Haibin Wang
Florentin Smarandache
Yan-Qing Zhang
Rajshekhar Sunderraman

INTERVAL NEUTROSOPHIC SETS AND LOGIC:
## Theory and Applications in Computing

Haibin Wang
Florentin Smarandache
Yan-Qing Zhang
Rajshekhar Sunderraman

Interval Neutrosophic Sets and Logic:
Theory and Applications in Computing

Neutrosophication

truth-membership
function

Input

indeterminacy-
membership
function

Neutrosophic
Inference

Neutrosophic
Type Reduction

Deneutrosophication

Crisp Output

falsity-membership
function

Neutrosophic
Rule Base

Interval Neutrosophic Sets and Logic: Theory and Applications in Computing

Haibin Wang[1]
Florentin Smarandache[2]
Yan-Qing Zhang[1]
Rajshekhar Sunderraman[1]

[1]Department of Computer Science
Georgia State University
Atlanta, GA 30302, USA

[2]Department of Mathematics and Science
University of New Mexico
Gallup, NM 87301, USA

Cover design by Xiaoyuan Suo.

Many books can be downloaded from the following E-Library of Science:
http://www.gallup.unm.edu/~smarandache/eBooks-otherformats.htm

Contents

Preface

This book presents the advancements and applications of neutrosophics. Chapter 1 first introduces the interval neutrosophic sets which is an instance of neutrosophic sets. In this chapter, the definition of interval neutrosophic sets and set-theoretic operators are given and various properties of interval neutrosophic set are proved. Chapter 2 defines the interval neutrosophic logic based on interval neutrosophic sets including the syntax and semantics of first order interval neutrosophic propositional logic and first order interval neutrosophic predicate logic. The interval neutrosophic logic can reason and model fuzzy, incomplete and inconsistent information. In this chapter, we also design an interval neutrosophic inference system based on first order interval neutrosophic predicate logic. The interval neutrosophic inference system can be applied to decision making. Chapter 3 gives one application of interval neutrosophic sets and logic in the field of relational databases. Neutrosophic data model is the generalization of fuzzy data model and paraconsistent data model. Here, we generalize various set-theoretic and relation-theoretic operations of fuzzy data model to neutrosophic data model. Chapter 4 gives another application of interval neutrosophic logic. A soft semantic Web Services agent framework is proposed to faciliate the registration and discovery of high quality semantic Web Services agent. The intelligent inference engine module of soft Semantic Web Services agent is implemented using interval neutrosophic logic.

The neutrosophic logic, neutrosophic set, neutrosophic probability, and neutrosophic statistics are increasingly used in engineering applications (especially for software and information fusion), medicine, military, cybernetics, physics.

In order to familiarize the reader with the concepts of neutrosophics here below we present their definitions and short explanations, followed by new research subjects in the field.

Neutrosophic Logic is a general framework for unification of many existing logics. The main idea of NL is to characterize each logical statement in a 3D Neutrosophic Space, where each dimension of the space represents respectively the truth (T), the falsehood (F), and the indeterminacy (I) of the statement under consideration, where T, I, F are standard or non-standard real subsets of $]0^-, 1^+[$.

For software engineering proposals the classical unit interval $[0, 1]$ can be used. T, I, F are independent components, leaving room for incomplete information (when their superior sum < 1), paraconsistent and contradictory information (when the superior sum > 1), or complete information (sum of components $= 1$).

As an example: a statement can be between $[0.4, 0.6]$ true, 0.1 or between $(0.15, 0.25)$ indeterminate, and either 0.4 or 0.6 false.

Neutrosophic Set.

Let U be a universe of discourse, and M a set included in U. An element x from U is noted with respect to the set M as $x(T, I, F)$ and belongs to M in the following way: it is $t\%$ true in the set, $i\%$ indeterminate (unknown if it is) in the set, and $f\%$ false, where t varies in T, i varies in I, f varies in F. Statically T, I, F are subsets, but dynamically T, I, F are functions/operators depending on many known or unknown parameters.

Neutrosophic Probability is a generalization of the classical probability and imprecise probability in which the chance that an event A occurs is $t\%$ true - where t varies in the subset T, $i\%$ indeterminate - where i varies in the subset I, and $f\%$ false - where f varies in the subset F. In classical probability $n_s up \leq 1$, while in neutrosophic probability $n_s up \leq 3^+$. In imprecise probability: the probability of an event is a subset T in $[0, 1]$, not a number p in $[0, 1]$, what's left is supposed to be the opposite, subset F (also from the unit interval $[0, 1]$); there is no indeterminate subset I in imprecise probability.

Neutrosophic Statistics is the analysis of events described by the neutrosophic probability. The function that models the neutrosophic probability of a random variable x is called neutrosophic distribution: $NP(x) = (T(x), I(x), F(x))$, where $T(x)$ represents the probability that value x occurs, $F(x)$ represents the probability that value x does not occur, and $I(x)$ represents the indeterminate / unknown probability of value x.

Neutrosophy is a new branch of philosophy that studies the origin, nature, and scope of neutralities, as well as their interactions with different ideational spectra. The neutrosophics were introduced by Dr. F. Smarandache in 1995. This theory considers every notion or idea $\langle A \rangle$ together with its opposite or negation $\langle Anti - A \rangle$ and the spectrum of "neutralities" $\langle Neut - A \rangle$ (i.e. notions or ideas located between the two extremes, supporting neither $\langle A \rangle$ nor $\langle Anti - A \rangle$). The $\langle Neut - A \rangle$ and $\langle Anti - A \rangle$ ideas together are referred to as $\rangle Non - A \rangle$. According to this theory every idea $\langle A \rangle$ tends to be neutralized and balanced by $\langle Anti - A \rangle$ and $\langle Non - A \rangle$ ideas - as a state of equilibrium.

In a classical way $\langle A \rangle, \langle Neut - A \rangle, \langle Anti - A \rangle$ are disjoint two by two. But, since in many cases the borders between notions are vague, imprecise, it is possible that $\langle A \rangle, \langle Neut - A \rangle, \langle Anti - A \rangle$ (and $\langle Non - A \rangle$ of course) have common parts two by two as well.

Neutrosophy is the base of neutrosophic logic, neutrosophic set, neutrosophic probability and statistics.

The **NEUTROSOPHIC WORLD** is expanding with new research subjects:

1. Neutrosophic topology including neutrosophic metric spaces and smooth topological spaces

2. Neutrosophic numbers and arithmetical operations, including ranking procedures for neutrosophic numbers

3. Neutrosophic rough sets

4. Neutrosophic relational structures, including neutrosophic relational equations, neutrosophic similarity relations, and neutrosophic orderings,

5. Neutrosophic geometry

6. Neutrosophic probability

7. Neutrosophic logical operations, including n-norms, n-conorms, neutrosophic implicators, neutrosophic quantifiers

8. Measures of neutrosophication

9. Deneutrosophication techniques

10. Neutrosophic measures and neutrosophic integrals

11. Neutrosophic multivalued mappings

12. Neutrosophic differential calculus

13. Neutrosophic mathematical morphology

14. Neutrosophic algebraic structures

15. Neutrosophic models

16. Neutrosophic cognitive maps

17. Neutrosophic matrix

18. Neutrosophic graph

19. Neutrosophic fusion rules

20. Neutrosophic relational maps

21. Applications:

   neutrosophic relational databases, neutrosophic image processing, neutrosophic linguistic variables, neutrosophic decision making and preference structures, neutrosophic expert systems, neutrosophic reliability theory, neutrosophic soft computing techniques in e-commerce and e-learning

Readers can download e-books on neutrosophic logic (a generalization of the fuzzy logic and IFL), neutrosophic set (generalization of the fuzzy set and IFS), neutrosophic probability (generalization of classical and imprecise probabilities) from this site: www.gallup.unm.edu/∼smarandache/philos.htm.

Researchers interested in neutrosophics can contact Dr. Florentin Smarandache, University of New Mexico, Department of Mathematics and Science, 200 College Road, Gallup, NM 87301, USA, E-mail: smarand@unm.edu.

May 2005

Haibin Wang
Florentin Smarandache
Yan-Qing Zhang
Rajshekhar Sunderraman

An Ode to Fuzzy & Neutrosophic Logic and their Creators

When everything seems a murky mess
and you are forced to second guess
The way you are headed when you're going straight
And whether you're there on time 'cause early may be late!
When your eyes start playing tricks - it's neither night nor day
But the magic hour; when you just can't for sure say
The white from the black as mostly all is grey
Take a moment to close your eyes and thank Zadeh!
For inventing a way to tell black from white and the big from the small
When none knows for sure how black is black or how tall is tall!
'Cause when chance becomes a possibility, you know you sure can bet
That you're in one group or the other, 'cause you're in a fuzzy set!

When you are in a pensive mood having a hard think
On deep issues like the origin of Man and why there is a missing link!
Whether the egg came before the hen like the dough came before the bread
Whether atoms are really balls of matter, whether Schrödinger's cat is dead!
Whether bulls will keep tossing the markets up while the big bears hibernate
And all these answers may be True or False only if they aren't Indeterminate!
It is not always easy to take a side as will become apparent
But if you will still prefer to have a say over remaining silent
Then you'll have to make a choice based on the information at hand
And it won't be easy as you'll feel as lost as Alice in wonderland!
Arguments will be as strong for one side as for any other
And just when you feel too weighed down to even stop and bother;
In will come Smarandache waving his wand of neutrosophic logic
And suddenly all will make perfect sense - like a spell of magic!
Fusing information into knowledge is what neutrosophy aims to do
Does not matter if information is partial, incoherent and paradoxical too!
What Zadeh started and Smarandache perfected; has now come here to stay
Helping us to find order in chaos like the proverbial needle in a stack of hay!
Generalizing the rules of Probability into a philosophy quite brand new
A new branch of science has emerged that has detractors but very few!

Sukanto Bhattacharya
Alaska Pacific University
Anchorage, AK 99508, USA

Chapter 1

Interval Neutrosophic Sets

A neutrosophic set is a part of neutrosophy that studies the origin, nature, and scope of neutralities, as well as their interactions with different ideational spectra. The neutrosophic set is a powerful general formal framework that has been recently proposed. However, the neutrosophic set needs to be specified from a technical point of view. Now we define the set-theoretic operators on an instance of a neutrosophic set, and call it an Interval Neutrosophic Set (INS). We prove various properties of INS, which are connected to operations and relations over INS. Finally, we introduce and prove the convexity of interval neutrosophic sets.

1.1   Introduction

The concept of fuzzy sets was introduced by Zadeh in 1965 [Zad65b]. Since then fuzzy sets and fuzzy logic have been applied in many real applications to handle uncertainty. The traditional fuzzy set uses one real number $\mu_A(x) \in [0, 1]$ to represent the grade of membership of fuzzy set A defined on universe X. Sometimes $\mu_A(x)$ itself is uncertain and hard to be defined by a crisp value. So the concept of interval valued fuzzy sets was proposed [Tur86] to capture the uncertainty of grade of membership. Interval valued fuzzy set uses an interval value $[\mu_A^L(x), \mu_A^U(x)]$ with $0 \le \mu_A^L(x) \le \mu_A^U(x) \le 1$ to represent the grade of membership of fuzzy set A. In some applications such as expert system, belief system and information fusion, we should consider not only the truth-membership supported by the evidence but also the falsity-membership against by the evidence. That is beyond the scope of fuzzy sets and interval valued fuzzy sets. In 1986, Atanassov introduced the intuitionistic fuzzy sets [Ata86] that is a generalization of fuzzy sets and provably equivalent to interval valued fuzzy sets. The intuitionistic fuzzy sets consider both truth-membership and falsity-membership. Later on, intuitionistic fuzzy sets were extended to the interval valued intuitionistic fuzzy sets [Ata89]. The interval valued intuitionistic fuzzy set uses a pair of intervals $[t^-, t^+]$, $0 \le t^- \le t^+ \le 1$ and $[f^-, f^+]$, $0 \le f^- \le f^+ \le 1$ with $t^+ + f^+ \le 1$ to describe the degree of true belief and false belief. Because of the restriction that $t^+ + f^+ \le 1$, intuitionistic fuzzy sets and interval valued intuitionistic fuzzy sets can only handle incomplete information not the indeterminate information and inconsistent information which exists commonly in belief systems. For example, when we ask the

opinion of an expert about certain statement, he or she may say that the possibility that the statement is true is between 0.5 and 0.7 and the statement is false is between 0.2 and 0.4 and the degree that he or she is not sure is between 0.1 and 0.3. Here is another example, suppose there are 10 voters during a voting process. In time t_1, three vote "yes", two vote "no" and five are undecided, using neutrosophic notation, it can be expressed as $x(0.3, 0.5, 0.2)$. In time t_2, three vote "yes", two vote "no", two give up and three are undecided, it then can be expressed as $x(0.3, 0.3, 02)$. That is beyond the scope of the intuitionistic fuzzy set. So, the notion of neutrosophic set is more general and overcomes the aforementioned issues.

In neutrosophic set, indeterminacy is quantified explicitly and truth-membership, indeterminacy-membership and falsity-membership are independent. This assumption is very important in many applications such as information fusion in which we try to combine the data from different sensors. Neutrosophy was introduced by Florentin Smarandache in 1980. "It is a branch of philosophy which studies the origin, nature and scope of neutralities, as well as their interactions with different ideational spectra" [Sma99]. Neutrosophic set is a powerful general formal framework which generalizes the concept of the classic set, fuzzy set [Zad65b], interval valued fuzzy set [Tur86], intuitionistic fuzzy set [Ata86], interval valued intuitionistic fuzzy set [Ata89], paraconsistent set [Sma99], dialetheist set [Sma99], paradoxist set [Sma99], tautological set [Sma99]. A neutrosophic set A defined on universe U. $x = x(T, I, F) \in A$ with T, I and F being the real standard or non-standard subsets of $]0^-, 1^+[$. T is the degree of truth-membership function in the set A, I is the degree of indeterminacy-membership function in the set A and F is the degree of falsity-membership function in the set A.

