A Formal Approach to the Assessment of Priorities of Imprecise Conflicting Requirements

John Yen and Xiaoqing Liu

Department of Computer Science, Texas A&M University, College Station, Texas 77843, U.S.A.

Abstract

Two major challenges with requirement engineering are (1) requirements are often *imprecise* in nature, and (2) requirements are usually *conflicting* with each other. One of the most important issues in the trade-off analysis of conflicting requirements is to understand the priority of requirements. Existing formal methods for requirement engineering are very limited in addressing these issues. This paper presents a formal methodology for specifying imprecise requirements, for characterizing complex relationships among them, for assessing their priority, and for facilitating the trade-off analysis. Imprecise requirements are represented by the canonical form in test-score semantics in fuzzy logic. Two kinds of relationships between requirements (i.e., conflicting and cooperative) are formally introduced. Multiple requirements with different priorities are aggregated based on the fuzzy multi-criteria decision technique. The assessment of the priority of requirements is facilitated by analyzing the trade-off between partial satisfaction of these imprecise requirements. The proposed approach can contribute not only to an early identification of conflicting requirements, but also to a better understanding the nature of conflict, which is vital to the negotiation process and the trade-off analysis during the design phase.

1 Introduction

Two major challenges with requirement engineering are (1) requirements are often *imprecise* in nature [1, 21], and (2) requirements are usually *conflicting* with each other [7, 15, 11, 21]. Most existing formal specification methodologies either require the requirements to be stated precisely as formal specification language Z [18], Larch [5], etc. or convert informal requirements into formal ones as ARIES [8], SAFE [1], Requirement Apprentice [13]), and SCENARIO [9]. The basic formal tool used for the specification of software systems is predicate logic and set theory[12]. Although they are powerful, it is difficult to use them to capture the imprecision of the requirements. Most earlier specification methods consider that a requirement specification containing conflicting requirements is inconsistent and should be avoided [17, 7]. More recently, a negotiation process was developed for detecting and resolving conflicts on design issues from multiple perspectives [15]. It facilitates collaboration which can benefit the requirement engineering process [6, 16]. This is an important step toward conflict resolution although it has not explicitly dealt with the analysis of conflicting imprecise requirements and the assessment of their priorities. The issue-based information systems (IBIS) method [14, 3] and the process-oriented approaches [10, 11] have been developed to document design rationales. Even though they can be used to detect conflicts, they are limited in their support for priority assessment and trade-off analysis.

One of the most important issues in the trade-off analysis of conflicting requirements is the assessment of their priorities. It has been demonstrated in [2] that the priority of requirements play an important role in the evaluation of software systems and in the choice of the optimal realization. However it is usually difficult for customers to accurately specify the importance of requirements due to the complex relationships between them. Thus we need a formal and systematic approach to assess and reason about the priority of imprecise conflicting requirements.

This paper presents a formal methodology for specifying imprecise requirements, for characterizing complex relationships among them, for assessing their priorities, and for facilitating the trade-off analysis. Imprecise requirements are represented by the canonical form in test-score semantics in fuzzy logic. The relationships between requirements are classified into being conflicting and cooperative. The priority and trade-offs between conflicting requirements are analyzed and formulated based on the intrinsic relationships between requirements. Multiple requirements with different priorities are aggregated based on the fuzzy multi-criteria decision technique. A technique which is based on the Extended Analytic Matrix is developed to assist analysts to identify the priorities of requirements with complex relationships between them.

2 The Representation, Classification, and Aggregation of Imprecise Requirements

2.1 Individual Imprecise Requirements

A target system T can be specified as a set of state transitions [19]. Let ST be the set of plausible state transitions $< s_1, s_2 >$ that can be performed by T, where s_1 is a before state, s_2 is a after state. A before state s_1, in general, may have more than one plausible after state s_2 such that $< s_1, s_2 > \in ST$. It indicates that different realization of T may implement different state transition. Let P be a realization of T, we use $exec(P, s_1)$ to denote the after state which is obtained by P when it is given a before state s_1. Based on the notion of state transitions, the functional requirements of a target system can be specified by characterizing the before states and its plausible after states using *preconditions* and *postconditions* respectively.

We distinguish two kinds of requirements: *rigid requirements* and *soft requirements*. A *rigid requirement* is a requirement that is either completely satisfied or completely dissatisfied. A *soft requirement* is a requirement that can be satisfied to a degree. To formally introduce soft functional requirements, we first extend the set of plausible state transitions of a requirement R of a target system to a fuzzy set of desired state transitions DST_R, which is characterized by a membership function μ_{DST_R} that maps a state transition $< s_1, s_2 >$ to the degree the state transition satisfies the soft requirement R of the target system. We can characterize a soft functional requirement using soft precondition and soft postcondition defined below:

Definition 1 *(Soft Functional Requirement [21]):*
A soft functional requirement R of the target system is specified as a pair of formula $< \varphi_1, \varphi_2 >$ where φ_1 is a soft precondition and φ_2 is a soft postcondition, such that $\forall < s_1, s_2 > \in Supt(DST_R)$,

$$\mu_{DST_R}(< s_1, s_2 >) \;=\; \mu_{fhold(\varphi_1, s_1) \overset{f}{\Longrightarrow} fhold(\varphi_2, s_2)}(< s_1, s_2 >)$$

where $Supt(DST_R)$ is the support of the fuzzy set DST_R (i.e., $Supt(DST_R) = \{< s_1, s_2 > | \mu_{DST_R}(< s_1, s_2 >) > 0\}$), "fhold" is a function that returns the degree to which a formula φ is true in state s, and $\overset{f}{\Longrightarrow}$ is a fuzzy implication operator.

Thus, a soft requirement specifies the "state changes" that are desired to be achieved to some degrees by a realization of the target system. In the following discussion, B_R will denote a set of before state s_1 of R and A_R a set of after state s_2 of R such that $\mu_{DST_R} < s_1, s_2 > > 0$

A soft requirement can be represented using the *canonical form* in Zadeh's test score semantics [23]. It has been established by the following definition [20].

Definition 2 *Let p be a proposition in its canonical form, X is A, X is a state variable in state s, and u_i is the value of X in s. Then*

$$fhold(p, s) \ = \ \mu_A(u_i)$$

2.2 Complex Relationships between Individual Requirements

Two kinds of relationships between requirements are of special importance: conflicting and cooperative. Using the elasticity in imprecise requirements, we are able to formally define and quantify *partially conflicting requirements* and *partially cooperative requirements.*

Two soft requirements are said to conflict with each other to a degree if an increase in the degree to which one requirement is satisfied often decreases the degree to which another requirement is satisfied.

