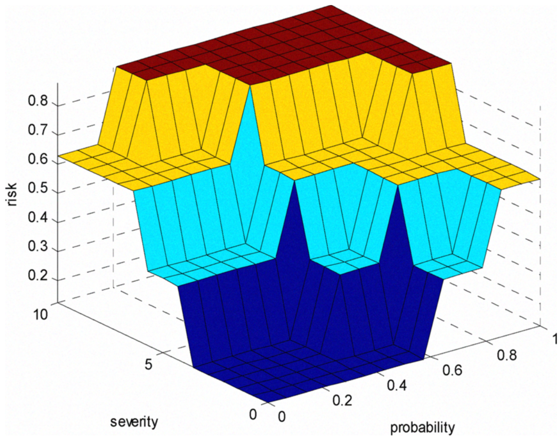

Figure 6 Graphical display of a non-fuzzy risk matrix

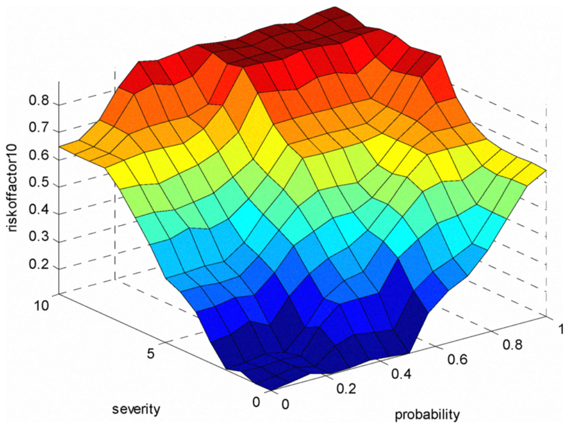

Figure 7 Graphical display of proposed fuzzy risk matrix

The second layer of the model is another multi input single output (called MISO) fuzzy inference system. The inputs of this layer are 12 factors' risk. The complete rule base comprises 4^{12}=16777216 rules. To build the rule base, first we gathered three experts' opinions about the total risk of a software project. These experts who have positions like project manager and process engineer closely deal with software and Information technology projects. A questionnaire was developed and experts were asked to determine the total risk of a software project by having different combinations of our factors' risk. By summarizing the answers we achieved some criteria. For example, the following statement is a rule which is concluded from one of the criteria:

If at least any two factors of the project have HIGH risk then the total risk of the project is related to HIGH.

2.4 SOLUTION METHOD

In our fuzzy inference system we use constructive method. We apply the most commonly used fuzzy inference technique called Mamdani method. The FITA (first inference then aggregate) approach is used because it has less computational complexity in comparison with FATI (first aggregate then inference) approach. The Mamdani style fuzzy inference process is performed in four steps:

1. Fuzzification of the input variables,
2. Rule evaluation,
3. Aggregation of the rule outputs,
4. Defuzzification

For defuzzification, we used the centroid technique. This deffuzzification method finds a point representing the center of the fuzzy set area; in fact this point is the center of gravity (COG).

2.5 CODING

For coding our model MATLAB programming language was used.

One way to code the rule base is to program the entire rule base and build all the rules for inference. Since in our case the number of rules are about 17 million, this approach will be highly time consuming in building the rule base as well as in inference stage while running the model. To deal with this problem we programmed the model so that in each run only the satisfied rules were built. To do this, we considered the fact that according to the shape of membership functions each input variable at most can satisfy two linguistic variables with non zero degree of membership. Therefore total fired rules cannot exceed 2^{12}=4096 for each input. The program only creates the satisfied rules and uses them in calculating the total risk of project.