The neutrosophic set generalizes the above mentioned sets from philosophical point of view. From scientific or engineering point of view, the neutrosophic set and set-theoretic operators need to be specified. Otherwise, it will be difficult to apply in the real applications. In this chapter, we define the set-theoretic operators on an instance of neutrosophic set called Interval Neutrosophic Set (INS). We call it as "interval" because it is subclass of neutrosophic set, that is we only consider the subunitary interval of $[0, 1]$.

An interval neutrosophic set A defined on universe X, $x = x(T, I, F) \in A$ with T, I and F being the subinterval of $[0, 1]$. The interval neutrosophic set can represent uncertain, imprecise, incomplete and inconsistent information which exist in real world. The interval neutrosophic set generalizes the following sets:

1. the *classical set*, $I = \emptyset$, $\inf T = \sup T = 0$ or 1, $\inf F = \sup F = 0$ or 1 and $\sup T + \sup F = 1$.

2. the *fuzzy set*, $I = \emptyset$, $\inf T = \sup T \in [0, 1]$, $\inf F = \sup F \in [0, 1]$ and $\sup T + \sup F = 1$.

3. the *interval valued fuzzy set*, $I = \emptyset$, $\inf T, \sup T, \inf F, \sup F \in [0, 1]$, $\sup T + \inf F = 1$ and $\inf T + \sup F = 1$.

4. the *intuitionistic fuzzy set*, $I = \emptyset$, $\inf T = \sup T \in [0, 1]$, $\inf F = \sup F \in [0, 1]$ and $\sup T + \sup F \leq 1$.

5. the *interval valued intuitionistic fuzzy set*, $I = \emptyset$, $\inf T, \sup T, \inf F, \sup F \in [0, 1]$ and $\sup T + \sup F \leq 1$.

6. the *paraconsistent set*, $I = \emptyset$, $\inf T = \sup T \in [0, 1]$, $\inf F = \sup F \in [0, 1]$ and $\sup T + \sup F > 1$.

7. the *interval valued paraconsistent set*, $I = \emptyset$, $\inf T, \sup T, \inf F, \sup F \in [0,1]$ and $\inf T + \inf F > 1$.

The relationship among interval neutrosophic set and other sets is illustrated in Fig 1.1.
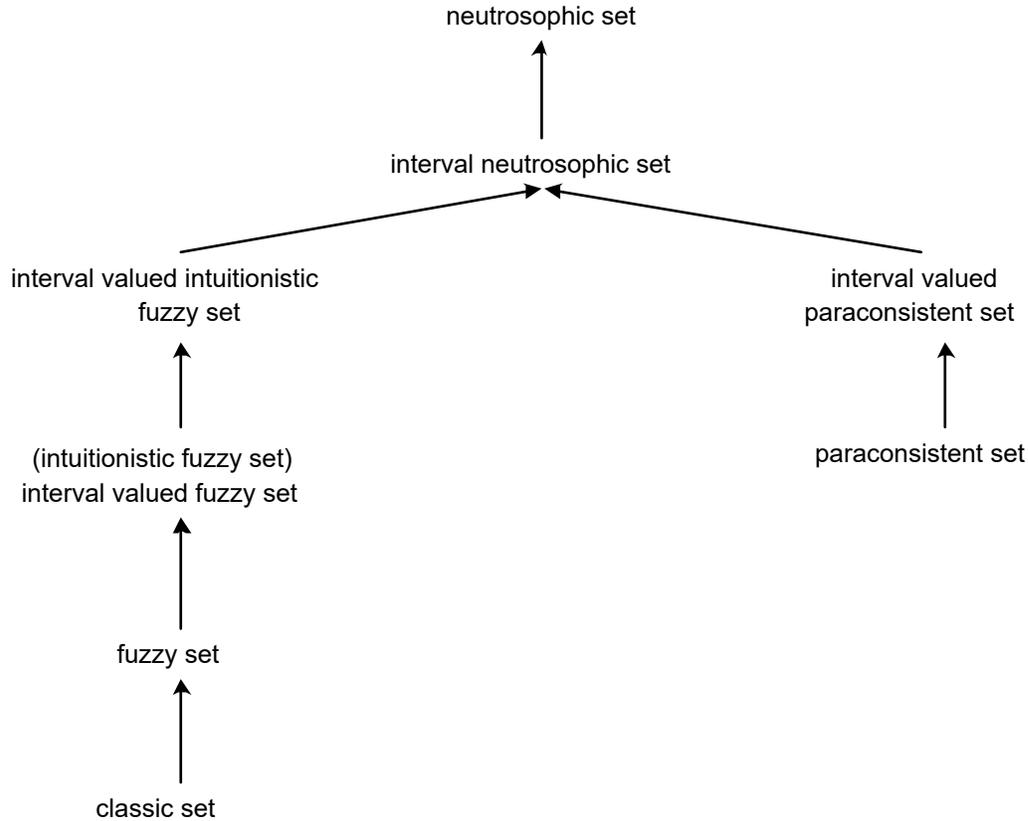
Figure 1.1: Relationship among interval neutrosophic set and other sets

Note that $\rightarrow$ in Fig. 1.1 such as $a \rightarrow b$ means that b is a generalization of a.

We define the set-theoretic operators on the interval neutrosophic set. Various properties of INS are proved, which are connected to the operations and relations over INS.

The rest of chapter is organized as follows. Section 1.2 gives a brief overview of the neutrosophic set. Section 1.3 gives the definition of the interval neutrosophic set and set-theoretic operations. Section 1.4 presents some properties of set-theoretic operations. Section 1.5 defines the convexity of the interval neutrosophic sets and proves some properties of convexity. Section 1.6 concludes the chapter. To maintain a smooth flow throughout the chapter, we present the proofs to all theorems in Appendix.

1.2   Neutrosophic Set

This section gives a brief overview of concepts of neutrosophic set defined in [Sma99]. Here, we use different notations to express the same meaning. Let S_1 and S_2 be two real standard or non-standard subsets, then $S_1 \oplus S_2 = \{x | x = s_1 + s_2, s_1 \in S_1 \text{ and } s_2 \in S_2\}$, $\{1^+\} \oplus S_2 = \{x | x = 1^+ + s_2, s_2 \in S_2\}$. $S_1 \ominus S_2 = \{x | x = s_1 - s_2, s_1 \in S_1 \text{ and } s_2 \in S_2\}$, $\{1^+\} \ominus S_2 = \{x | x = 1^+ - s_2, s_2 \in S_2\}$. $S_1 \odot S_2 = \{x | x = s_1 \cdot s_2, s_1 \in S_1 \text{ and } s_2 \in S_2\}$.

Definition 1 (Neutrosophic Set) *Let X be a space of points (objects), with a generic element in X denoted by x.*
A neutrosophic set A in X is characterized by a truth-membership function T_A, *a* indeterminacy-membership function I_A *and a* falsity-membership *function F_A. $T_A(x), I_A(x)$ and $F_A(x)$ are real standard or non-standard subsets of $]0^-, 1^+[$. That is*

$$
\begin{aligned}
T_A : X &\rightarrow]0^-, 1^+[, & (1.1)\\
I_A : X &\rightarrow]0^-, 1^+[, & (1.2)\\
F_A : X &\rightarrow]0^-, 1^+[. & (1.3)
\end{aligned}
$$

There is no restriction on the sum of $T_A(x)$, $I_A(x)$ and $F_A(x)$, so $0^- \leq \sup T_A(x) + \sup I_A(x) + \sup F_A(x) \leq 3^+$.

Definition 2 *The* complement *of a neutrosophic set A is denoted by $\bar{A}$ and is defined by*

$$
\begin{aligned}
T_{\bar{A}}(x) &= \{1^+\} \ominus T_A(x), & (1.4)\\
I_{\bar{A}}(x) &= \{1^+\} \ominus I_A(x), & (1.5)\\
F_{\bar{A}}(x) &= \{1^+\} \ominus F_A(x), & (1.6)
\end{aligned}
$$

for all x in X.

Definition 3 (Containment) *A neutrosophic set A is* contained *in the other neutrosophic set B, $A \subseteq B$, if and only if*

$$
\begin{aligned}
\inf T_A(x) \leq \inf T_B(x) \quad &, \quad \sup T_A(x) \leq \sup T_B(x), & (1.7)\\
\inf I_A(x) \geq \inf I_B(x) \quad &, \quad \sup I_A(x) \geq \sup I_B(x), & (1.8)\\
\inf F_A(x) \geq \inf F_B(x) \quad &, \quad \sup F_A(x) \geq \sup F_B(x), & (1.9)
\end{aligned}
$$

for all x in X.

Definition 4 (Union) *The* union *of two neutrosophic sets A and B is a neutrosophic set C, written as $C = A \cup B$, whose truth-membership, indeterminacy-membership and falsity-membership functions are related to those of A and B by*

$$
\begin{align}
T_C(x) &= T_A(x) \oplus T_B(x) \ominus T_A(x) \odot T_B(x), \tag{1.10} \\
I_C(x) &= I_A(x) \oplus I_B(x) \ominus I_A(x) \odot I_B(x), \tag{1.11} \\
F_C(x) &= F_A(x) \oplus F_B(x) \ominus F_A(x) \odot F_B(x), \tag{1.12}
\end{align}
$$

for all x in X.

Definition 5 (Intersection) *The* intersection *of two neutrosophic sets A and B is a neutrosophic set C, written as $C = A \cap B$, whose truth-membership, indeterminacy-membership and falsity-membership functions are related to those of A and B by*

$$
\begin{align}
T_C(x) &= T_A(x) \odot T_B(x), \tag{1.13} \\
I_C(x) &= I_A(x) \odot I_B(x), \tag{1.14} \\
F_C(x) &= F_A(x) \odot F_B(x), \tag{1.15}
\end{align}
$$

for all x in X.

Definition 6 (Difference) *The* difference *of two neutrosophic sets A and B is a neutrosophic set C, written as $C = A \setminus B$, whose truth-membership, indeterminacy-membership and falsity-membership functions are related to those of A and B by*

$$
\begin{align}
T_C(x) &= T_A(x) \ominus T_A(x) \odot T_B(x), \tag{1.16} \\
I_C(x) &= I_A(x) \ominus I_A(x) \odot I_B(x), \tag{1.17} \\
F_C(x) &= F_A(x) \ominus F_A(x) \odot F_B(x), \tag{1.18}
\end{align}
$$

for all x in X.

Definition 7 (Cartesian Product) *Let A be the neutrosophic set defined on universe E_1 and B be the neutrosophic set defined on universe E_2. If $x(T_A^1, I_A^1, F_A^1) \in A$ and $y(T_A^2, I_A^2, F_A^2) \in B$, then the* cartesian product *of two neutrosophic sets A and B is defined by*

$$
(x(T_A^1, I_A^1, F_A^1), y(T_A^2, I_A^2, F_A^2)) \in A \times B \tag{1.19}
$$

1.3   Interval Neutrosophic Set

In this section, we present the notion of the *interval neutrosophic set* (INS). The interval neutrosophic set (INS) is an instance of neutrosophic set which can be used in real scientific and engineering applications.

Definition 8 (Interval Neutrosophic Set) *Let X be a space of points (objects), with a generic element in X denoted by x.*
An interval neutrosophic set (INS) A in X is characterized by truth-membership function T_A, indeterminacy-membership function I_A and falsity-membership function F_A. For each point x in X, $T_A(x), I_A(x), F_A(x) \subseteq [0, 1]$.

An interval neutrosophic set (INS) in R^1 is illustrated in Fig. 1.2.
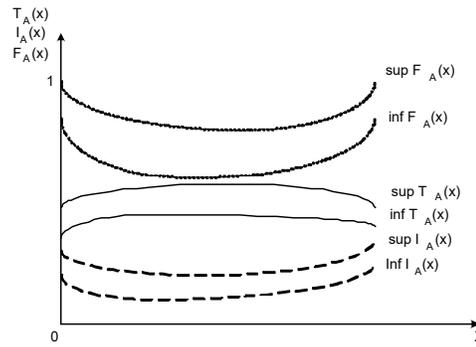
Figure 1.2: Illustration of interval neutrosophic set in R^1

When X is continuous, an INS A can be written as

$$A = \int_X \langle T(x), I(x), F(x) \rangle / x, \ x \in X \tag{1.20}$$

When X is discrete, an INS A can be written as

$$A = \sum_{i=1}^n \langle T(x_i), I(x_i), F(x_i) \rangle / x_i, \ x_i \in X \tag{1.21}$$

Consider parameters such as capability, trustworthiness and price of semantic Web services. These parameters are commonly used to define quality of service of semantic Web services. In this section, we will use the evaluation of quality of service of semantic Web services [WZS04] as running example to illustrate every set-theoretic operation on interval neutrosophic set.

Example 1 *Assume that $X = [x_1, x_2, x_3]$. x_1 is capability, x_2 is trustworthiness and x_3 is price. The values of x_1, x_2 and x_3 are in $[0, 1]$. They are obtained from the questionnaire of some domain experts, their*

option could be degree of good, degree of indeterminacy and degree of poor. A is an interval neutrosophic set of X defined by
$A = \langle [0.2, 0.4], [0.3, 0.5], [0.3, 0.5] \rangle / x_1 + \langle [0.5, 0.7], [0, 0.2], [0.2, 0.3] \rangle / x_2 + \langle [0.6, 0.8], [0.2, 0.3], [0.2, 0.3] \rangle / x_3.$
B is an interval neutrosophic set of X defined by
$B = \langle [0.5, 0.7], [0.1, 0.3], [0.1, 0.3] \rangle / x_1 + \langle [0.2, 0.3], [0.2, 0.4], [0.5, 0.8] \rangle / x_2 + \langle [0.4, 0.6], [0, 0.1], [0.3, 0.4] \rangle / x_3.$

Definition 9 *An interval neutrosophic set A is* empty *if and only if its* $\inf T_A(x) = \sup T_A(x) = 0$, $\inf I_A(x) = \sup I_A(x) = 1$ *and* $\inf F_A(x) = \sup T_A(x) = 0$, *for all x in X.*

We now present the set-theoretic operators on interval neutrosophic set.