Definition 3 *(conflicting degree with respect to a before state)*
Let $R_1 =< \varphi_1^1, \varphi_2^1 >$ and $R_2 =< \varphi_1^2, \varphi_2^2 >$ be two soft requirements of a target system. For a before state $b \in B_{R_1} \cup B_{R_2}$, let the set of common after states of R_1 and R_2 with respect to (wrt) b be denoted as $A_{R_1,R_2}(b) = A_{R_1} \cap A_{R_2}$. $\mathcal{F}(b)$ denotes the set of after state pairs, in which an increase in the degree to which a requirement is satisfied decreases the degree to which another requirement is satisfied, i.e.,

$$\begin{aligned}\mathcal{F}(b) = \{(a_1, a_2) | a_1, a_2 \in A_{R_1,R_2}(b), a_1 \neq a_2, and \\ (\mu_{DST_{R_1}}(b, a_1) - \mu_{DST_{R_1}}(b, a_2)) \times (\mu_{DST_{R_2}}(b, a_1) - \mu_{DST_{R_2}}(b, a_2)) < 0\}.\end{aligned}$$

Then the degree R_1 and R_2 are conflicting wrt the before state b, denoted as $conf(b)$, is

$$\frac{\|\mathcal{F}(b)\|}{\|A_{R_1,R_2}(b)\| \times (\|A_{R_1,R_2}(b)\| - 1)}.$$

The denominator of the formula for calculating conflicting degree is the cardinality of the set of after state pairs constructed from $A_{R_1,R_2}(b)$.

Having defined the conflicting degree wrt a given before state, we are ready to introduce an overall conflicting measure between two requirements.

Definition 4 *(conflicting degree)*
Let $R_1 =< \varphi_1^1, \varphi_2^1 >$ *and* $R_2 =< \varphi_1^2, \varphi_2^2 >$ *be two soft requirements of a target system. The conflicting degree of* R_1 *and* R_2 *is*

$$conf(R_1, R_2) \;=\; \frac{\sum_{b_k \in B_{R_1} \cup B_{R_2}} conf(b_k)}{\|B_{R_1} \cup B_{R_2}\|}.$$

Two soft requirements may cooperate with each other. It means an increase in the degree to which one requirement is satisfied often increases the degree to which another requirement is satisfied.

Definition 5 *(cooperative degree wrt a given before state)*
Assume that $R_1 =< \varphi_1^1, \varphi_2^1 >$ *and* $R_2 =< \varphi_1^2, \varphi_2^2 >$ *be two soft requirements of a target system* T*. For a before state* $b_k \in B_{R_1} \cup B_{R_2}$*, let* $\mathcal{G}(b_k)$ *denote the set of after state pairs, in which an increase in the degree to which a requirement is satisfied also increases the degree to which another requirement is satisfied, i.e.,*

$$\begin{aligned}\mathcal{G}(b_k) = \{&(a_1, a_2) | a_1, a_2 \in A_{R_1,R_2}(b_k), a_1 \neq a_2, and, \\ &(fhold(\varphi_2^1, a_1) - fhold(\varphi_2^1, a_2)) \times (fhold(\varphi_2^2, a_1) - fhold(\varphi_2^2, a_2)) \geq 0\}.\end{aligned}$$

Then the degree R_1 *and* R_2 *are cooperative wrt before state* b_k*, denoted as* $coop(b_k)$*, is*

$$\frac{\|\mathcal{G}(b_k)\|}{\|A_{R_1,R_2}(b_k)\|(\|A_{R_1,R_2}(b_k)\| - 1)}.$$

We now introduce an overall cooperative measure between two requirements.

Definition 6 *(cooperative degree)*
Assume that $R_1 =< \varphi_1^1, \varphi_2^1 >$ *and* $R_2 =< \varphi_1^2, \varphi_2^2 >$ *are two soft requirements of a target system. The cooperative degree of* R_1 *and* R_2 *is defined as*

$$coop(R_1, R_2) \;=\; \frac{\sum_{b_k \in B_{R_1} \cup B_{R_2}} coop(b_k)}{\|B_{R_1} \cup B_{R_2}\|}.$$

There is a dual relationship between conflicting degree and conflicting degree.

Theorem 1 *let* R_1 *and* R_2 *be two requirements, then*

$$conf(R_1, R_2) \;=\; 1 - coop(R_1, R_2).$$

This theorem can be proved easily according to the definitions given before.

Based on our definitions on the cooperative measure and conflicting measure, we introduce below the meaning of two important qualitative relationships between requirements: cooperative relationship and conflicting relationship.

Definition 7 *(conflicting or cooperative requirements) Two requirements R_1 and R_2 are said to be conflicting if and only if $conf(R_1, R_2) \geq coop(R_1, R_2)$; otherwise they are said to be cooperative.*

An approach based on case analysis (called Analytic Matrix) has been developed in [21] to estimate conflicting and cooperative degrees. We will describe an Extended Analytic Matrix for the assessment of the priority of requirements in Section 3. We have also developed a set of heuristics to reason about the relationships between requirements [21].

2.3 An Example

We use a knowledge-based system ISPBEX [4] to illustrate our methodology presented so far. The system will also be used to demonstrate our priority assessment method in Section 3. ISPBEX is a software system designed to determine the proper treatment recommendations for the suppression of Southern Pine Beetle (SPB), *Dendroctonus Frontalis Zimmerman* infestations. The postconditions of the system's soft requirements can be represented in the canonical form as follows:

- R_1 : Amount_of_Resource (Treatment Recommendation) should be SMALL.
- R_2 : Cost_of_implementation (Treatment Recommendation) should be LOW.
- R_3 : Responsive_Time (Treatment Recommendation) should be SHORT.
- R_4 : Negative_impact_to_endangered_species (Treatment Recommendation) should be LOW.
- R_5 : Profit_from_lumber (Treatment Recommendation) should be HIGH.

The relationships between requirements, which is the result of qualitative and quantitative analysis based on the proposed approach, is described in Figure 1 where $\ominus$ denotes conflicting requirements and $\oplus$ denotes cooperative requirements. A more detailed discussion on the identification of these relationships can be found in [21].