3. COMPUTATIONAL EXPERIMENTS

The system is tested for a proposed project denoted by project A in a total solution composer company. The new risk factors list was used for risk analysis with both traditional method and fuzzy inference system. The traditional method calculates risk with multiplication of probability of failure and severity of impact in the same scale for each factor. Probability of failure was achieved from the previous statistics of software projects. Experts were asked to score the severity of impact of each factor's failure by considering the situation of projects and employers' behavior. The results of both methods were calculated. Figure 8 shows the difference between the outcomes of the two methods. Risks of factor 9 and factor 10 are obvious examples of discrepancy. As the traditional method deals with linear relation, it cannot be reliable. Also it is important to note traditional method will have no reasonable result for total risk of the project.

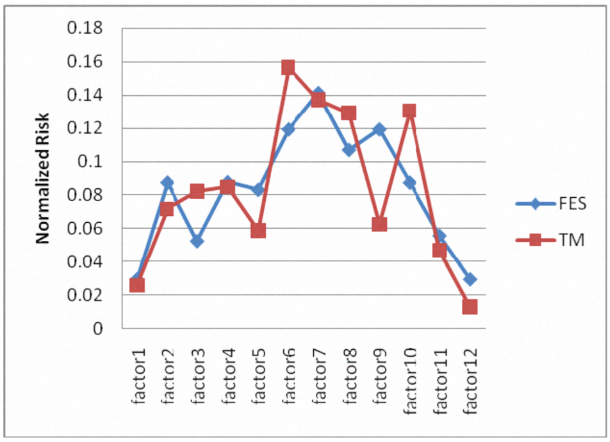

Figure 8 FES versus Traditional Method

To analyze the behavior of the system, some scenarios were tested. The results of sensitivity analysis for risk factor No.10 of project A are shown in Figure 8. We assumed all input parameters fixed except the probability of failure in factor 10. We changed the probability within the range of 0–1. As the consequence the increase in risk of factor 10 and its impact on the project were achieved. Figure 9 illustrates the results.

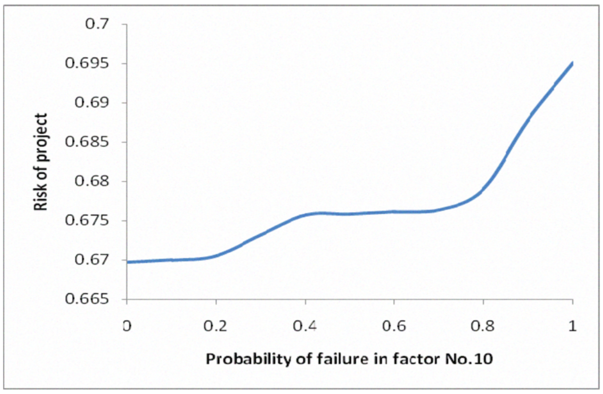

Figure 9 Effect of probability increase of factor 10 on the total risk

Finally to analyze the sensitivity of model, we assumed a virtual project, denoted by project B, which is very similar to project A with some differences in input parameters. Complete input parameters for both A and B projects, the risk of each factor and risk of projects evaluated by fuzzy expert system are shown in Table 4. Figure 10 compares the results.

In both projects we witness that either of the projects has factors with a greater risk versus the other one. Since project B has greater risk in a high risk factor its total risk is consequently higher than the total risk of project A.

Table 4 Input variables of numerical example

		The number of software risk factors adopted from table 1												
		1	2	3	4	5	6	7	8	9	10	11	12	
project A	probability of failure	0.2	0.11	0.21	0.13	0.1	0.2	0.15	0.18	0.08	0.2	0.12	0.1	total risk of project A
	severity of impact	1	5	3	5	4.5	6	7	5.5	6	5	3	1	
	risk	0.1180	0.3478	0.209	0.348	0.3305	0.4736	0.561	0.4248	0.4736	0.3475	0.2220	0.1180	0.6700
project B	probability of failure	0.3	0.11	0.21	0.13	0.1	0.2	0.15	0.18	0.08	0.1	0.12	0.1	total risk of project B
	severity of impact	1	5	3	5	3.5	7	7	5.5	6	5	2	2	
	risk	0.2128	0.3478	0.209	0.348	0.2673	0.5763	0.561	0.4248	0.4736	0.3475	0.1226	0.1180	0.6721