Definition 10 (Containment) *An interval neutrosophic set A is* contained *in the other interval neutrosophic set B, $A \subseteq B$, if and only if*

$$\inf T_A(x) \leq \inf T_B(x) \quad , \quad \sup T_A(x) \leq \sup T_B(x), \tag{1.22}$$
$$\inf I_A(x) \geq \inf I_B(x) \quad , \quad \sup I_A(x) \geq \sup I_B(x), \tag{1.23}$$
$$\inf F_A(x) \geq \inf F_B(x) \quad , \quad \sup F_A(x) \geq \sup F_B(x), \tag{1.24}$$

for all x in X.

Definition 11 *Two interval neutrosophic sets A and B are* equal, *written as $A = B$, if and only if $A \subseteq B$ and $B \subseteq A$*

Let $\underline{0} = \langle 0, 1, 1 \rangle$ and $\underline{1} = \langle 1, 0, 0 \rangle$.

Definition 12 (Complement) *Let C_N denote a neutrosophic* complement *of A. Then C_N is a function*

$$C_N : N \rightarrow N$$

and C_N must satisfy at least the following three axiomatic requirements:

1. *$C_N(\underline{0}) = \underline{1}$ and $C_N(\underline{1}) = \underline{0}$ (boundary conditions).*

2. *Let A and B be two interval neutrosophic sets defined on X, if $A(x) \leq B(x)$, then $C_N(A(x)) \geq C_N(B(x))$, for all x in X. (monotonicity).*

3. *Let A be an interval neutrosophic set defined on X, then $C_N(C_N(A(x))) = A(x)$, for all x in X. (involutivity).*

$\square$

There are many functions which satisfy the requirement to be the complement operator of interval neutrosophic sets. Here we give one example.

Definition 13 (Complement C_{N_1}) *The complement of an interval neutrosophic set A is denoted by $\bar{A}$ and is defined by*

$$
\begin{aligned}
T_{\bar{A}}(x) &= F_A(x), & (1.25)\\
\inf I_{\bar{A}}(x) &= 1 - \sup I_A(x), & (1.26)\\
\sup I_{\bar{A}}(x) &= 1 - \inf I_A(x), & (1.27)\\
F_{\bar{A}}(x) &= T_A(x), & (1.28)
\end{aligned}
$$

for all x in X.   □

Example 2 *Let A be the interval neutrosophic set defined in Example 1. Then,*
$\bar{A} = \langle [0.3, 0.5], [0.5, 0.7], [0.2, 0.4] \rangle / x_1 + \langle [0.2, 0.3], [0.8, 1.0], [0.5, 0.7] \rangle / x_2 + \langle [0.2, 0.3], [0.7, 0.8], [0.6, 0.8] \rangle / x_3.$

Definition 14 (N-norm) *Let I_N denote a neutrosophic* intersection *of two interval neutrosophic sets A and B. Then I_N is a function*

$$I_N : N \times N \to N$$

and I_N must satisfy at least the following four axiomatic requirements:

1. *$I_N(A(x), \underline{1}) = A(x)$, for all x in X. (boundary condition).*

2. *$B(x) \leq C(x)$ implies $I_N(A(x), B(x)) \leq I_N(A(x), C(x))$, for all x in X. (monotonicity).*

3. *$I_N(A(x), B(x)) = I_N(B(x), A(x))$, for all x in X. (commutativity).*

4. *$I_N(A(x), I_N(B(x), C(x))) = I_N(I_N(A(x), B(x)), C(x))$, for all x in X. (associativity).*

□

Here we give one example of intersection of two interval neutrosophic sets which satisfies above N-norm axiomatic requirements. Other different definitions can be given for different applications.

Definition 15 (Intersection I_{N_1}) *The intersection of two interval neutrosophic sets A and B is an interval neutrosophic set C, written as $C = A \cap B$, whose truth-membership, indeterminacy-membership, and false-membership are related to those of A and B by*

$$
\begin{aligned}
\inf T_C(x) &= \min(\inf T_A(x), \inf T_B(x)), & (1.29)\\
\sup T_C(x) &= \min(\sup T_A(x), \sup T_B(x)), & (1.30)\\
\inf I_C(x) &= \max(\inf I_A(x), \inf I_B(x)), & (1.31)\\
\sup I_C(x) &= \max(\sup I_A(x), \sup I_B(x)), & (1.32)\\
\inf F_C(x) &= \max(\inf F_A(x), \inf F_B(x)), & (1.33)\\
\sup F_C(x) &= \max(\sup F_A(x), \sup F_B(x)), & (1.34)
\end{aligned}
$$

for all x in X. □

Example 3 *Let A and B be the interval neutrosophic sets defined in Example 1. Then, $A \cap B =$*
$\langle [0.2, 0.4], [0.3, 0.5], [0.3, 0.5] \rangle / x_1 +$
$\langle [0.2, 0.3], [0.2, 0.4], [0.5, 0.8] \rangle / x_2 + \langle [0.4, 0.6], [0.2, 0.3], [0.3, 0.4] \rangle / x_3.$

Theorem 1 *$A \cap B$ is the largest interval neutrosophic set contained in both A and B.*

Definition 16 (N-conorm) *Let U_N denote a neutrosophic union of two interval neutrosophic sets A and B. Then U_N is a function*

$$U_N : N \times N \to N$$

and U_N must satisfy at least the following four axiomatic requirements:

1. *$U_N(A(x), \underline{0}) = A(x)$, for all x in X. (boundary condition).*

2. *$B(x) \leq C(x)$ implies $U_N(A(x), B(x)) \leq U_N(A(x), C(x))$, for all x in X. (monotonicity).*

3. *$U_N(A(x), B(x)) = U_N(B(x), A(x))$, for all x in X. (commutativity).*

4. *$U_N(A(x), U_N(B(x), C(x))) = U_N(U_N(A(x), B(x)), C(x))$, for all x in X. (associativity).*

□

Here we give one example of union of two interval neutrosophic sets which satisfies above N-conorm axiomatic requirements. Other different definitions can be given for different applications.

Definition 17 (Union U_{N_1}) *The union of two interval neutrosophic sets A and B is an interval neutrosophic set C, written as $C = A \cup B$, whose truth-membership, indeterminacy-membership, and false-membership are related to those of A and B by*

$$\begin{aligned}
\inf T_C(x) &= \max(\inf T_A(x), \inf T_B(x)), & (1.35) \\
\sup T_C(x) &= \max(\sup T_A(x), \sup T_B(x)), & (1.36) \\
\inf I_C(x) &= \min(\inf I_A(x), \inf I_B(x)), & (1.37) \\
\sup I_C(x) &= \min(\sup I_A(x), \sup I_B(x)), & (1.38) \\
\inf F_C(x) &= \min(\inf F_A(x), \inf F_B(x)), & (1.39) \\
\sup F_C(x) &= \min(\sup F_A(x), \sup F_B(x)), & (1.40)
\end{aligned}$$

for all x in X. □

Example 4 *Let A and B be the interval neutrosophic sets defined in Example 1. Then, $A \cup B =$*
$\langle [0.5, 0.7], [0.1, 0.3], [0.1, 0.3] \rangle / x_1 +$
$\langle [0.5, 0.7], [0, 0.2], [0.2, 0.3] \rangle / x_2 + \langle [0.6, 0.8], [0, 0.1], [0.2, 0.3] \rangle / x_3$.

The intuition behind the union operator is that if one of elements in A and B is true then it is true in $A \cup B$, only both are indeterminate and false in A and B then it is indeterminate and false in $A \cup B$. The other operators should be understood similarly.

Theorem 2 *$A \cup B$ is the smallest interval neutrosophic set containing both A and B.*

Theorem 3 *Let P be the power set of all interval neutrosophic sets defined in the universe X. Then $\langle P; I_{N_1}, U_{N_1} \rangle$ is a distributive lattice.*

Proof Let A, B, C be the arbitrary interval neutrosophic sets defined on X. It is easy to verify that $A \cap A = A, A \cup A = A$ (idempotency), $A \cap B = B \cap A, A \cup B = B \cup A$ (commutativity), $(A \cap B) \cap C = A \cap (B \cap C), (A \cup B) \cup C = A \cup (B \cup C)$ (associativity), and $A \cap (B \cup C) = (A \cap B) \cup (A \cap C), A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ (distributivity).

Definition 18 (Interval neutrosophic relation) *Let X and Y be two non-empty crisp sets. An interval neutrosophic relation $R(X, Y)$ is a subset of product space $X \times Y$, and is characterized by the truth membership function $T_R(x, y)$, the indeterminacy membership function $I_R(x, y)$, and the falsity membership function $F_R(x, y)$, where $x \in X$ and $y \in Y$ and $T_R(x, y), I_R(x, y), F_R(x, y) \subseteq [0, 1]$.*

Definition 19 (Interval Neutrosophic Composition Functions) *The membership functions for the composition of interval neutrosophic relations $R(X, Y)$ and $S(Y, Z)$ are given by the* interval neutrosophic sup-star composition *of R and S*

$$T_{R \circ S}(x, z) \;\; = \;\; \sup_{y \in Y} \min(T_R(x, y), T_S(y, z)), \tag{1.41}$$

$$I_{R \circ S}(x, z) \;\; = \;\; \sup_{y \in Y} \min(I_R(x, y), I_S(y, z)), \tag{1.42}$$

$$F_{R \circ S}(x, z) \;\; = \;\; \inf_{y \in Y} \max(F_R(x, y), F_S(y, z)). \tag{1.43}$$

If R is an interval neutrosophic set rather than an interval neutrosophic relation, then $Y = X$ and $\sup_{y \in Y} \min(T_R(x, y), T_S(y, z))$ becomes $\sup_{y \in Y} \min(T_R(x), T_S(y, z))$, which is only a function of the output variable z. It is similar for $\sup_{y \in Y} \min(I_R(x, y), I_S(y, z))$ and $\inf_{y \in Y} \max(F_R(x, y), F_S(y, z))$. Hence, the notation of $T_{R \circ S}(x, z)$ can be simplified to $T_{R \circ S}(z)$, so that in the case of R being just an interval neutrosophic set,

$$T_{R \circ S}(z) \;\; = \;\; \sup_{x \in X} \min(T_R(x), T_S(x, z)), \tag{1.44}$$

$$I_{R \circ S}(z) \;\; = \;\; \sup_{x \in X} \min(I_R(x), I_S(x, z)), \tag{1.45}$$

$$F_{R \circ S}(z) \;\; = \;\; \inf_{x \in X} \max(F_R(x), F_S(x, z)). \tag{1.46}$$

Definition 20 (Difference) *The* difference *of two interval neutrosophic sets A and B is an interval neutrosophic set C, written as $C = A \setminus B$, whose truth-membership, indeterminacy-membership and falsity-membership functions are related to those of A and B by*

$$
\begin{align}
\inf T_C(x) &= \min(\inf T_A(x), \inf F_B(x)), \tag{1.47}\\
\sup T_C(x) &= \min(\sup T_A(x), \sup F_B(x)), \tag{1.48}\\
\inf I_C(x) &= \max(\inf I_A(x), 1 - \sup I_B(x)), \tag{1.49}\\
\sup I_C(x) &= \max(\sup I_A(x), 1 - \inf I_B(x)), \tag{1.50}\\
\inf F_C(x) &= \max(\inf F_A(x), \inf T_B(x)), \tag{1.51}\\
\sup F_C(x) &= \max(\sup F_A(x), \sup T_B(x)), \tag{1.52}
\end{align}
$$

for all x in X.

Example 5 *Let A and B be the interval neutrosophic sets defined in Example 1. Then, $A \setminus B =$*
$\langle [0.1, 0.3], [0.7, 0.9], [0.5, 0.7] \rangle / x_1 +$
$\langle [0.5, 0.7], [0.6, 0.8], [0.2, 0.3] \rangle / x_2 + \langle [0.3, 0.4], [0.9, 1.0], [0.4, 0.6] \rangle / x_3.$

Theorem 4 $A \subseteq B \leftrightarrow \bar{B} \subseteq \bar{A}$

Definition 21 (Addition) *The* addition *of two interval neutrosophic sets A and B is an interval neutrosophic set C, written as $C = A + B$, whose truth-membership, indeterminacy-membership and falsity-membership functions are related to those of A and B by*

$$
\begin{align}
\inf T_C(x) &= \min(\inf T_A(x) + \inf T_B(x), 1), \tag{1.53}\\
\sup T_C(x) &= \min(\sup T_A(x) + \sup T_B(x), 1), \tag{1.54}\\
\inf I_C(x) &= \min(\inf I_A(x) + \inf I_B(x), 1), \tag{1.55}\\
\sup I_C(x) &= \min(\sup I_A(x) + \sup I_B(x), 1), \tag{1.56}\\
\inf F_C(x) &= \min(\inf F_A(x) + \inf F_B(x), 1), \tag{1.57}\\
\sup F_C(x) &= \min(\sup F_A(x) + \sup F_B(x), 1), \tag{1.58}
\end{align}
$$

for all x in X.

Example 6 *Let A and B be the interval neutrosophic sets defined in Example 1. Then, $A + B =$*
$\langle [0.7, 1.0], [0.4, 0.8], [0.4, 0.8] \rangle / x_1 +$
$\langle [0.7, 1.0], [0.2, 0.6], [0.7, 1.0] \rangle / x_2 + \langle [1.0, 1.0], [0.2, 0.4], [0.5, 0.7] \rangle / x_3.$

Definition 22 (Cartesian product) *The* cartesian product *of two interval neutrosophic sets A defined on universe X_1 and B defined on universe X_2 is an interval neutrosophic set C, written as $C = A \times B$,*

whose truth-membership, indeterminacy-membership and falsity-membership functions are related to those of A and B by

$$\inf T_C(x,y) = \inf T_A(x) + \inf T_B(y) - \inf T_A(x) \cdot \inf T_B(y), \quad (1.59)$$
$$\sup T_C(x,y) = \sup T_A(x) + \sup T_B(y) - \sup T_A(x) \cdot \sup T_B(y), \quad (1.60)$$
$$\inf I_C(x,y) = \inf I_A(x) \cdot \sup I_B(y), \quad (1.61)$$
$$\sup I_C(x,y) = \sup I_A(x) \cdot \sup I_B(y), \quad (1.62)$$
$$\inf F_C(x,y) = \inf F_A(x) \cdot \inf F_B(y), \quad (1.63)$$
$$\sup F_C(x,y) = \sup F_A(x) \cdot \sup F_B(y), \quad (1.64)$$

for all x in X_1, y in X_2.