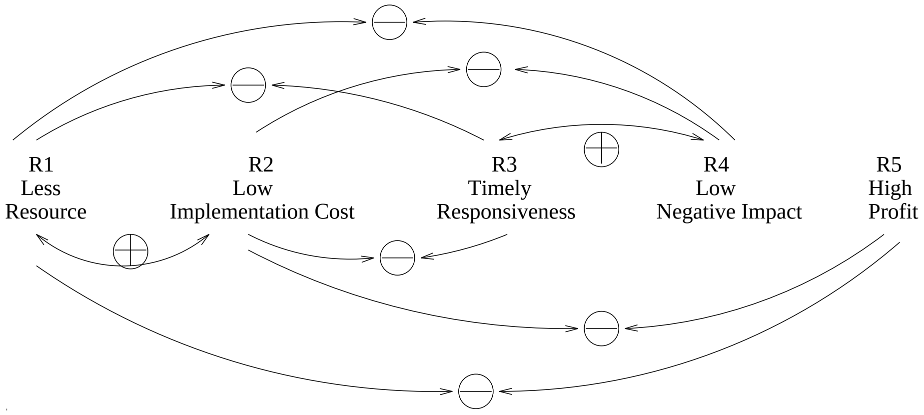

Figure 1: The Specification of Complex Relationships Between Requirements

2.4 Representing Priority

A set of requirements of a software system typically differ in their priority (i.e., degree of importance). A requirement R may carry a weight to reflect its priority. The weight may be a real number or as general as a linguistic value. In general, we describe a requirement R with criticality qualification using the form of

$$R \; is \; c$$

where c, a criticality qualifier, is a fuzzy subset of $[0, 1]$, the range of the criticality measure (one denotes the most important, zero denotes the least important). For example, in ISPBEX, a criticality can be attached to a requirement as following:

- R_1 : Amount_of_Resource (Treatment Recommendation) should be SMALL is IMPORTANT, or
- R_1 : Amount_of_Resource (Treatment Recommendation) should be SMALL is 0.6 degree important.

It is also important to express relative criticality such as *MORE-IMPORTANT-THAN*. More complex relative criticality relationships, such as *MUCH-MORE-IMPORTANT-THAN* and *A-LITTLE-MORE-IMPORTANT-THAN*, can also be introduced using fuzzy relation in fuzzy set theory [24]. In this paper, however, we will use only the simple relative criticality relationship: *MORE-IMPORTANT-THAN*.

2.5 The Composition of Soft Requirements With Priorities

Individual soft requirements can be combined into a composite soft requirements in different ways. A fuzzy conjunctive operator should be used if all of the requirements are required to be satisfied at the same same. They may be combined using a fuzzy disjunctive operator [1] if only one of them is required to be satisfied. A trade-off between requirements can be achieved through a fuzzy compromise operator, which is both an averaging operator and compensatory operator in fuzzy set theory [21]. It realizes the trade-offs by allowing compensation between requirements. The resulting compromise is between the minimal and the maximal degree of satisfaction of individual requirements. Examples of compromise operator include arithmetic average and geometric average. We briefly summarize the semantics of these three types of composite requirements below.

Definition 8 *(Aggregation of Soft Requirements):*

$$\mu_{DST_{R_1 \wedge R_2}}(< b, a >) = \mu_{DST_{R_1}}(< b, a >) \sqcap \mu_{DST_{R_2}}(< b, a >). \tag{1}$$

$$\mu_{DST_{R_1 \vee R_2}}(< b, a >) = \mu_{DST_{R_1}}(< b, a >) \sqcup \mu_{DST_{R_2}}(< b, a >). \tag{2}$$

$$\mu_{DST_{R_1 \diamond R_2}}(< b, a >) = \mu_{DST_{R_1}}(< b, a >) \otimes \mu_{DST_{R_2}}(< b, a >). \tag{3}$$

where $\sqcap$ denotes a fuzzy conjunctive operator, $\sqcup$ denotes a fuzzy disjunctive operator, $\diamond$ denotes trade-off*, and $\otimes$ denotes a compromise operator.*

The aggregation of soft requirements with numeric criticality depends on the type of aggregation operators used. There are three types of aggregation operators: t-norms, t-conorms, and compromise operators. We discuss how to aggregate the soft requirements with criticality measures using these aggregation operators.

1. The t-norms operator is used to aggregate the conjunctive requirements. It was pointed out by Yager [22] that it is the smaller terms that are most influential in the aggregation of multiple criteria using t-norms operators. Similarly, in the requirement engineering, it is

[1] In fuzzy set theory [24], the class of fuzzy conjunctive operator is called the t-norms operators, and the class of disjunctive operators is called the t-conorms operators. A typical example of t-norms operator is *min* and a typical example of t-conorms operator is *max*

the smaller degrees to which requirements are satisfied that are the most influential in the formulation using the t-norms operator. Thus we use the following formula to aggregate conjunctive requirements with criticality measures [22].

$$\mu_{DST_{R_1 \wedge, \ldots, \wedge R_n}}(< b, a >) = (\mu_{DST_{R_1}}(< b, a >))^{w_1} \sqcap \ldots \sqcap (\mu_{DST_{R_n}}(< b, a >))^{w_n}. \quad (4)$$

2. The t-conorms operator is used to aggregate disjunctive requirements. It is the larger degrees to which requirements are satisfied that are most influential in the formulation using the t-conorms operator. Thus we use the following aggregation formula for the inclusion of criticality measures.

$$\mu_{DST_{R_1 \vee, \ldots, \vee R_n}}(< b, a >) = w_1 \times \mu_{DST_{R_1}}(< b, a >) \sqcup \ldots \sqcup w_n \times \mu_{DST_{R_2}}(< b, a >). \quad (5)$$

3. The following formula can be used to aggregate requirements with criticality measures that are combined using a compromise operator.

$$\mu_{DST_{R_1 \diamond \ldots \diamond R_n}}(< b, a >) = \alpha_1 \times \mu_{DST_{R_1}}(< b, a >) \otimes \ldots \otimes \alpha_n \times \mu_{DST_{R_n}}(< b, a >). \quad (6)$$

where α_i denotes normalized priority, i.e., $\alpha_i = \frac{w_i}{\sum_{i=1}^{n} w_i}$.