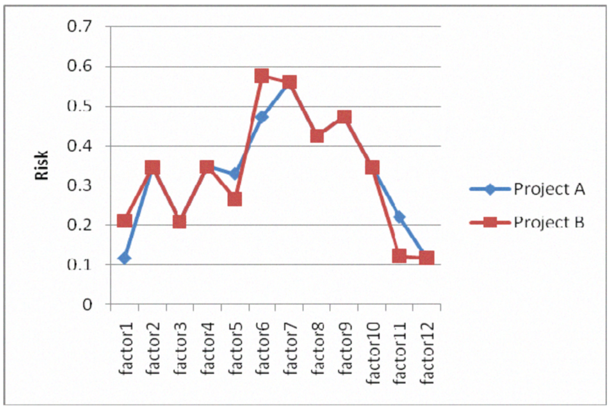

Figure 10 Project A versus project B

4. CONCLUSION

In this paper a risk assessment fuzzy expert system was developed to evaluate the risk of software projects. Risk factors were categorized in twelve categories. This system receives probability of failure and severity of impact as numerical values and after two layer fuzzy inference returns the risk of each factor as well as the total risk of the project. The achieved results might help decision makers to compare different projects. Required knowledge was reached from experts and the risk matrix. Numerical examples and different scenarios were deployed to validate the model. An extension of the system could be adding another layer to it by including risk sub factors and inference engine for them.

REFERENCES

[1] H. Iranmanesh, S. Nazari Shirkouhi, and M. R. Skandari, "Risk Evaluation of Information Technology Projects Based on Fuzzy Analytic Hierarchal Process", *International Journal of Computer and Information Science and Engineering* , Winter 2008.

[2] Kjell B.Zandin, "MAYNARD'S INDUSTRIAL ENGINEERING HANDBOOK", MCGRAW-HILL, Vol.1, 2001.

[3] Roy Schmidt, Kallelyytinen, Mark Keil, Paul Cule, "Identifying Software Projec Risk: An International Delphi Study", *Journal of Management Information System,* Spring 2001, Vol.17, No.4, pp.5-36

[4] N. Kasabov, "*Foundations of Neural Networks, Fuzzy Systems, and Knowledge Engineering"*, MIT Press, 1996, pp118-132.

[5] Zhiwei Xu, Taghi M. Khoshgoftaar, Edward B. Allen,"Application of fuzzy expert systems in assessing operational risk of software", *Information and Software Technology* 45 pp373–388, 2003

[6] Hadjimichael, M. "A fuzzy expert system for aviation risk assessment". *Expert Systems with Applications*, 2008.

[7] Irem Dikmen, M. Talat Birgonul, Sedat Han, "Using fuzzy risk assessment to rate cost overrun risk in international construction projects", *International Journal of Project Management* 25 pp.494–505, 2007.

[8] Mauro Roisenberg, Cintia Schoeninger, Reneu Rodrigues da Silva, "A hybrid fuzzy-probabilistic system for risk analysis in petroleum exploration prospects", *Expert Systems with Applications*, 2008.

[9] Chowdhury Shakhawat, Husain Tahrir, Bose Neil, "Fuzzy rule-based modelling for human health risk from naturally occurring radioactive materials in produced water", *Journal of Environmental Radioactivity* 89 pp.1-17, 2006

[10] R. Nait-Said, F. Zidani, N. Ouzraoui, "Modified risk graph method using fuzzy rule-based approach", *Journal of Hazardous Materials,*2008.

[11] Zadeh, L.. Fuzzy sets. Information Control, 8, pp.338-353, 1965.

[12] Y. J. Lai, Hwang, "Fuzzy Mathematical Programing ", *Springer*, pp.12-378, 1992.