Example 7 *Let A and B be the interval neutrosophic sets defined in Example 1. Then, $A \times B =$*
$\langle [0.6, 0.82], [0.03, 0.15], [0.03, 0.15] \rangle / x_1 +$
$\langle [0.6, 0.79], [0, 0.08], [0.1, 0.24] \rangle / x_2 + \langle [0.76, 0.92], [0, 0.03], [0.03, 0.12] \rangle / x_3.$

Definition 23 (Scalar multiplication) *The* scalar multiplication *of interval neutrosophic set A is an interval neutrosophic set B, written as $B = a \cdot A$, whose truth-membership, indeterminacy-membership and falsity-membership functions are related to those of A by*

$$\inf T_B(x) = \min(\inf T_A(x) \cdot a, 1), \quad (1.65)$$
$$\sup T_B(x) = \min(\sup T_A(x) \cdot a, 1), \quad (1.66)$$
$$\inf I_B(x) = \min(\inf I_A(x) \cdot a, 1), \quad (1.67)$$
$$\sup I_B(x) = \min(\sup I_A(x) \cdot a, 1), \quad (1.68)$$
$$\inf F_B(x) = \min(\inf F_A(x) \cdot a, 1), \quad (1.69)$$
$$\sup F_B(x) = \min(\sup F_A(x) \cdot a, 1), \quad (1.70)$$

for all x in X, $a \in R^+$.

Definition 24 (Scalar division) *The* scalar division *of interval neutrosophic set A is an interval neutrosophic set B, written as $B = a \cdot A$, whose truth-membership, indeterminacy-membership and falsity-membership functions are related to those of A by*

$$\inf T_B(x) = \min(\inf T_A(x)/a, 1), \quad (1.71)$$
$$\sup T_B(x) = \min(\sup T_A(x)/a, 1), \quad (1.72)$$
$$\inf I_B(x) = \min(\inf I_A(x)/a, 1), \quad (1.73)$$
$$\sup I_B(x) = \min(\sup I_A(x)/a, 1), \quad (1.74)$$
$$\inf F_B(x) = \min(\inf F_A(x)/a, 1), \quad (1.75)$$
$$\sup F_B(x) = \min(\sup F_A(x)/a, 1), \quad (1.76)$$

for all x in X, $a \in R^+$.

Now we will define two operators: truth-favorite ($\triangle$) and false-favorite (∇) to remove the indeterminacy in the interval neutrosophic sets and transform it into interval valued intuitionistic fuzzy sets or interval valued paraconsistent sets. These two operators are unique on interval neutrosophic sets.

Definition 25 (Truth-favorite) *The* truth-favorite *of interval neutrosophic set A is an interval neutrosophic set B, written as $B = \triangle A$, whose truth-membership and falsity-membership functions are related to those of A by*

$$
\begin{aligned}
\inf T_B(x) &= \min(\inf T_A(x) + \inf I_A(x), 1), & (1.77) \\
\sup T_B(x) &= \min(\sup T_A(x) + \sup I_A(x), 1), & (1.78) \\
\inf I_B(x) &= 0, & (1.79) \\
\sup I_B(x) &= 0, & (1.80) \\
\inf F_B(x) &= \inf F_A(x), & (1.81) \\
\sup F_B(x) &= \sup F_A(x), & (1.82)
\end{aligned}
$$

for all x in X.

Example 8 *Let A be the interval neutrosophic set defined in Example 1. Then, $\triangle A = \langle [0.5, 0.9], [0, 0], [0.3, 0.5] \rangle / x_1 + \langle [0.5, 0.9], [0, 0], [0.2, 0.3] \rangle / x_2 + \langle [0.8, 1.0], [0, 0], [0.2, 0.3] \rangle / x_3$.*

The purpose of truth-favorite operator is to evaluate the maximum of degree of truth-membership of interval neutrosophic set.

Definition 26 (False-favorite) *The* false-favorite *of interval neutrosophic set A is an interval neutrosophic set B, written as $B = \nabla A$, whose truth-membership and falsity-membership functions are related to those of A by*

$$
\begin{aligned}
\inf T_B(x) &= \inf T_A(x), & (1.83) \\
\sup T_B(x) &= \sup T_A(x), & (1.84) \\
\inf I_B(x) &= 0, & (1.85) \\
\sup I_B(x) &= 0, & (1.86) \\
\inf F_B(x) &= \min(\inf F_A(x) + \inf I_A(x), 1), & (1.87) \\
\sup F_B(x) &= \min(\sup F_A(x) + \sup I_A(x), 1), & (1.88)
\end{aligned}
$$

for all x in X.

Example 9 *Let A be the interval neutrosophic set defined in Example 1. Then, $\nabla A = \langle [0.2, 0.4], [0, 0], [0.6, 1.0] \rangle / x_1 +$*
$\langle [0.5, 0.7], [0, 0], [0.2, 0.5] \rangle / x_2 +$
$\langle [0.6, 0.8], [0, 0], [0.4, 0.6] \rangle / x_3$.

The purpose of false-favorite operator is to evaluate the maximum of degree of false-membership of interval neutrosophic set.

Theorem 5 *For every two interval neutrosophic sets A and B:*

  1. $\triangle(A \cup B) \subseteq \triangle A \cup \triangle B$

  2. $\triangle A \cap \triangle B \subseteq \triangle(A \cap B)$

  3. $\nabla A \cup \nabla B \subseteq \nabla(A \cup B)$

  4. $\nabla(A \cap B) \subseteq \nabla A \cap \nabla B$

1.4   Properties of Set-theoretic Operators

In this section, we will give some properties of set-theoretic operators defined on interval neutrosophic sets as in Section 1.3. The proof of these properties is left for the readers.

Property 1 (Commutativity) $A \cup B = B \cup A,\ A \cap B = B \cap A,\ A + B = B + A,\ A \times B = B \times A$

Property 2 (Associativity) $A \cup (B \cup C) = (A \cup B) \cup C,$
$A \cap (B \cap C) = (A \cap B) \cap C,$
$A + (B + C) = (A + B) + C,$
$A \times (B \times C) = (A \times B) \times C.$

Property 3 (Distributivity) $A \cup (B \cap C) = (A \cup B) \cap (A \cup C),\ A \cap (B \cup C) = (A \cap B) \cup (A \cap C).$

Property 4 (Idempotency) $A \cup A = A,\ A \cap A = A,\ \triangle\triangle A = \triangle A,\ \nabla\nabla A = \nabla A.$

Property 5 $A \cap \Phi = \Phi,\ A \cup X = X,\ where\ \inf T_\Phi = \sup T_\Phi = 0,\ \inf I_\Phi = \sup I_\Phi = \inf F_\Phi = \sup F_\Phi = 1$
$and\ \inf T_X = \sup T_X = 1,\ \inf I_X = \sup I_X = \inf F_X = \sup F_X = 0.$

Property 6 $\triangle(A + B) = \triangle A + \triangle B,\ \nabla(A + B) = \nabla A + \nabla B.$

Property 7 $A \cup \Psi = A,\ A \cap X = A,\ where\ \inf T_\Phi = \sup T_\Phi = 0,\ \inf I_\Phi = \sup I_\Phi = \inf F_\Phi = \sup F_\Phi = 1$
$and\ \inf T_X = \sup T_X = 1,\ \inf I_X = \sup I_X = \inf F_X = \sup F_X = 0.$

Property 8 (Absorption) $A \cup (A \cap B) = A$, $A \cap (A \cup B) = A$

Property 9 (DeMorgan's Laws) $\overline{A \cup B} = \bar{A} \cap \bar{B}$, $\overline{A \cap B} = \bar{A} \cup \bar{B}$.

Property 10 (Involution) $\overline{\overline{A}} = A$

Here, we notice that by the definitions of complement, union and intersection of interval neutrosophic set, interval neutrosophic set satisfies the most properties of class set, fuzzy set and intuitionistic fuzzy set. Same as fuzzy set and intuitionistic fuzzy set, it does not satisfy the principle of middle exclude.

1.5 Convexity of Interval Neutrosophic Set

We assume that X is a real Euclidean space E^n for correctness.

Definition 27 (Convexity) *An interval neutrosophic set A is convex if and only if*

$$
\begin{align}
\inf T_A(\lambda x_1 + (1 - \lambda)x_2) &\geq \min(\inf T_A(x_1), \inf T_A(x_2)), & (1.89) \\
\sup T_A(\lambda x_1 + (1 - \lambda)x_2) &\geq \min(\sup T_A(x_1), \sup T_A(x_2)), & (1.90) \\
\inf I_A(\lambda x_1 + (1 - \lambda)x_2) &\leq \max(\inf I_A(x_1), \inf I_A(x_2)), & (1.91) \\
\sup I_A(\lambda x_1 + (1 - \lambda)x_2) &\leq \max(\sup I_A(x_1), \sup I_A(x_2)), & (1.92) \\
\inf F_A(\lambda x_1 + (1 - \lambda)x_2) &\leq \max(\inf F_A(x_1), \inf F_A(x_2)), & (1.93) \\
\sup F_A(\lambda x_1 + (1 - \lambda)x_2) &\leq \max(\sup F_A(x_1), \sup F_A(x_2)), & (1.94)
\end{align}
$$

for all x_1 and x_2 in X and all λ in $[0, 1]$.

Fig. 1.2 is an illustration of convex interval neutrosophic set.

Theorem 6 *If A and B are convex, so is their intersection.*

Definition 28 (Strongly Convex) *An interval neutrosophic set A is strongly convex if for any two distinct points x_1 and x_2, and any λ in the open interval $(0, 1)$,*

$$
\begin{align}
\inf T_A(\lambda x_1 + (1 - \lambda)x_2) &> \min(\inf T_A(x_1), \inf T_A(x_2)), & (1.95) \\
\sup T_A(\lambda x_1 + (1 - \lambda)x_2) &> \min(\sup T_A(x_1), \sup T_A(x_2)), & (1.96) \\
\inf I_A(\lambda x_1 + (1 - \lambda)x_2) &< \max(\inf I_A(x_1), \inf I_A(x_2)), & (1.97) \\
\sup I_A(\lambda x_1 + (1 - \lambda)x_2) &< \max(\sup I_A(x_1), \sup I_A(x_2)), & (1.98) \\
\inf F_A(\lambda x_1 + (1 - \lambda)x_2) &< \max(\inf F_A(x_1), \inf F_A(x_2)), & (1.99) \\
\sup F_A(\lambda x_1 + (1 - \lambda)x_2) &< \max(\sup F_A(x_1), \sup F_A(x_2)), & (1.100)
\end{align}
$$

for all x_1 and x_2 in X and all λ in $[0, 1]$.

Theorem 7 *If A and B are strongly convex, so is their intersection.*

1.6   Conclusions

In this chapter, we have presented an instance of neutrosophic set called the interval neutrosophic set (INS). The interval neutrosophic set is a generalization of classic set, fuzzy set, interval valued fuzzy set, intuitionistic fuzzy sets, interval valued intuitionistic fuzzy set, interval type-2 fuzzy set [LM00] and para-consistent set. The notions of containment, complement, N-norm, N-conorm, relation, and composition have been defined on interval neutrosophic set. Various properties of set-theoretic operators have been proved. In the next chapter, we will discuss the interval neutrosophic logic and logic inference system based on interval neutrosophic set.

1.7   Appendix

Theorem 1 $A \cup B$ *is the smallest interval neutrosophic set containing both A and B.*

Proof Let $C = A \cup B$. $\inf T_C = \max(\inf T_A, \inf T_B)$, $\inf T_C \geq \inf T_A$, $\inf T_C \geq \inf T_B$. $\sup T_C = \max(\sup T_A, \sup T_B$, $\sup T_C \geq \sup T_A$, $\sup T_C \geq \sup T_B$. $\inf I_C = \min(\inf I_A, \inf I_B)$, $\inf I_C \leq \inf I_A$, $\inf I_C \leq \inf I_B$,
$\sup I_C = \min(\sup I_A, \sup I_B)$, $\sup I_C \leq \sup I_A$, $\sup I_C \leq \sup I_B$, $\inf F_C = \min(\inf F_A, \inf F_B)$, $\inf F_C \leq \inf F_A$, $\inf F_C \leq \inf F_B$.
$\sup F_C = \min(\sup F_A, \sup F_B)$, $\sup F_C \leq \sup F_A$, $\sup F_C \leq \sup F_B$. That means C contains both A and B.
Furthermore, if D is any extended vague set containing both A and B, then $\inf T_D \geq \inf T_A$, $\inf T_D \geq \inf T_B$, so $\inf T_D \geq \max(\inf T_A, \inf T_B) = \inf T_C$. $\sup T_D \geq \sup T_A$, $\sup T_D \geq \sup T_B$, so $\sup T_D \geq \max(\sup T_A, \sup T_B) = \sup T_C$. $\inf I_D \leq \inf I_A$, $\inf I_D \leq \inf I_B$, so $\inf I_D \leq \min(\inf I_A, \inf I_B) = \inf I_C$. $\sup I_D \leq \sup I_A$, $\sup I_D \leq \sup I_B$, so $\sup I_D \leq \min(\sup I_A, \sup I_B) = \sup I_C$. $\inf F_D \leq \inf F_A$, $\inf F_D \leq \inf F_B$, so $\inf F_D \leq \min(\inf F_A, \inf F_B) = \inf F_C$. $\sup F_D \leq \sup F_A$, $\sup F_D \leq \sup F_B$, so $\sup F_D \leq \min(\sup F_A, \sup F_B) = \sup F_C$. That implies $C \subseteq D$.