3 Priority Assessment Based on Case Analysis

The purpose of requirement priority analysis is to establish a linear ordering of requirements based on their importance. Thus we need to help requirement analysts to identify the most important requirement and find the relative importance of requirements. However it is often difficult for customers to directly provide the priorities for all requirements due to complex relationships between them. Thus it is desirable to develop a technique to assist them in identifying the priority of each requirement. Since the weighted arithmetic average operator is one of most widely used compromise operator to aggregate requirements with complex tradeoff relationship, we focus our discussion on assessing the priority of requirements combined using this operator. It is defined as follows

$$\frac{\sum_{k=1}^{n} w_i \times x_i}{\sum_{k=1}^{n} w_i},$$

Case	Plausible Solution	Preference rank	R1	R2	. . .	Rn
EC_1	$S_{1,1}$	$r_{1,1}$	$d_{1,1,1}$	$d_{1,1,2}$	. . .	$d_{1,1,n}$
	$S_{1,2}$	$r_{1,2}$	$d_{1,2,1}$	$d_{1,2,2}$	. . .	$d_{1,2,n}$
	$\vdots$		$\vdots$	$\vdots$		$\vdots$
	$S_{1,m}$	$r_{1,m}$	$d_{1,m,1}$	$d_{1,m,2}$	. . .	$d_{1,m,n}$
$\vdots$				$\vdots$		
EC_l	$S_{l,1}$	$r_{l,1}$	$d_{l,1,1}$	$d_{l,1,2}$	. . .	$d_{l,1,n}$
	$S_{l,2}$	$r_{l,2}$	$d_{l,2,1}$	$d_{l,2,2}$	. . .	$d_{l,2,n}$
	$\vdots$		$\vdots$	$\vdots$		$\vdots$
	$S_{l,m}$	$r_{l,m}$	$d_{l,m,1}$	$d_{l,m,2}$	. . .	$d_{l,m,n}$

Figure 2: An Illustration of Extended Analytic Matrix

Case	Solution	Preference Rank	Requirement				
			R1	R2	R3	R4	R5
Case 1	C/L	3	0.4	0.4	0.6	0.7	0.2
	C/R	1	0.2	0.3	0.6	0.6	0.9
	P/B	4	0.1	0.1	0.8	0.1	0.1
	Mon	5	0.8	0.8	0.3	0.5	0.1
	C/HS	2	0.3	0.2	0.7	0.8	0.1
Case 2	C/L	3	0.3	0.3	0.4	0.5	0.1
	C/R	2	0.4	0.3	0.6	0.6	0.4
	P/B	1	0.1	0.1	0.7	0.7	0.1
	Mon	5	0.7	0.9	0.2	0.3	0.1
	C/HS	4	0.1	0.4	0.4	0.4	0.1
Case 3	C/L	1	0.6	0.7	0.6	0.6	0.3
	C/R	4	0.4	0.4	0.6	0.3	0.6
	P/B	5	0.2	0.1	0.7	0.1	0.1
	Mon	2	0.8	0.8	0.2	0.5	0.1
	C/HS	3	0.5	0.2	0.3	0.4	0.1
Case 4	C/L	3	0.3	0.3	0.7	0.7	0.5
	C/R	4	0.1	0.1	0.8	0.6	0.7
	P/B	5	0.1	0.1	0.5	0.5	0.1
	Mon	1	0.9	0.8	0.6	0.9	0.1
	C/HS	2	0.5	0.5	0.7	0.7	0.1

Figure 3: The Extended Analytic Matrix for ISPBEX

where w_i is the weight of x_i and $w_i \geq 0$. In this section, we describe a technique for inferring qualitative statements about relative or absolute priority based on case analysis. More specifically, we have developed several theorems to assess the priority of requirements aggregated by a compromise operator based on an analytic tool called Extended Analytic Matrix (EAM), which is an extension of the Analytic Matrix described in [21]. Let T denote a target system with n individual requirements, denoted as R_k, $1 \leq k \leq n$. A set of l typical cases, denoted as EC_i, $1 \leq i \leq l$ are formulated by the customers. The customer is then asked to provide a set of m possible solutions for each case EC_i, denoted as $S_{i,j}$, $1 \leq j \leq m$. For each solution $S_{i,j}$, a degree $d_{i,j,k}$, to which the solution $S_{i,j}$ for the case EC_i satisfies the requirement R_k can be assigned. In addition, for each case, solutions are ranked from one to m according to their preference by the customers. The rank of preference of the jth solution for the ith case is represented by $r_{i,j}$ (a lower ranking number means a preferred solution). These degrees are organized into a matrix, called Extended Analytic Matrix (EAM), $EAM = \{r_{i,j}, d_{i,j,k}\}$, where $1 \leq i \leq l, 1 \leq j \leq m$, and $1 \leq k \leq n$ as shown in Figure 2.

To illustrate the concept of the Extended Analytic Matrix, let us consider the system ISPBEX which we discussed in Section 2.3. In ISPBEX, the input parameters are treatment month, type of forest (general forest, wilderness), colony-condition (active, not active), breeding season (spring, summer, fall, winter), type of trees (loblobby or short leaf pine), characteristics of trees (age, size), growth distance (in feet) and direction of growth (in degrees). The output is a treatment recommendation and explanation why a particular recommendation is provided. Thus a case corresponds to a set of fixed input parameters. For example, Case 1 in the extended analytic matrix shown in Figure 3 represents that the treatment season is March, the type of forest is wilderness, SPB spot is active, endangered species are breeding, SPB is going to impact the endangered species colony within 60 days, and so on. The extended analytic matrix for ISPBEX includes four typical cases. For each case, five major recommendations are possible. They include Cut and Leave(C/L), Cut and Remove(C/R), Pile and Burn(P/B), Monitor(Mon), and Cut and Hand Spray(C/HS). They have been ranked for each case by the customer based on their preference.

3.1 Identifying the Requirement of the Highest Priority

Since an individual requirement can be viewed as a requirement group containing only one requirement, the identification of the most important group of requirements is a more general problem than the identification of the most important requirement. A set of requirements $R1, R_2, \ldots, R_n$ can be partitioned into disjoint groups of requirements $G_1, G_2, \ldots, G_p$ so that all requirements in each group G_h $(1 \leq h \leq p)$ have the same priority. Intuitively, a group of requirements G_{h_1} must be the most important one if the customer is willing to trade off a larger total decrease on the satisfaction of every other groups of requirement than G_{h_1} for a small total increase in the degree G_{h_1} is satisfied. This is formally stated in the theorem below.