Theorem 2 $A \cap B$ *is the largest interval neutrosophic set contained in both A and B.*

Proof The proof is analogous to the proof of theorem 1.

Theorem 3 *Let P be the power set of all interval neutrosophic sets defined in the universe X. Then $\langle P; I_{N_1}, U_{N_1} \rangle$ is a distributive lattice.*

Proof Let A, B, C be the arbitrary interval neutrosophic sets defined on X. It is easy to verify that $A \cap A = A, A \cup A = A$ (idempotency), $A \cap B = B \cap A, A \cup B = B \cup A$ (commutativity), $(A \cap B) \cap C = A \cap (B \cap C), (A \cup B) \cup C = A \cup (B \cup C)$ (associativity), and $A \cap (B \cup C) = (A \cap B) \cup (A \cap C), A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ (distributivity).

Theorem 4 $A \subseteq B \leftrightarrow \bar{B} \subseteq \bar{A}$

Proof $A \subseteq B \Leftrightarrow \inf T_A \leq \inf T_B, \sup T_A \leq \sup T_B, \inf I_A \geq \inf I_B, \sup I_A \geq \sup I_B, \inf F_A \geq \inf F_B, \sup F_A \geq$
$\sup F_B \Leftrightarrow \inf F_B \leq \inf F_A, \sup F_B \leq \sup F_A,$
$1 - \sup I_B \geq 1 - \sup I_A, 1 - \inf I_B \geq 1 - \inf I_A, \inf T_B \geq \inf T_A, \sup T_B \geq \sup T_A \Leftrightarrow \bar{B} \subseteq \bar{A}.$

Theorem 5 *For every two interval neutrosophic sets A and B:*

1. $\triangle(A \cup B) \subseteq \triangle A \cup \triangle B$

2. $\triangle A \cap \triangle B \subseteq \triangle(A \cap B)$

3. $\nabla A \cup \nabla B \subseteq \nabla(A \cup B)$

4. $\nabla(A \cap B) \subseteq \nabla A \cap \nabla B$

Proof We now prove the first identity. Let $C = A \cup B$.
$\inf T_C(x) = \max(\inf T_A(x), \inf T_B(x)),$
$\sup T_C(x) = \max(\sup T_A(x), \sup T_B(x)),$
$\inf I_C(x) = \min(\inf I_A(x), \inf I_B(x)),$
$\sup I_C(x) = \min(\sup I_A(x), \sup I_B(x)),$
$\inf F_C(x) = \min(\inf F_A(x), \inf F_B(x)),$
$\sup F_C(x) = \min(\sup F_A(x), \sup F_B(x)).$
$\inf T_{\triangle C}(x) = \min(\inf T_C(x) + \inf I_C(x), 1),$
$\sup T_{\triangle C}(x) = \min(\sup T_C(x) + \sup T_C(x), 1),$
$\inf I_{\triangle C}(x) = \sup I_{\triangle C}(x) = 0,$
$\inf F_{\triangle C}(x) = \inf I_C(x),$
$\sup F_{\triangle C}(x) = \sup I_C(x).$
$\inf T_{\triangle A}(x) = \min(\inf T_A(x) + \inf I_A(x), 1),$
$\sup T_{\triangle A}(x) = \min(\sup T_A x + \sup I_A(x), 1),$
$\inf I_{\triangle A}(x) = \sup I_{\triangle A}(x) = 0,$
$\inf F_{\triangle A}(x) = \inf I_A(x),$
$\sup F_{\triangle A}(x) = \sup I_A(x).$
$\inf T_{\triangle B}(x) = \min(\inf T_B(x) + \inf I_B(x), 1),$
$\sup T_{\triangle B}(x) = \min(\sup T_B x + \sup I_B(x), 1),$
$\inf I_{\triangle B}(x) = \sup I_{\triangle B}(x) = 0,$
$\inf F_{\triangle B}(x) = \inf I_B(x),$
$\sup F_{\triangle B}(x) = \sup I_B(x).$
$\inf T_{\triangle A \cup \triangle B}(x) = \max(\inf T_{\triangle A}(x), \inf T_{\triangle B}(x)),$
$\sup T_{\triangle A \cup \triangle B}(x) = \max(\sup T_{\triangle A}(x), \sup T_{\triangle B}(x)),$
$\inf I_{\triangle A \cup \triangle B}(x) = \sup I_{\triangle A \cup \triangle B}(x) = 0,$
$\inf F_{\triangle A \cup \triangle B}(x) = \min(\inf F_{\triangle A}(x), \inf F_{\triangle B}(x)),$
$\sup F_{\triangle A \cup \triangle B}(x) = \min(\inf F_{\triangle A}(x), \inf F_{\triangle B}(x)).$

Because,

$\inf T_{\triangle(A\cup B)} \le \inf T_{\triangle A\cup\triangle B}$,

$\sup T_{\triangle(A\cup B)} \le \sup T_{\triangle A\cup\triangle B}$,

$\inf I_{\triangle(A\cup B)} = \inf T_{\triangle A\cup\triangle B} = 0$,

$\sup I_{\triangle(A\cup B)} = \sup T_{\triangle A\cup\triangle B} = 0$,

$\inf F_{\triangle(A\cup B)} = \inf F_{\triangle A\cup\triangle B}$,

$\sup F_{\triangle(A\cup B)} = \sup T_{\triangle A\cup\triangle B}$,

so, $\triangle(A\cup B) \subseteq \triangle A \cup \triangle B$. The other identities can be proved in a similar manner.

Theorem 6 *If A and B are convex, so is their intersection.*

Proof Let $C = A \cap B$, then

$\inf T_C(\lambda x_1 + (1-\lambda)x_2) \ge \min(\inf T_A(\lambda x_1 + (1-\lambda)x_2), \inf T_B(\lambda x_1 + (1-\lambda)x_2))$, $\sup T_C(\lambda x_1 + (1-\lambda)x_2) \ge \min(\sup T_A(\lambda x_1 + (1 - \lambda)x_2), \sup T_B(\lambda x_1 + (1 - \lambda)x_2))$, $\inf I_C(\lambda x_1 + (1 - \lambda)x_2) \le \max(\inf I_A(\lambda x_1 + (1 - \lambda)x_2), \inf I_B(\lambda x_1 + (1 - \lambda)x_2))$, $\sup I_C(\lambda x_1 + (1-\lambda)x_2) \le \max(\sup I_A(\lambda x_1 + (1 - \lambda)x_2), \sup I_B(\lambda x_1 + (1 - \lambda)x_2))$, $\inf F_C(\lambda x_1 + (1-\lambda)x_2) \le \max(\inf F_A(\lambda x_1 + (1-\lambda)x_2), \inf F_B(\lambda x_1 + (1-\lambda)x_2))$, $\sup F_C(\lambda x_1 + (1-\lambda)x_2) \le \max(\inf F_A(\lambda x_1 + (1-\lambda)x_2), \inf F_B(\lambda x_1 + (1-\lambda)x_2))$, Since A and B are convex: $\inf T_A(\lambda x_1 + (1-\lambda)x_2) \ge \min(\inf T_A(x1), \inf T_A(x2))$, $\sup T_A(\lambda x_1 + (1 - \lambda)x_2) \ge \min(\sup T_A(x1), \sup T_A(x2))$, $\inf I_A(\lambda x_1 + (1 - \lambda)x_2) \le \max(\inf I_A(x1), \inf I_A(x2))$, $\sup I_A(\lambda x_1 + (1 - \lambda)x_2) \le$
$\max(\sup I_A(x1), \sup I_A(x2))$, $\inf F_A(\lambda x_1 + (1 - \lambda)x_2) \le$
$\max(\inf F_A(x1), \inf F_A(x2))$, $\sup F_A(\lambda x_1 + (1 - \lambda)x_2) \le$
$\max(\sup F_A(x1), \sup F_A(x2))$,
$\inf T_B(\lambda x_1 + (1-\lambda)x_2) \ge \min(\inf T_B(x1), \inf T_A(x2))$, $\sup T_B(\lambda x_1 + (1-\lambda)x_2) \ge \min(\sup T_B(x1), \sup T_A(x2))$,
$\inf I_B(\lambda x_1 + (1 - \lambda)x_2) \le$
$\max(\inf I_B(x1), \inf I_A(x2))$, $\sup I_B(\lambda x_1 + (1 - \lambda)x_2) \le$
$\max(\sup I_B(x1), \sup I_A(x2))$,
$\inf F_B(\lambda x_1 + (1 - \lambda)x_2) \le \max(\inf F_B(x1), \inf F_A(x2))$,
$\sup F_B(\lambda x_1 + (1 - \lambda)x_2) \le \max(\sup F_B(x1), \sup F_A(x2))$,
Hence,
$\inf T_C(\lambda x_1 + (1 - \lambda)x_2) \ge \min(\min(\inf T_A(x_1), \inf T_A(x_2))$
$, \min(\inf T_B(x_1), \inf T_B(x_2))) = \min(\min(\inf T_A(x_1), \inf T_B(x_1))$,
$\min(\inf T_A(x_2), \inf T_B(x_2))) = \min(\inf T_C(x_1), \inf T_C(x_2))$, $\sup T_C(\lambda x_1 + (1-\lambda)x_2) \ge \min(\min(\sup T_A(x_1), \sup T_A(x_2))$,
$\min(\sup T_B(x_1), \sup T_B(x_2))) = \min(\min(\sup T_A(x_1), \sup T_B(x_1))$,
$\min(\sup T_A(x_2), \sup T_B(x_2))) = \min(\sup T_C(x_1), \sup T_C(x_2))$,
$\inf I_C(\lambda x_1 + (1 - \lambda)x_2) \le \max(\max(\inf I_A(x_1)$,
$\inf I_A(x_2)), \max(\inf I_B(x_1), \inf I_B(x_2))) = \max(\max(\inf I_A(x_1)$,
$\inf I_B(x_1)), \max(\inf I_A(x_2), \inf I_B(x_2))) = \max(\inf I_C(x_1), \inf I_C(x_2))$,
$\sup I_C(\lambda x_1 + (1 - \lambda)x_2) \le \max(\max(\sup I_A(x_1), \sup I_A(x_2))$
$, \max(\sup I_B(x_1), \sup I_B(x_2))) = \max(\max(\sup I_A(x_1), \sup I_B(x_1))$,
$\max(\sup I_A(x_2), \sup I_B(x_2))) = \max(\sup I_C(x_1), \sup I_C(x_2))$,

$\inf F_C(\lambda x_1 + (1 - \lambda)x_2) \leq \max(\max(\inf F_A(x_1), \inf F_A(x_2)),$
$\max(\inf F_B(x_1), \inf F_B(x_2))) = \max(\max(\inf F_A(x_1), \inf F_B(x_1)),$
$\max(\inf F_A(x_2), \inf F_B(x_2))) =$
$\max(\inf F_C(x_1), \inf F_C(x_2)), \sup F_C(\lambda x_1 + (1 - \lambda)x_2)$
$\leq \max(\max(\sup F_A(x_1), \sup F_A(x_2)),$
$\max(\sup F_B(x_1), \sup F_B(x_2))) = \max(\max(\sup F_A(x_1), \sup F_B(x_1)),$
$\max(\sup F_A(x_2), \sup F_B(x_2))) = \max(\sup F_C(x_1), \sup F_C(x_2)).$

Theorem 7 *If A and B are strongly convex, so is their intersection.*

Proof The proof is analogous to the proof of Theorem 6.

Chapter 2

Interval Neutrosophic Logic

In this chapter, we present a novel interval neutrosophic logic that generalizes the interval valued fuzzy logic, the intuitionistic fuzzy logic and paraconsistent logics which only consider truth-degree or falsity-degree of a proposition. In the interval neutrosophic logic, we consider not only truth-degree and falsity-degree but also indeterminacy-degree which can reliably capture more information under uncertainty. We introduce mathematical definitions of an interval neutrosophic propositional calculus and an interval neutrosophic predicate calculus. We propose a general method to design an interval neutrosophic logic system which consists of neutrosophication, neutrosophic inference, a neutrosophic rule base, neutrosophic type reduction and deneutrosophication. A neutrosophic rule contains input neutrosophic linguistic variables and output neutrosophic linguistic variables. A neutrosophic linguistic variable has neutrosophic linguistic values which defined by interval neutrosophic sets characterized by three membership functions: truth-membership, falsity-membership and indeterminacy-membership. The interval neutrosophic logic can be applied to many potential real applications where information is imprecise, uncertain, incomplete and inconsistent such as Web intelligence, medical informatics, bioinformatics, decision making, etc.

2.1 Introduction

The concept of fuzzy sets was introduced by Zadeh in 1965 [Zad65b]. Since then fuzzy sets and fuzzy logic have been applied to many real applications to handle uncertainty. The traditional fuzzy set uses one real value $\mu_A(x) \in [0, 1]$ to represent the grade of membership of fuzzy set A defined on universe X. The corresponding fuzzy logic associates each proposition p with a real value $\mu(p) \in [0, 1]$ which represents the degree of truth. Sometimes $\mu_A(x)$ itself is uncertain and hard to be defined by a crisp value. So the concept of interval valued fuzzy sets was proposed [Tur86] to capture the uncertainty of grade of membership. The interval valued fuzzy set uses an interval value $[\mu_A^L(x), \mu_A^U(x)]$ with $0 \leq \mu_A^L(x) \leq \mu_A^U(x) \leq 1$ to represent the grade of membership of fuzzy set. The traditional fuzzy logic can be easily extended to the interval valued fuzzy logic. There are related works such as type-2 fuzzy sets and type-2 fuzzy logic [KM98, LM00, MJ02]. The family of fuzzy sets and fuzzy logic can only handle "complete" information that is if a grade of truth-membership is $\mu_A(x)$ then a grade of false-membership is $1 - \mu_A(x)$ by default. In some applications

such as expert systems, decision making systems and information fusion systems, the information is both uncertain and incomplete. That is beyond the scope of traditional fuzzy sets and fuzzy logic. In 1986, Atanassov introduced the intuitionistic fuzzy set [Ata86] which is a generalization of a fuzzy set and provably equivalent to an interval valued fuzzy set. The intuitionistic fuzzy sets consider both truth-membership and false-membership. The corresponding intuitionistic fuzzy logic [Ata88, AG90, AG98] associates each proposition p with two real values $\mu(p)$-truth degree and $\nu(p)$-falsity degree, respectively, where $\mu(p), \nu(p) \in [0,1], \mu(p) + \nu(p) \leq 1$. So intuitionistic fuzzy sets and intuitionistic fuzzy logic can handle uncertain and incomplete information.

However, inconsistent information exists in a lot of real situations such as those mentioned above. It is obvious that the intuitionistic fuzzy logic cannot reason with inconsistency because it requires $\mu(p) + \nu(p) \leq 1$. Generally, two basic approaches are used to solve the inconsistency problem in knowledge bases: the belief revision and paraconsistent logics. The goal of the first approach is to make an inconsistent theory consistent, either by revising it or by representing it by a consistent semantics. On the other hand, the paraconsistent approach allows reasoning in the presence of inconsistency as contradictory information can be derived or introduced without trivialization [dACM02a]. de Costa's $\mathcal{C}_w$ logic [Cos77b] and Belnap's four-valued logic [Bel77a] are two well-known paraconsistent logics.

Neutrosophy was introduced by Smarandache in 1995. "Neutrosophy is a branch of philosophy which studies the origin, nature and scope of neutralities, as well as their interactions with different ideational spectra" [Sma03]. Neutrosophy includes neutrosophic probability, neutrosophic sets and neutrosophic logic. In a neutrosophic set (neutrosophic logic), indeterminacy is quantified explicitly and truth-membership (truth-degree), indeterminacy-membership (indeterminacy-degree) and false-membership (falsity-degree) are independent. The independence assumption is very important in a lot of applications such as information fusion when we try to combine different data from different sensors. A neutrosophic set (neutrosophic logic) is different from an intuitionistic fuzzy set (intuitionistic fuzzy logic) where indeterminacy membership (indeterminacy-degree) is $1 - \mu_A(x) - \nu_A(x)$ $(1 - \mu(p) - \nu(p))$ by default.

The neutrosophic set generalizes the above mentioned sets from a philosophical point of view. From a scientific or engineering point of view, the neutrosophic set and set-theoretic operators need to be specified meaningfully. Otherwise, it will be difficult to apply to the real applications. In chapter 1 we discussed a special neutrosophic set called an interval neutrosophic set and defined a set of set-theoretic operators. It is natural to define the interval neutrosophic logic based on interval neutrosophic sets. In this chapter, we give mathematical definitions of an interval neutrosophic propositional calculus and a first order interval neutrosophic predicate calculus.

The rest of this chapter is organized as follows. Section 2.2 gives the mathematical definition of the interval neutrosophic propositional calculus. Section 2.3 gives the mathematical definition of the first order interval neutrosophic predicate calculus. Section 2.4 provides one application example of interval neutrosophic logic as the foundation for the design of interval neutrosophic logic system. In section 2.5 we conclude the chapter and discuss the future research directions.

2.2 Interval Neutrosophic Propositional Calculus

In this section, we introduce the elements of an interval neutrosophic propositional calculus based on the definition of the interval neutrosophic sets by using the notations from the theory of classical propositional calculus [Men87].

2.2.1 Syntax of Interval Neutrosophic Propositional Calculus

Here we give the formalization of syntax of the interval neutrosophic propositional calculus.

Definition 29 *An* alphabet *of the interval neutrosophic propositional calculus consists of three classes of symbols:*

1. *A set of* interval neutrosophic propositional variables, *denoted by lower-case letters, sometimes indexed;*

2. *Five* connectives $\wedge, \vee, \neg, \rightarrow, \leftrightarrow$ *which are called conjunction, disjunction, negation, implication, and biimplication symbols respectively;*

3. *The parentheses (and).*

$\square$

The alphabet of the interval neutrosophic propositional calculus has combinations obtained by assembling connectives and interval neutrosophic propositional variables in strings. The purpose of the construction rules is to have the specification of distinguished combinations, called formulas.

Definition 30 *The set of formulas (well-formed formulas) of interval neutrosophic propositional calculus is defined as follows.*

1. *Every* interval neutrosophic propositional variable *is a formula;*

2. *If p is a formula, then so is $(\neg p)$;*

3. *If p and q are formulas, then so are*

    (a) $(p \wedge q)$,

    (b) $(p \vee q)$,

    (c) $(p \rightarrow q)$, and

    (d) $(p \leftrightarrow q)$.

4. *No sequence of symbols is a formula which is not required to be by 1, 2, and 3.*

$\square$

To avoid having formulas cluttered with parentheses, we adopt the following precedence hierarchy, with the highest precedence at the top:

$$\neg,$$
$$\wedge, \vee,$$
$$\rightarrow, \leftrightarrow.$$

Here is an example of the interval neutrosophic propositional calculus formula:

$$\neg p_1 \wedge p_2 \vee (p_1 \rightarrow p_3) \rightarrow p_2 \wedge \neg p_3$$

Definition 31 *The* language of interval neutrosophic propositional calculus *given by an alphabet consists of the set of all formulas constructed from the symbols of the alphabet.* □

2.2.2   Semantics of Interval Neutrosophic Propositional Calculus

The study of interval neutrosophic propositional calculus comprises, among others, a *syntax*, which has the distinction of well-formed formulas, and a *semantics*, the purpose of which is the assignment of a meaning to well-formed formulas.

To each interval neutrosophic proposition p, we associate it with an ordered triple components $\langle t(p), i(p), f(p) \rangle$, where $t(p), i(p), f(p) \subseteq [0, 1]$. $t(p), i(p), f(p)$ is called truth-degree, indeterminacy-degree and falsity-degree of p, respectively. Let this assignment be provided by an *interpretation function* or *interpretation INL* defined over a set of propositions P in such a way that

$$INL(p) = \langle t(p), i(p), f(p) \rangle.$$

Hence, the function $INL : P \rightarrow N$ gives the truth, indeterminacy and falsity degrees of all propositions in P. We assume that the interpretation function INL assigns to the logical truth $T : INL(T) = \langle 1, 0, 0 \rangle$, and to $F : INL(F) = \langle 0, 1, 1 \rangle$.

An interpretation which makes a formula true is a *model* of the formula.

Let i, l be the subinterval of $[0, 1]$. Then $i + l = [\inf i + \inf l, \sup i + \sup l]$, $i - l = [\inf i - \sup l, \sup i - \inf l]$, $\max(i, l) = [\max(\inf i, \inf l), \max(\sup i, \sup l)]$, $\min(i, l) = [\min(\inf i, \inf l), \min(\sup i, \sup l)]$.

The semantics of four interval neutrosophic propositional connectives is given in Table I. Note that $p \leftrightarrow q$ if and only if $p \rightarrow q$ and $q \rightarrow p$.

Example 10 $INL(p) = \langle 0.5, 0.4, 0.7 \rangle$ *and* $INL(q) = \langle 1, 0.7, 0.2 \rangle$. *Then,* $INL(\neg p) = \langle 0.7, 0.6, 0.5 \rangle$, $INL(p \wedge \neg p) = \langle 0.5, 0.4, 0.7 \rangle$, $INL(p \vee q) = \langle 1, 0.7, 0.2 \rangle$, $INL(p \rightarrow q) = \langle 1, 1, 0 \rangle$. □

A given well-formed interval neutrosophic propositional formula will be called a tautology (valid) if $INL(A) = \langle 1, 1, 0 \rangle$, for all interpretation functions INL. It will be called a contradiction (inconsistent) if $INL(A) = \langle 0, 0, 1 \rangle$, for all interpretation functions INL.

Table 2.1: Semantics of Four Connectives in Interval Neutrosophic Propositional Logic

Connectives	Semantics
$INL(\neg p)$	$\langle f(p), 1 - i(p), t(p) \rangle$
$INL(p \wedge q)$	$\langle \min(t(p), t(q)), \max(i(p), i(q)), \max(f(p), f(q)) \rangle$
$INL(p \vee q)$	$\langle \max(t(p), t(q)), \min(i(p), i(q)), \min(f(p), f(q)) \rangle$
$INL(p \rightarrow q)$	$\langle \min(1, 1 - t(p) + t(q)), \max(0, i(q) - i(p)), \max(0, f(q) - f(p)) \rangle$

Definition 32 *Two formulas p and q are said to be* equivalent, *denoted $p = q$, if and only if the $INL(p) = INL(q)$ for every interpretation function INL.* □

Theorem 8 *Let F be the set of formulas and $\wedge$ be the meet and $\vee$ the join, then $\langle F; \wedge, \vee \rangle$ is a distributive lattice.*

Proof We leave the proof to the reader.

Theorem 9 *If p and $p \rightarrow q$ are tautologies, then q is also a tautology.*

Proof Since p and $p \rightarrow q$ are tautologies then for every INL, $INL(p) = INL(p \rightarrow q) = \langle 1, 0, 0 \rangle$, that is $t(p) = 1, i(p) = f(p) = 0$, $t(p \rightarrow q) = \min(1, 1 - t(p) + t(q)) = 1$, $i(p \rightarrow q) = \max(0, i(q) - f(p)) = 0$, $f(p \rightarrow q) = \max(0, f(q) - f(p)) = 0$. Hence,
t(q) = 1, i(q) = f(q) = 0. So q is a tautology.

2.2.3 Proof Theory of Interval Neutrosophic Propositional Calculus

Here we give the proof theory for interval neutrosophic propositional logic to complement the semantics part.

Definition 33 *The interval neutrosophic propositional logic is defined by the following axiom schema.*

$$p \rightarrow (q \rightarrow p)$$
$$p_1 \wedge \ldots \wedge p_m \rightarrow q_1 \vee \ldots q_n \; provided \; some \; p_i \; is \; some \; q_j$$
$$p \rightarrow (q \rightarrow p \wedge q)$$
$$(p \rightarrow r) \rightarrow ((q \rightarrow r) \rightarrow (p \vee q \rightarrow r))$$
$$(p \vee q) \rightarrow r \; iff \; p \rightarrow r \; and \; q \rightarrow r$$
$$p \rightarrow q \; iff \; \neg q \rightarrow \neg p$$
$$p \rightarrow q \; and \; q \rightarrow r \; implies \; p \rightarrow r$$
$$p \rightarrow q \; iff \; p \leftrightarrow (p \wedge q) \; iff \; q \rightarrow (p \vee q)$$

□

The concept of (formal) deduction of a formula from a set of formulas, that is, using the standard notation, $\Gamma \vdash p$, is defined as usual; in this case, we say that p is a syntactical consequence of the formulas in T.

Theorem 10 *For interval neutrosophic propositional logic, we have*

1. $\{p\} \vdash p$,

2. $\Gamma \vdash p$ *entails* $\Gamma \cup \Delta \vdash p$,

3. *if* $\Gamma \vdash p$ *for any* $p \in \Delta$ *and* $\Delta \vdash q$, *then* $\Gamma \vdash q$.

Proof It is immediate from the standard definition of the syntactical consequence ($\vdash$).

Theorem 11 *In interval neutrosophic propositional logic, we have:*

1. $\neg\neg p \leftrightarrow p$

2. $\neg(p \wedge q) \leftrightarrow \neg p \vee \neg q$

3. $\neg(p \vee q) \leftrightarrow \neg p \wedge \neg q$

Proof Proof is straight forward by following the semantics of interval neutrosophic propositional logic.

Theorem 12 *In interval neutrosophic propositional logic, the following schema do not hold:*

1. $p \vee \neg p$

2. $\neg(p \wedge \neg p)$

3. $p \wedge \neg p \rightarrow q$

4. $p \wedge \neg p \rightarrow \neg q$

5. $\{p, p \rightarrow q\} \vdash q$

6. $\{p \rightarrow q, \neg q\} \vdash \neg p$

7. $\{p \vee q, \neg q\} \vdash p$

8. $\neg p \vee q \leftrightarrow p \rightarrow q$

Proof Immediate from the semantics of interval neutrosophic propositional logic.

Example 11 *To illustrate the use of the interval neutrosophic propositional consequence relation, let's consider the following example.*

$$p \rightarrow (q \wedge r)$$

$$r \rightarrow s$$

$$q \rightarrow \neg s$$

$$a$$

From $p \rightarrow (q \wedge r)$, we get $p \rightarrow q$ and $p \rightarrow r$. From $p \rightarrow q$ and $q \rightarrow \neg s$, we get $p \rightarrow \neg s$. From $p \rightarrow r$ and $r \rightarrow s$, we get $p \rightarrow s$. Hence, p is equivalent to $p \wedge s$ and $p \wedge \neg s$. However, we cannot detach s from p nor $\neg s$ from p. This is in part due to interval neutrosophic propositional logic incorporating neither modus ponens nor and elimination. □

2.3 Interval Neutrosophic Predicate Calculus

In this section, we will extend our consideration to the full language of first order interval neutrosophic predicate logic. First we give the formalization of syntax of first order interval neutrosophic predicate logic as in classical first-order predicate logic.

2.3.1 Syntax of Interval Neutrosophic Predicate Calculus

Definition 34 *An* alphabet *of the first order interval neutrosophic predicate calculus consists of seven classes of symbols:*

1. variables, *denoted by lower-case letters, sometimes indexed;*

2. constants, *denoted by lower-case letters;*

3. function symbols, *denoted by lower-case letters, sometimes indexed;*

4. predicate symbols, *denoted by lower-case letters, sometimes indexed;*

5. *Five* connectives $\wedge, \vee, \neg, \rightarrow, \leftrightarrow$ *which are called the conjunction, disjunction, negation, implication, and biimplication symbols respectively;*

6. *Two* quantifiers, *the* universal quantifier $\forall$ *(for all) and the* existential quantifier $\exists$ *(there exists);*

7. *The* parentheses *(and).*

□

To avoid having formulas cluttered with brackets, we adopt the following precedence hierarchy, with the highest precedence at the top:

$$\neg, \forall, \exists$$

$$\wedge, \vee$$

$$\rightarrow, \leftrightarrow$$

Next we turn to the definition of the first order interval neutrosophic language given by an alphabet.