Theorem 2 *Suppose that requirements $R_1, R_2, \ldots, R_n$ are aggregated by the weighted arithmetic average operator and partitioned into disjoint groups $G_1, G_2, \ldots, G_p$ so that all requirements in each group G_h $(1 \leq h \leq p)$ have the same priority w_{G_h}. Let j_1 and j_2 be two solutions for case i, δ_k denote the difference between the degrees R_k is satisfied by the two solutions, i.e., $\delta_k = d_{i,j_2,k} - d_{i,j_1,k}$ and δ_{G_h} denote the difference between the total degrees group G_h is satisfied by the two solutions, i.e., $\delta_{G_h} = \sum_{R_{k'} \in G_h} \delta_{k'}$. If (1) solution j_2 is preferred to solution j_1, i.e.,*

$r_{i,j_1} > r_{i,j_2}$, *(2) The solution j_2 improves the total degree the group of requirement G_{h_1} is satisfied in case i, i.e., $\delta_{G_{h_1}} > 0$ where $1 \le h_1 \le p$, (3) The solution j_2 decreases the total degree every other group G_h of requirements are satisfied, i.e., $\delta_{G_h} < 0$ for all $h \ne h_1$ and $1 \le h \le p$, and (4) The amount of increase on the total degree G_{h_1} is satisfied is not larger than the amount of decrease on the total degree every other group of requirements is satisfied introduced by solution j_2, i.e., $\|\delta_{G_h}\| \ge \delta_{G_{h_1}}$, then G_{h_1} is the most important group of requirements, i.e., for all $h \ne h_1$ and $1 \le h \le p$,*

$$w_{G_{h_1}} > w_{G_h}.$$

where $w_{G_{h_1}}$ and w_{G_h} denote the priorities of requirements in G_{h_1} and G_h respectively.

Proof: Let $D_{i,j}$ denote the overall degree that the composite requirement is satisfied by solution j in the ith case, i.e., $D_{i,j} = \frac{\sum_{k=1}^{n} w_k \times d_{i,j,k}}{\sum_{k=1}^{n} w_k}$. Since $r_{i,j_1} > r_{i,j_2}$, we have $D_{i,j_2} > D_{i,j_1}$. Thus,

$$\sum_{k=1}^{n} w_k \times d_{i,j_2,k} > \sum_{k=1}^{n} w_k \times d_{i,j_1,k}.$$

Thus,

$$\sum_{k=1}^{n} w_k \times \delta_k > 0.$$

Since $G_1, \ldots, G_p$ forms a partition of the n requirements, we have

$$\sum_{h=1}^{p} w_{G_h} \times (\sum_{R_{k'} \in G_h} \delta_{k'}) > 0.$$

That is,

$$\sum_{h=1}^{p} w_{G_h} \times \delta_{G_h} > 0.$$

That is,

$$w_{h_1} \times \delta_{G_{h_1}} > -\sum_{h=1, h \ne h_1}^{p} w_{G_h} \times \delta_{G_h}.$$

Since δ_{G_h} $(h \ne h_1)$ are all negative from the third condition in the theorem, we have

$$w_{G_{h_1}} \times \delta_{G_{h_1}} > \sum_{h=1, h \ne h_1}^{p} w_{G_h} \times \|\delta_{G_h}\|. \tag{7}$$

Let G_{h_2} be an arbitrary group of requirements that is not G_{h_1} and $1 \le h_2 \le p$. Since all the weights w_{G_h} are non-negative, we have

$$\sum_{h=1, h \ne h_1}^{p} w_{G_h} \times \|\delta_{G_h}\| \ge w_{G_{h_2}} \times \|\delta_{G_{h_2}}\|. \tag{8}$$

Case	Solution	Preference Rank	R1	R2	R3	R4	R5
			Requirement				
Case 2	C/L	3	0.3	0.3	0.4	0.5	0.1
	C/R	2	0.4	0.3	0.6	0.6	0.4
	P/B	1	0.1	0.1	0.7	0.7	0.1
	Mon	5	0.7	0.9	0.2	0.3	0.1
	C/HS	4	0.1	0.4	0.4	0.4	0.1

$\Longrightarrow$ R3 and R4 are more important than any other requirement

Figure 4: An Illustration of Finding the Most Important Requirement Group

Combining equations 7 and 8, we have

$$w_{G_{h_1}} \times \delta_{G_{h_1}} > w_{G_{h_2}} \times \|\delta_{G_{h_2}}\|.$$

Since $\|\delta_{G_{h_2}}\| \geq \delta_{G_{h_1}}$ from the fourth condition,

$$\frac{w_{G_{h_1}}}{w_{G_{h_2}}} > \frac{\|\delta_{G_{h_2}}\|}{\delta_{G_{h_1}}} \geq 1.$$

Thus, we have

$$w_{G_{h_1}} > w_{G_{h_2}}.$$

Since this holds for all $h_2 \neq h_1$, we have proved the theorem. $\Box$

This theorem is illustrated by an example shown in Figure 4, which is a portion of the Extended Analytic Matrix of ISPBEX shown in Figure 3. The customer indicated that requirements R_1 is same important as R_2 and R_3 is same important as R_4. Thus requirements R_1, R_2, R_3, R_4 and R_5 can partitioned into three groups: $G_1 : R_1, R_2$, $G_2 : R_3, R_4$ and $G_3 : R_5$. The customer also indicates the treatment P/B is better than C/R for the second case. Since the preferred solution increases G_3's total satisfaction degree by only 0.2 although it reduces G_1's total satisfaction degree by 0.5 and G_3's total satisfaction 0.3 respectively, we can infer from the theorem that group G_2 is the most important one, that is, the priority of R_3 and R_4 are higher than those of R_1, R_2 and R_5.

It is easily seen from the theorem that if a group of requirements have the same priority, and the customer is willing to trade-off a larger decrease on every other requirement for a small total increase on the degrees requirements in the group are satisfied, then each requirement in the group has a higher priority than those of requirements not in the group.

If a group containing only one requirement is inferred as the most important one from the theorem, the requirement is the most important one. If each group contains only one requirement in the above theorem, we naturally arrive at the following corollary to identify the most important requirement. Intuitively, a requirement R_{k_1} must be the most important one if the customer is willing to trade off a larger amount of decrease on the satisfaction of all requirements except R_{k_1} for a small amount of increase in the degree R_{k_1} is satisfied. This is formally stated in the theorem below.

Corollary 1 *Suppose that requirements $R_1, R_2, \ldots, R_n$ are aggregated by the weighted arithmetic average operator. Let j_1 and j_2 be two solutions for case i and δ_k denotes the difference between the degrees R_k is satisfied by the two solutions, i.e., $\delta_k = d_{i,j_2,k} - d_{i,j_1,k}$. If (1) solution j_2 is preferred to solution j_1, i.e., $r_{i,j_1} > r_{i,j_2}$, (2) The solution j_2 improves the degree requirement R_{k_1} is satisfied, i.e., $\delta_{k_1} > 0$ where $1 \leq k_1 \leq n$, (3) The solution j_2 decreases the degrees all other requirements are satisfied, i.e., $\delta_k < 0$ for all $k \neq k_1$ and $1 \leq k \leq n$, and (4) The amount of increase on the degree R_{k_1} is satisfied is not larger than the amount of decrease on the degree every other requirement is satisfied introduced by solution j_2, i.e., $\|\delta_k\| \geq \delta_{k_1}$, then R_{k_1} is the most important requirement, i.e., for all $k \neq k_1$ and $1 \leq k \leq n$,*

$$w_{k_1} > w_k.$$

where w_{k_1} denotes the priority of requirement R_{k_1} and w_k denotes the priority of requirement R_k.