Definition 35 *A* term *is defined as follows:*

1. *A variable is a term.*

2. *A constant is a term.*

3. *If f is an n-ary function symbol and $t_1, \ldots, t_n$ are terms, then $f(t_1, \ldots, f_n)$ is a term.*

$\square$

Definition 36 *A* (well-formed)formula *is defined inductively as follows:*

1. *If p is an n-ary predicate symbol and $t_1, \ldots, t_n$ are terms, then $p(t_1, \ldots, t_n)$ is a formula (called an* atomic formula *or, more simply, an* atom*).*

2. *If F and G are formulas, then so are $(\neg F), (F \wedge G), (F \vee G), (F \rightarrow G)$ and $(F \leftrightarrow G)$.*

3. *If F is a formula and x is a variable, then $(\forall x F)$ and $(\exists x F)$ are formulas.*

$\square$

Definition 37 *The* first order interval neutrosophic language *given by an alphabet consists of the set of all formulas constructed from the symbols of the alphabet.* $\square$

Example 12 $\forall x \exists y(p(x, y) \rightarrow q(x)), \neg \exists x(p(x, a) \wedge q(x))$ *are formulas.* $\square$

Definition 38 *The* scope *of $\forall x$ (resp. $\exists x$) in $\forall x F$ (resp. $\exists x F$) is F. A* bound occurrence *of a variable in a formula is an occurrence immediately following a quantifier or an occurrence within the scope of a quantifier, which has the same variable immediately after the quantifier. Any other occurrence of a variable is* free. $\square$

Example 13 *In the formula $\forall x p(x, y) \vee q(x)$, the first two occurrences of x are bound, while the third occurrence is free, since the scope of $\forall x$ is $p(x, y)$ and y is free.* $\square$

2.3.2 Semantics of Interval Neutrosophic Predicate Calculus

In this section, we study the semantics of interval neutrosophic predicate calculus, the purpose of which is the assignment of a meaning to well-formed formulas. In the interval neutrosophic propositional logic, an interpretation is an assignment of truth values (ordered triple component) to propositions. In the first order interval neutrosophic predicate logic, since there are variables involved, we have to do more than that. To define an interpretation for a well-formed formula in this logic, we have to specify two things, the domain and an assignment to constants and predicate symbols occurring in the formula. The following is the formal definition of an interpretation of a formula in the first order interval neutrosophic predicate logic.

Definition 39 *An* interpretation function (or interpretation) *of a formula F in the first order interval neutrosophic predicate logic consists of a nonempty domain D, and an assignment of "values" to each constant and predicate symbol occurring in F as follows:*

1. *To each constant, we assign an element in D.*

2. *To each n-ary function symbol, we assign a mapping from D^n to D. (Note that $D^n = \{(x_1, \ldots, x_n) | x_1 \in D, \ldots, x_n \in D\}$).*

3. *Predicate symbols get their meaning through evaluation functions E which assign to each variable x an element $E(x) \in D$. To each n-ary predicate symbol p, there is a function $INP(p) : D^n \to N$. So $INP(p(x_1, \ldots, x_n)) = INP(p)(E(x_1), \ldots, E(x_n))$.*

$\square$

That is, $INP(p)(a_1, \ldots, a_n) = \langle t(p(a_1, \ldots, a_n)), i(p(a_1, \ldots, a_n)), f(p(a_1, \ldots, a_n))\rangle$, where $t(p(a_1, \ldots, a_n)), i(p(a_1, \ldots, a_n)), f(p(a_1, \ldots, a_n)) \subseteq [0, 1]$. They are called truth-degree, indeterminacy-degree and falsity-degree of $p(a_1, \ldots, a_n)$ respectively. We assume that the interpretation function INP assigns to the logical truth $T : INP(T) = \langle 1, 1, 0\rangle$, and to $F : INP(F) = \langle 0, 0, 1\rangle$.

The semantics of four interval neutrosophic predicate connectives and two quantifiers is given in Table II. For simplification of notation, we use p to denote $p(a_1, \ldots, a_i)$. Note that $p \leftrightarrow q$ if and only if $p \to q$ and $q \to p$.

Example 14 *Let $D = 1, 2, 3$ and $p(1) = \langle 0.5, 1, 0.4\rangle, p(2) = \langle 1, 0.2, 0\rangle, p(3) = \langle 0.7, 0.4, 0.7\rangle$. Then $INP(\forall x p(x)) = \langle 0.5, 0.2, 0.7\rangle$, and $INP(\exists x p(x)) = \langle 1, 1, 0\rangle$.* $\square$

Definition 40 *A formula F is* consistent (satisfiable) *if and only if there exists an interpretation I such that F is evaluated to $\langle 1, 1, 0\rangle$ in I. If a formula F is T in an interpretation I, we say that I is a* model *of F and I* satisfies *F.* $\square$

Definition 41 *A formula F is* inconsistent (unsatisfiable) *if and only if there exists no interpretation that satisfies F.* $\square$

Table 2.2: Semantics of Four Connectives and Two Quantifiers in Interval Neutrosophic Predicate Logic

Connectives	Semantics
$INP(\neg p)$	$\langle f(p), 1 - i(p), t(p) \rangle$
$INP(p \wedge q)$	$\langle \min(t(p), t(q)), \max(i(p), i(q)), \max(f(p), f(q)) \rangle$
$INP(p \vee q)$	$\langle \max(t(p), t(q)), \min(i(p), i(q)), \min(f(p), f(q)) \rangle$
$INP(p \rightarrow q)$	$\langle \min(1, 1 - t(p) + t(q)), \max(0, i(q) - i(p)), \max(0, f(q) - f(p)) \rangle$
$INP(\forall x F)$	$\langle \min t(F(E(x))), \min i(F(E(x))), \max f(F(E(x))) \rangle,\ E(x) \in D$
$INP(\exists x F)$	$\langle \max t(F(E(x))), \max i(F(E(x))), \min f(F(E(x))) \rangle,\ E(x) \in D$

Definition 42 *A formula F is* valid *if and only if every interpretation of F satisfies F.* □

Definition 43 *A formula F is a* logical consequence *of formulas $F_1, \ldots, F_n$ if and only if for every interpretation I, if $F_1 \wedge \ldots \wedge F_n$ is true in I, F is also true in I.* □

Example 15 $(\forall x)(p(x) \rightarrow (\exists y)p(y))$ *is valid,* $(\forall x)p(x) \wedge (\exists y)\neg p(y)$ *is consistent.* □

Theorem 13 *There is no inconsistent formula in the first order interval neutrosophic predicate logic.*

Proof It is direct from the definition of semantics of interval neutrosophic predicate logic.

Note that the first order interval neutrosophic predicate logic can be considered as an extension of the interval neutrosophic propositional logic. When a formula in the first order logic contains no variables and quantifiers, it can be treated just as a formula in the propositional logic.

2.3.3   Proof Theory of Interval Neutrosophic Predicate Calculus

In this part, we give the proof theory for first order interval neutrosophic predicate logic to complement the semantics part.

Definition 44 *The first order interval neutrosophic predicate logic is defined by the following axiom schema.*

$$(p \rightarrow q(x)) \rightarrow (p \rightarrow \forall x q(x))$$
$$\forall x p(x) \rightarrow p(a)$$
$$p(x) \rightarrow \exists x p(x)$$
$$(p(x) \rightarrow q) \rightarrow (\exists x p(x) \rightarrow q)$$

□

Theorem 14 *In the first order interval neutrosophic predicate logic, we have:*

1. $p(x) \vdash \forall x p(x)$

2. $p(a) \vdash \exists x p(x)$

3. $\forall x p(x) \vdash p(y)$

4. $\Gamma \cup \{p(x)\} \vdash q$, *then* $\Gamma \cup \{\exists x p(x)\} \vdash q$

Proof Directly from the definition of the semantics of first order interval neutrosophic predicate logic.

Theorem 15 *In the first order interval neutrosophic predicate logic, the following schemes are valid, where r is a formula in which x does not appear free:*

1. $\forall x r \leftrightarrow r$

2. $\exists x r \leftrightarrow r$

3. $\forall x \forall y p(x, y) \leftrightarrow \forall y \forall x p(x, y)$

4. $\exists x \exists y p(x, y) \leftrightarrow \exists y \exists x p(x, y)$

5. $\forall x \forall y p(x, y) \rightarrow \forall x p(x, x)$

6. $\exists x p(x, x) \rightarrow \exists x \exists y p(x, y)$

7. $\forall x p(x) \rightarrow \exists x p(x)$

8. $\exists x \forall y p(x, y) \rightarrow \forall y \exists x p(x, y)$

9. $\forall x (p(x) \land q(x)) \leftrightarrow \forall x p(x) \land \forall x q(x)$

10. $\exists x (p(x) \lor q(x)) \leftrightarrow \exists x p(x) \lor \exists x q(x)$

11. $p \land \forall x q(x) \leftrightarrow \forall x (p \land q(x))$

12. $p \lor \forall x q(x) \leftrightarrow \forall x (p \lor q(x))$

13. $p \land \exists x q(x) \leftrightarrow \exists x (p \land q(x))$

14. $p \lor \exists x q(x) \leftrightarrow \exists x (p \lor q(x))$

15. $\forall x (p(x) \rightarrow q(x)) \rightarrow (\forall x p(x) \rightarrow \forall x q(x))$

16. $\forall x (p(x) \rightarrow q(x)) \rightarrow (\exists x p(x) \rightarrow \exists x q(x))$

17. $\exists x (p(x) \land q(x)) \rightarrow \exists x p(x) \land \exists x q(x)$

18. $\forall x p(x) \lor \forall x q(x) \rightarrow \forall x (p(x) \lor q(x))$

19. $\neg \exists x \neg p(x) \leftrightarrow \forall x p(x)$

20. $\neg \forall x \neg p(x) \leftrightarrow \exists p(x)$

21. $\neg \exists x p(x) \leftrightarrow \forall x \neg p(x)$

22. $\exists x \neg p(x) \leftrightarrow \neg \forall x p(x)$

Proof It is straightforward from the definition of the semantics and axiomatic schema of first order interval neutrosophic predicate logic.

2.4   An Application of Interval Neutrosophic Logic

In this section we provide one practical application of the interval neutrosophic logic – Interval Neutrosophic Logic System (INLS). INLS can handle rule uncertainty as same as type-2 FLS [LM00], besides, it can handle rule inconsistency without the danger of trivialization. Like the classical FLS, INLS is also characterized by IF-THEN rules. INLS consists of neutrosophication, neutrosophic inference, a neutrosophic rule base, neutrosophic type reduction and deneutrosophication. Given an input vector $x = (x_1, \ldots, x_n)$, where $x_1, \ldots, x_n$ can be crisp inputs or neutrosophic sets, the INLS will generate a crisp output y. The general scheme of INLS is shown in Fig. 2.1.
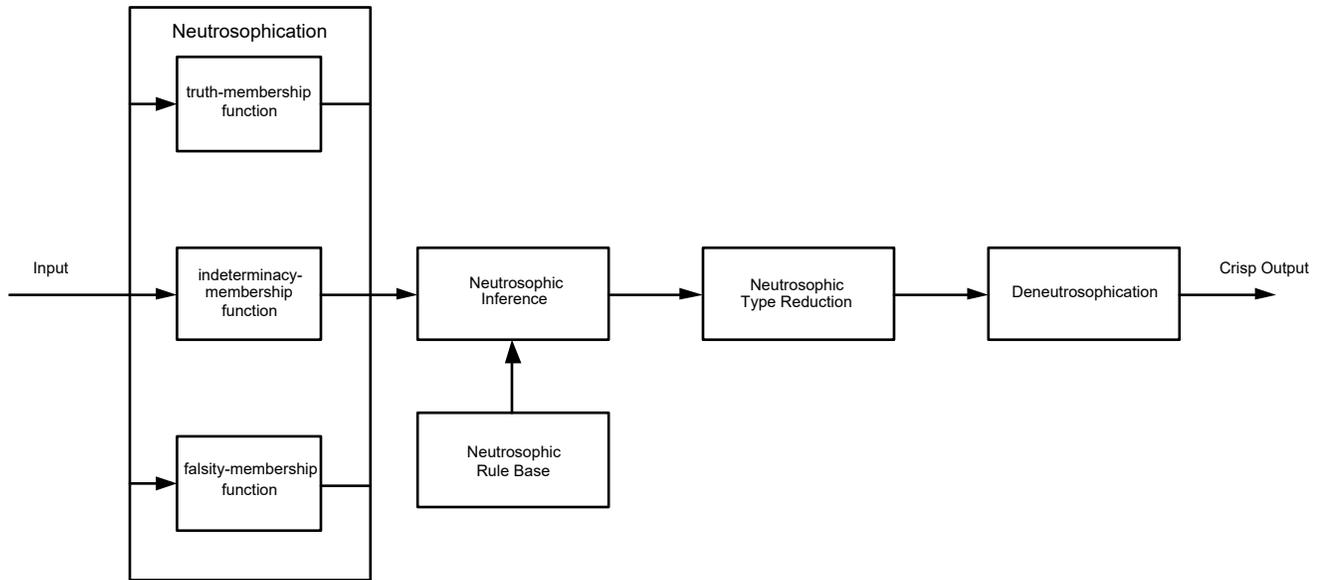

Figure 2.1: General Scheme of an INLS

Suppose the neutrosophic rule base consists of M rules in which each rule has n antecedents and one consequent. Let the kth rule be denoted by R^k such that IF x_1 is A_1^k, x_2 is A_2^k, ..., and x_n is A_n^k,

THEN y is B^k. A_i^k is an interval neutrosophic set defined on universe X_i with truth-membership function $T_{A_i^k}(x_i)$, indeterminacy-membership function $I_{A_i^k}(x_i)$ and falsity-membership function $F_{A_i^k}(x_i)$, where $T_{A_i^k}(x_i), I_{A_i^k}(x_i), F_{A_i^k}(x_i) \subseteq [0,1], 1 \leq i \leq n$. B^k is an interval neutrosophic set defined on universe Y with truth-membership function $T_{B^k}(y)$, indeterminacy-membership function $I_{B^k}(y)$ and falsity-membership function
$F_{B^k}(y)$, where $T_{B^k}(y), I_{B^k}(y), F_{B^k}(y) \subseteq [0,1]$. Given fact x_1 is $\tilde{A}_1^k, x_2$ is $\tilde{A}_2^k, \ldots$, and x_n is $\tilde{A}_n^k$, then consequence y is $\tilde{B}^k$. $\tilde{A}_i^k$ is an interval neutrosophic set defined on universe X_i with truth-membership function $T_{\tilde{A}_i^k}(x_i)$, indeterminacy-membership function $I_{\tilde{A}_i^k}(x_i)$ and falsity-membership function $F_{\tilde{A}_i^k}(x_i)$, where $T_{\tilde{A}_i^k}(x_i), I_{\tilde{A}_i^k}(x_i), F_{\tilde{A}_i^k}(x_i) \subseteq [0,1], 1 \leq i \leq n$. $\tilde{B}^k$ is an interval neutrosophic set defined on universe Y with truth-membership function $T_{\tilde{B}^k}(y)$, indeterminacy-membership function $I_{\tilde{B}^k}(y)$ and falsity-membership function
$F_{\tilde{B}^k}(y)$, where $T_{\tilde{B}^k}(y), I_{\tilde{B}^k}(y), F_{\tilde{B}^k}(y) \subseteq [0,1]$. In this chapter, we consider $a_i \leq x_i \leq b_i$ and $\alpha \leq y \leq \beta$.