This corollary is illustrated by an example shown in Figure 5. The customer prefers the second solution over the first one. Since the preferred solution increases R_1's satisfaction degree by 0.1, but reduces satisfaction degree of R_2 and R_3 by 0.3 and 0.2 respectively, we can infer from the corollary that R_1's priority is the highest.

3.2 Identifying the Relative Importance Relationship between Requirements

So far we have described a theorem that facilitates the identification of the most important individual requirement or the most important requirement group by analyzing the customer's

Case	Plausible Solution	Preference rank	R1	R2	R3
EC_1	$S_{1,1}$	2	0.4	0.7	0.8
	$S_{1,2}$	1	0.5	0.4	0.6
. . .				. . .	

$\Longrightarrow$ R1 is the most important requirement

Figure 5: An Illustration of Identifying the Most Important Requirement

preferred trade-offs through case analysis. This theorem also form the basis for identifying relative importance relationship between individual requirements or requirement groups. Since an individual requirement can be viewed as a requirement group containing only one requirement, the identification of relative importance between two requirement groups is a more general problem than the other two problems, i.e. the identification of single-group or single-single relative importance relationship.

Assuming that G_1 and G_2 are two disjoint requirement groups, each of which contains a set of requirements of same priority. If the customer is willing to trade-off a larger total decrease in the satisfaction of G_2 for a small total increase in the satisfaction of G_1, the requirements in G_1 must have a higher priority than those in G_2. This is formally stated in the theorem below.

Theorem 3 *Suppose that requirements $R_1, R_2, \ldots, R_n$ are aggregated by the weighted arithmetic average operator and partitioned into disjoint groups $G_1, G_2, \ldots, G_p$ so that all requirements in each group G_h $(1 \leq h \leq p)$ have the same priority w_{G_h}. Let j_1 and j_2 be two solutions for case i. δ_k denotes the difference between the degrees R_k is satisfied by the two solutions, i.e., $\delta_k = d_{i,j_2,k} - d_{i,j_1,k}$ and δ_{G_h} denotes the difference between the total degrees group G_h is satisfied by the two solutions, i.e., $\delta_{G_h} = \sum_{R_{k'} \in G_h} \delta_{k'}$. If (1) solution j_2 is preferred to solution j_1, i.e., $r_{i,j_1} > r_{i,j_2}$, (2) The solution j_2 improves the total degree the group of requirement G_{h_1} is satisfied in case i, i.e., $\delta_{G_{h_1}} > 0$ where $1 \leq h_1 \leq p$, (3) The solution j_2 decreases the total degree another group G_{h_2} of requirements are satisfied, i.e., $\delta_{G_{h_2}} < 0$ where $h \neq h_1$ and $1 \leq h \leq p$, and (4) The amount of increase on the total degree G_{h_1} is satisfied is not larger than the amount of decrease on the total degree G_{h_2} is satisfied introduced by solution j_2, i.e., $\|\delta_{G_{h_2}}\| \geq \delta_{G_{h_1}}$, (5) There is no change on the total degree every other group is satisfied, i.e., $\delta_{G_h} = 0$ where $h \neq h_1, h_2$, $1 \leq h \leq p$*

Case	Solution	Preference Rank	R1	R2	R3	R4	R5
Case 4	C/L	3	0.3	0.3	0.7	0.7	0.5
	C/R	4	0.1	0.1	0.8	0.6	0.7
	P/B	5	0.1	0.1	0.5	0.5	0.1
	Mon	1	0.9	0.8	0.6	0.9	0.1
	C/HS	2	0.5	0.5	0.7	0.7	0.1

$\Longrightarrow$ R1 and R2 is more important than R5

Figure 6: An Illustration of Finding the More Important Requirement Group

and then we have

$$w_{G_{h_1}} > w_{G_{h_2}}.$$

That is, the priority of the requirements in G_{h_1} is higher than that of requirements in G_{h_2}

This theorem can be proved similarly as Theorem 2. It is illustrated by an example shown in Figure 6, which is a also portion of the Extended Analytic Matrix of ISPBEX shown in Figure 3. As stated before, R_1, R_2, R_3, R_4 and R_5 can partitioned into three groups: $G_1 : R_1, R_2$, $G_2 : R_3, R_4$ and $G_3 : R_5$. All requirements in each group have the same priority. The customer indicated that treatment C/HS is preferred over C/L for the fourth case. Comparing the two solutions, we can realize that the customer is willing to trade off a decrease in R_5 for the same amount of total increase (0.4) in $G_1 : R_1, R_2$, while degrees to satisfy $G_2 : R_3, R_4$ remain the same in two solutions. Based on the Theorem 3, we can conclude that $G_1 : R_1, R_2$ has a higher priority than R_5 does. By combining the relative importance identified in Figure 4 and Figure 6, we obtain the ordering of the requirements' priority in ISPBEX shown in Figure 7, i.e.,

$$P(R_3) = P(R_4) > P(R_1) = P(R_2) > P(R_5),$$

where $P(R_i)$ denotes the priority of requirement R_i.

Notice that a requirement group may contain only one requirement in Theorem 3. Thus the theorem can facilitate the identification of group-group, single-group, single-single relative importance between requirements. If each group contains only one requirement in the theorem, we naturally arrive at the following corollary for identifying single-single relative importance between requirements. Intuitively, The priority of requirement R_{k_1} must be higher that of requirement R_{k_2} if the customer is willing to trade off a larger amount of decrease on the satisfaction of

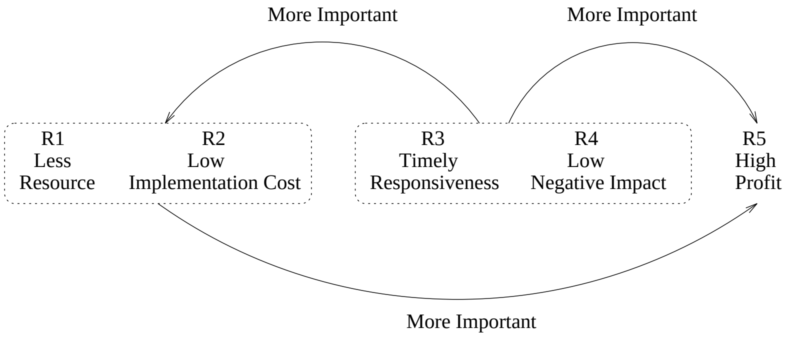

Figure 7: An Illustration of Priorities of Requirements in ISPBEX

requirement R_{k_2} for a small amount of increase in the degree R_{k_1} is satisfied. This is formally stated in the corollary below.

Corollary 2 *Suppose that $R_1, R_2, \ldots, R_n$ are aggregated by the weighted arithmetic average operator. Let j_1 and j_2 be two solutions for case i and δ_k denotes the difference between the degrees R_k is satisfied by the two solutions, i.e., $\delta_k = d_{i,j_2,k} - d_{i,j_1,k}$. If (1) solution j_2 is preferred to solution j_1, i.e., $r_{i,j_1} > r_{i,j_2}$, (2) The solution j_2 improves the degree requirement R_{k_1} is satisfied, i.e., $\delta_{k_1} > 0$ where $1 \leq k_1 \leq n$, (3) The solution j_2 decreases the degrees all other requirements are satisfied, i.e., $\delta_{k_2} < 0$ where $k_2 \neq k_1$ and $1 \leq k_2 \leq n$, (4) The amount of increase on the degree R_{k_1} is satisfied is not larger than the amount of decrease on the degree R_{k_2} is satisfied introduced by solution j_2, i.e., $\delta_{k_1} \leq \|\delta_{k_2}\|$, and (5) The degree every other requirement is satisfied is not changed, i.e., $\delta_k = 0$ where $k \neq k_1, k_2$, then the priority of requirement R_{k_1} is higher than that of requirement R_{k_2}, i.e.,*

$$w_{k_1} > w_{k_2}.$$

When analyzing the relative importance of requirement groups, we have so far assumed that all requirements in the group have the same priority, which can be a strong assumption. In situations in which requirements in the group do not have the same priority, the conclusion one can draw is typically weaker. One kind of such weak theorem is discussed below. The idea behind the theorem is that if a customer is not willing to trade off a small total decrease in satisfying a group of requirements whose priorities may be different for a larger increase in a requirement $R_{k'}$, at least one requirement in the group must have a higher priority than $R_{k'}$. It is formally stated in the following theorem.

Theorem 4 *Suppose that requirements $R_1, R_2, \ldots, R_n$ are aggregated by the weighted arithmetic average operator. Assuming further that there is a subset G_1 of p requirements R_{k_h}, $1 \leq h \leq p$ and $1 \leq k_h \leq n$. Let j_1 and j_2 be two solutions for case i, δ_k denote the difference between the degrees R_k is satisfied by the two solutions, i.e., $\delta_k = d_{i,j_2,k} - d_{i,j_1,k}$ and δ_{G_1} denote the difference between the total degree group G_1 is satisfied by the two solutions, i.e., $\delta_{G_1} = \sum_{h=1}^{p} \delta_{k_h}$. If (1) solution j_1 is preferred to solution j_2, i.e., $r_{j_1} < r_{j_2}$, (2) The solution j_2 improves the degree $R_{k'}$ is satisfied, i.e., $\delta_{k'} > 0$ where $1 \leq k' \leq n$, (3) The solution j_2 decreases the degree every requirement in the group G_1 is satisfied, i.e., $\delta_{k_h} < 0$ where $1 \leq h \leq p$, (4) The amount of decrease on the total degree the group G_1 is satisfied is smaller than the amount of increase on the degree the requirement $R_{k'}$ is satisfied, i.e., $\delta_{k'} > \|\delta_{G_1}\|$, (5) There is no change in the degree every other requirement is satisfied, i.e., $\delta_k = 0$ for $k \neq k', k_h$ and $1 \leq h \leq p$. Then at least one requirement in the group G_1 must have a higher priority than $R_{k'}$, i.e.,*

$$w_{k'} < \max_{h=1}^{p} w_{k_h}.$$

Proof: Let $D_{i,j} = \sum_{k=1}^{n} w_k \times d_{i,j,k}$. Since $r_{i,j_1} < r_{i,j_2}$, we have $D_{i,j_1} > D_{i,j_2}$. That is,

$$\sum_{k=1}^{n} w_k \times d_{i,j_1,k} > \sum_{k=1}^{n} w_k \times d_{i,j_2,k}.$$

Thus,

$$\sum_{k=1}^{n} w_k \times \delta_k < 0.$$

Since $\delta_k = 0$ for all $k \neq k', k_h$ and $1 \leq h \leq p$, from condition 5 of the theorem, we have

$$w_{k'} \times \delta_{k'} + \sum_{h=1}^{p} (w_{k_h} \times \delta_{k_h}) < 0.$$

That is,

$$w_{k'} \times \delta_{k'} < -\sum_{h=1}^{p} (w_{k_h} \times \delta_{k_h}).$$

Since for all $1 \leq h \leq p$, $\delta_{k_h} < 0$ from condition 3, we have

$$w_{k'} \times \delta_{k'} < \sum_{h=1}^{p} (w_{k_h} \times \|\delta_{k_h}\|).$$

Since all weights w_{k_h} are non-negative, we have

$$w_{k'} \times \delta_{k'} < \max_{h=1}^{p} w_{k_h} \times (\sum_{h=1}^{p} \|\delta_{k_h}\|).$$

Since for all $1 \leq h \leq p$, $\delta_{k_h} < 0$ from condition 3, we arrive at

$$w_{k'} \times \delta_{k'} < \max_{h=1}^{p} w_{k_h} \times \|\delta_{G_1}\|.$$

In addition, since $\delta_{k'} > \|\delta_{G_1}\|$ from condition 4, we have

$$\frac{w_{k'}}{\max_{h=1}^{p} w_{k_h}} < \frac{\|\delta_{G_1}\|}{\delta_{k'}} < 1.$$

That is,

$$w_{k'} < \max_{h=1}^{p} w_{k_h}.$$

□

4 Discussion

We have presented several theorems that can help both the requirement analyst and the customer to develop a better understanding about the priority of requirements. However, there are at least two issues that may limit the practical application of the formal methodology. First, the conditions stated in the theorems are not always satisfied. Second, the assignment of degree of satisfaction in the case analysis may not always be consistent. We would like to discuss approaches for addressing these two issues below.

The first issue can be addressed by developing heuristics that relax certain conditions in the theorems. For instance, the condition that all requirements in a group should have **the same priority** in Theorems 2 and 3 can be relaxed to one that only requires the priorities of requirements in a group to be **close**, rather than identical. The strength of a conclusion made by such heuristics can depend on how close these priorities are based on fuzzy logic reasoning or probabilistic reasoning. Similarly, we can obtain another heuristic rule by relaxing the condition that all requirement subgroups, except for two, have **the same** total degree of satisfaction for two alternative solutions (i.e., condition 5 in Theorem 4) into one that only requires the difference to be **small**. These heuristics can draw plausible conclusions, which can be combined using existing reasoning under uncertainty schemes. Such a reasoning under uncertainty approach to identifying importance relationship between requirements is more general than the one purely based on theorems described in the previous section, because each theorem can be viewed as a special case of a heuristic rule. However, the cost-effectiveness of such an approach remains to be shown.

The second issue regarding the difficulty in obtaining consistent degree of satisfactions from the customer can be addressed by developing techniques for detecting the *inconsistency* in Extended Analytic Matrix. This can be achieved by (1) detecting contradictory conclusions about relative importance of requirements (e.g., R1 is more important than R2, R2 is more important than R3, and R3 is more important than R1), or by (2) identifying inferred relative importance relationships that are counter-intuitive to the customer (e.g., The case analysis indicates that R1 is more important than R2, but the customer disagrees). Whenever an inconsistency is detected, the customer can revise satisfaction measures, or, even though unlikely, to change preference ranking. This cycle of inconsistency detection and refinement of the Extended Analytic Matrix can be continued until no inconsistency is found.

5 Conclusion

We have presented a formal approach for the specification and analysis of imprecise conflicting functional requirements. The methodology extends the fundamental notion of desired state transitions into a fuzzy set. By capturing the elasticity of the requirement, the proposed approach provides a sound theoretical foundation for the analysis of the priority based on analyzing the trade-off between partial satisfaction of imprecise conflicting requirements. It also provides a practical method to help analysts to assess the priority of requirements with complex relationships among them.

6 Acknowledgements

We wish to thank Professor Lotfi A. Zadeh for suggesting that we use fuzzy multi-criteria decision techniques to aggregate soft requirements. We also thank Dr. Jonathan Lee for his early work in this research and Dr. Swee Hor Teh for providing us the ISPBEX example. This research is supported by NSF Young Investigator Award IRI-9257293.

References

[1] R. Balzer, N. Goldman, and D. Wile, "Informality in program specifications", *IEEE Transaction on Software Enginering*, 4(2):94–103, 1978.

[2] Sergio Cardenas and Marvin V. Zelkowite, "Evaluation criteria for functional specification", *Proc. of the 12th IEEE International Conference on Software Engineering*, 26-33, 1990.

[3] J. Conklin and M. Begeman, "gIBIS: A hypertext tool for exploratory policy discussion", *Proc. of the Conference on Computer Support for Cooperative Work*, Portland, Oregon, 1988.

[4] R. O. Flamm, *et. al*, "The Integrated Southern Pine Beetle Expert System: ISPBEX", *Journal of Expert Systems with Applications*, Vol. 2, 97-105, 1991.

[5] J.V. Guttag, J.J. Horning, and J.M. Wing, "The larch family of specification languages", *IEEE Software*, 2(5):24–36, 1985.

[6] Martin S. Feather, "Constructing specifications by combining parallel elaborations", *IEEE Transactions on Software Engineering*, 15(2):198-208, 1989.

[7] Pei Hsia, Alan Davis and David Kung, "Status report: requirements engineering", *IEEE Software*, 75-79, November, 1993.

[8] W. Lewis Johnson, Martin S. Feather and David R. Harris, "The KBSA requirements/Specification Facet: ARIES", *Proc. of the 6th Annual Knowledge-Based Software Engineering Conference*, Syracuse, New York, Sept. 22-25, 1991.

[9] Pei Hsia, *et al.* "Formal Approach to Scenario Analysis", *IEEE Software*, 33-41, March, 1994.

[10] M. Jarke, *et al.*, "DAIDA: An Enviroment for envolving information systems", *ACM Transaction on Information Systems*, 10(1), 1992.

[11] John Mylopoulos, Lawrence Chung, and Brian Nixon, "Representing and using nonfunctional requirements: a process-oriented approach", *IEEE Transaction on Software Engineering*, Vol. 18, No. 6, June, 1992.

[12] David Lorge Parnes, "Predicate logic for software engineering", *IEEE Transactions on Software Engineering*, 19(9):856-862 1993.

[13] H.B. Reubenstein and R.C. Waters, "The requirements apprentice: Automated assistance for requirements acquisition", *IEEE Transaction on Software Engineering*, 17(3), 1991.

[14] B. Ramesh and V. Dhar, "Support Systems Development by Capturing Deliberations During Requirements Engineering", *IEEE Transaction on Software Engineering*, 18(6), 1992.

[15] W. N. Robinson, "Negotiation behavior during requirement specification", *Proc. of the 12th International Conference on Software Engineering*, 268-276, 1990.

[16] W. N. Robinson and S. Fickas, "Supporting multi-perspective requirements engineering", *Proc. of the first international conference on requirements engineering*, 18-22, 1994.

[17] Gruia-Catalin Roman, "A Taxonomy of current issues in requirements engineering", *IEEE Computer*, 14-21, April, 1985.

[18] J.M. Spivey, The Z-Notation: a Reference Manual, Prentice-Hall, Englewood Cliffs, NJ, 1987.

[19] John Yen and J. Lee, "A task-based methodology for specifying expert systems", *IEEE Expert*, Vol. 8, No. 1, Pages 8-15, Feb. 1993.

[20] J. Yen and J. Lee, "Fuzzy Logic as a Basis for Specifying Imprecise Requirements", *Proc. of the IEEE International Conference on Fuzzy Systems*, 745-749, 1993.

[21] John Yen, Xiaoqing Liu and Swee Hor Teh, "The Acquisition, analysis and evaluation of imprecise requirements for knowledge-based systems", *Proc. of the Twelfth National Conference on Artificial Intelligence Conference (AAAI-94)*, Seattle, Washington, 1994.

[22] Ronald R. Yager, "Connectives and quantifiers in fuzzy sets", *Fuzzy Sets and Systems*, 40(1991) 39-75.

[23] L.A. Zadeh, "Test-score semantics as a basis for a computational approach to the representation of meaning", *Literacy Linguistic Computing*, 1:24–35, 1986.

[24] H.-J. Zimmermann, *Fuzzy set theory and its applications*, Kluwer Academic, MA, 1991.