An unconditional neutrosophic proposition is expressed by the phrase: "Z is C", where Z is a variable that receives values z from a universal set U, and C is an interval neutrosophic set defined on U that represents a neutrosophic predicate. Each neutrosophic proposition p is associated with $\langle t(p), i(p), f(p) \rangle$ with $t(p), i(p), f(p) \subseteq [0,1]$. In general, for any value z of Z, $\langle t(p), i(p), f(p) \rangle = \langle T_C(z), I_C(z), F_C(z) \rangle$.

For implication $p \rightarrow q$, we define the semantics as:

$$\sup t_{p \rightarrow q} = \min(\sup t(p), \sup t(q)), \tag{2.1}$$
$$\inf t_{p \rightarrow q} = \min(\inf t(p), \inf t(q)), \tag{2.2}$$
$$\sup i_{p \rightarrow q} = \max(\sup i(p), \sup i(q)), \tag{2.3}$$
$$\inf i_{p \rightarrow q} = \max(\inf i(p), \inf i(q)), \tag{2.4}$$
$$\sup f_{p \rightarrow q} = \max(\sup f(p), \sup f(q)), \tag{2.5}$$
$$\inf f_{p \rightarrow q} = \max(\inf f(p), \inf f(q)), \tag{2.6}$$

where $t(p), i(p), f(p), t(q), i(q), f(q) \subseteq [0,1]$.

Let $X = X_1 \times \cdots \times X_n$. The truth-membership function, indeterminacy-membership function and falsity-membership function $T_{\tilde{B}^k}(y), I_{\tilde{B}^k}(y), F_{\tilde{B}^k}(y)$ of a fired kth rule can be represented using the definition of interval neutrosophic composition functions (1.44–1.46) and the semantics of conjunction and disjunction

defined in Table 2.2 and equations (2.1–2.6) as:

$$\sup T_{\tilde{B}^k}(y) = \sup_{x \in X} \min(\sup T_{\tilde{A}_1^k}(x_1), \sup T_{A_1^k}(x_1), \ldots, \sup T_{\tilde{A}_n^k}(x_n), \sup T_{A_n^k}(x_n), \sup T_{B^k}(y)), \quad (2.7)$$

$$\inf T_{\tilde{B}^k}(y) = \sup_{x \in X} \min(\inf T_{\tilde{A}_1^k}(x_1), \inf T_{A_1^k}(x_1), \ldots, \inf T_{\tilde{A}_n^k}(x_n), \inf T_{A_n^k}(x_n), \inf T_{B^k}(y)), \quad (2.8)$$

$$\sup I_{\tilde{B}^k}(y) = \sup_{x \in X} \max(\sup I_{\tilde{A}_1^k}(x_1), \sup I_{A_1^k}(x_1), \ldots, \sup I_{\tilde{A}_n^k}(x_n), \sup I_{A_n^k}(x_n), \sup I_{B^k}(y)), \quad (2.9)$$

$$\inf I_{\tilde{B}^k}(y) = \sup_{x \in X} \max(\inf I_{\tilde{A}_1^k}(x_1), \inf I_{A_1^k}(x_1), \ldots, \inf I_{\tilde{A}_n^k}(x_n), \inf I_{A_n^k}(x_n), \inf I_{B^k}(y)), \quad (2.10)$$

$$\sup F_{\tilde{B}^k}(y) = \inf_{x \in X} \max(\sup F_{\tilde{A}_1^k}(x_1), \sup F_{A_1^k}(x_1), \ldots, \sup F_{\tilde{A}_n^k}(x_n), \sup F_{A_n^k}(x_n), \sup F_{B^k}(y)), (2.11)$$

$$\inf F_{\tilde{B}^k}(y) = \inf_{x \in X} \max(\inf F_{\tilde{A}_1^k}(x_1), \inf F_{A_1^k}(x_1), \ldots, \inf F_{\tilde{A}_n^k}(x_n), \inf F_{A_n^k}(x_n), \inf F_{B^k}(y)), \quad (2.12)$$

where $y \in Y$.

Now, we give the algorithmic description of INLS.

BEGIN

Step 1: Neutrosophication

The purpose of neutrosophication is to map inputs into interval neutrosophic input sets. Let $G_i{}^k$ be an interval neutrosophic input set to represent the result of neutrosophication of ith input variable of kth rule, then

$$\sup T_{G_i{}^k}(x_i) = \sup_{x_i \in X_i} \min(\sup T_{\tilde{A}_i^k}(x_i), \sup T_{A_i^k}(x_i)), \quad (2.13)$$

$$\inf T_{G_i^k}(x_i) = \sup_{x_i \in X_i} \min(\inf T_{\tilde{A}_i^k}(x_i), \inf T_{A_i^k}(x_i)), \quad (2.14)$$

$$\sup I_{G_i^k}(x_i) = \sup_{x_i \in X_i} \max(\sup I_{\tilde{A}_i^k}(x_i), \sup I_{A_i^k}(x_i)), \quad (2.15)$$

$$\inf I_{G_i^k}(x_i) = \sup_{x_i \in X_i} \max(\inf I_{\tilde{A}_i^k}(x_i), \inf I_{A_i^k}(x_i)), \quad (2.16)$$

$$\sup F_{G_i^k}(x_i) = \inf_{x_i \in X_i} \max(\sup F_{\tilde{A}_i^k}(x_i), \sup F_{A_i^k}(x_i)), \quad (2.17)$$

$$\inf F_{G_i^k}(x_i) = \inf_{x_i \in X_i} \max(\inf F_{\tilde{A}_i^k}(x_i), \inf F_{A_i^k}(x_i)), \quad (2.18)$$

where $x_i \in X_i$.

If x_i are crisp inputs, then equations (50–55) are simplified to

$$\sup T_{G_i^k}(x_i) = \sup T_{A_i^k}(x_i), \quad (2.19)$$

$$\inf T_{G_i^k}(x_i) = \inf T_{A_i^k}(x_i), \quad (2.20)$$

$$\sup I_{G_i^k}(x_i) = \sup I_{A_i^k}(x_i), \quad (2.21)$$

$$\inf I_{G_i^k}(x_i) = \inf I_{A_i^k}(x_i), \quad (2.22)$$

$$\sup F_{G_i^k}(x_i) = \sup F_{A_i^k}(x_i), \quad (2.23)$$

$$\inf F_{G_i^k}(x_i) = \inf F_{A_i^k}(x_i), \quad (2.24)$$

where $x_i \in X_i$.

Fig. 2 shows the conceptual diagram for neutrosophication of a crisp input x_1.
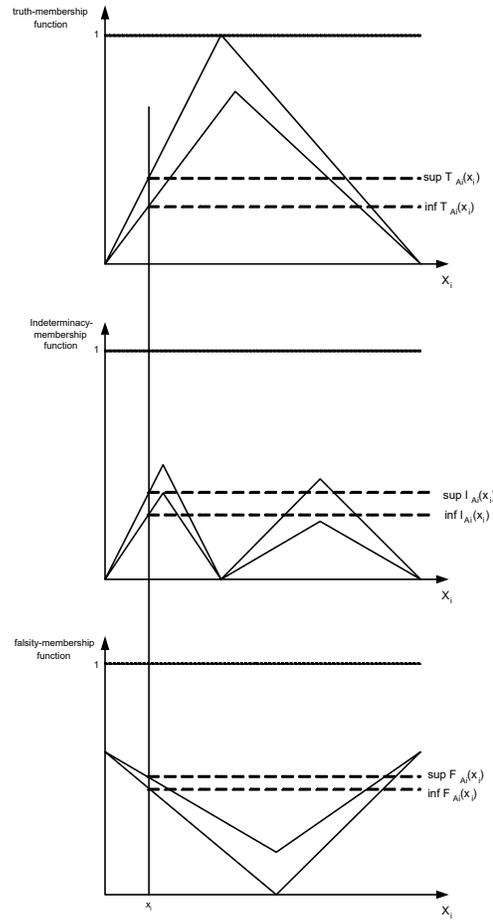
Figure 2.2: Conceptual Diagram for Neutrosophication of Crisp Input

Step 2: Neutrosophic Inference

The core of INLS is the neutrosophic inference, the principle of which has already been explained above. Suppose the kth rule is fired. Let G^k be an interval neutrosophic set to represent the result of the input

and antecedent operation for kth rule, then

$$\sup T_{G^k}(x) = \sup_{x\in X} \min(\sup T_{\tilde{A}_1^k}(x_1), \sup T_{A_1^k}(x_1), \ldots, \sup T_{\tilde{A}_n^k}(x_n), \sup T_{A_n^k}(x_n)), \qquad (2.25)$$

$$\inf T_{G^k}(x) = \sup_{x\in X} \min(\inf T_{\tilde{A}_1^k}(x_1), \inf T_{A_1^k}(x_1), \ldots, \inf T_{\tilde{A}_n^k}(x_n), \inf T_{A_n^k}(x_n)), \qquad (2.26)$$

$$\sup I_{G^k}(x) = \sup_{x\in X} \max(\sup I_{\tilde{A}_1^k}(x_1), \sup I_{A_1^k}(x_1), \ldots, \sup I_{\tilde{A}_n^k}(x_n), \sup I_{A_n^k}(x_n)), \qquad (2.27)$$

$$\inf I_{G^k}(x) = \sup_{x\in X} \max(\inf I_{\tilde{A}_1^k}(x_1), \inf I_{A_1^k}(x_1), \ldots, \inf I_{\tilde{A}_n^k}(x_n), \inf I_{A_n^k}(x_n)), \qquad (2.28)$$

$$\sup F_{G^k}(x) = \inf_{x\in X} \max(\sup F_{\tilde{A}_1^k}(x_1), \sup F_{A_1^k}(x_1), \ldots, \sup F_{\tilde{A}_n^k}(x_n), \sup F_{A_n^k}(x_n)), \qquad (2.29)$$

$$\inf F_{G^k}(x) = \inf_{x\in X} \max(\inf F_{\tilde{A}_1^k}(x_1), \inf F_{A_1^k}(x_1), \ldots, \inf F_{\tilde{A}_n^k}(x_n), \inf F_{A_n^k}(x_n)), \qquad (2.30)$$

where $x_i \in X_i$.

Here we restate the result of neutrosophic inference:

$$\sup T_{\tilde{B}^k}(y) = \min(\sup T_{G_1^k}(x), \sup T_{B^k}(y)), \qquad (2.31)$$

$$\inf T_{\tilde{B}^k}(y) = \min(\inf T_{G^k}(x), \inf T_{B^k}(y)), \qquad (2.32)$$

$$\sup I_{\tilde{B}^k}(y) = \max(\sup I_{G^k}(x), \sup I_{B^k}(y)), \qquad (2.33)$$

$$\inf I_{\tilde{B}^k}(y) = \max(\inf I_{G^k}(x), \inf I_{B^k}(y)), \qquad (2.34)$$

$$\sup F_{\tilde{B}^k}(y) = \max(\sup F_{G^k}(x), \sup F_{B^k}(y)), \qquad (2.35)$$

$$\inf F_{\tilde{B}^k}(y) = \max(\inf F_{G^k}(x), \inf F_{B^k}(y)), \qquad (2.36)$$

where $x \in X, y \in Y$.

Suppose that N rules in the neutrosophic rule base are fired, where $N \leq M$, then, the output interval neutrosophic set $\tilde{B}$ is:

$$\sup T_{\tilde{B}}(y) = \max_{k=1}^{N} \sup T_{\tilde{B}^k}(y), \qquad (2.37)$$

$$\inf T_{\tilde{B}}(y) = \max_{k=1}^{N} \inf T_{\tilde{B}^k}(y), \qquad (2.38)$$

$$\sup I_{\tilde{B}}(y) = \min_{k=1}^{N} \sup I_{\tilde{B}^k}(y), \qquad (2.39)$$

$$\inf I_{\tilde{B}}(y) = \min_{k=1}^{N} \inf I_{\tilde{B}^k}(y), \qquad (2.40)$$

$$\sup F_{\tilde{B}}(y) = \min_{k=1}^{N} \sup T_{\tilde{B}^k}(y), \qquad (2.41)$$

$$\inf T_{\tilde{B}}(y) = \min_{k=1}^{N} \inf T_{\tilde{B}^k}(y), \qquad (2.42)$$

where $y \in Y$.

Step 3: Neutrosophic type reduction

After neutrosophic inference, we will get an interval neutrosophic set $\tilde{B}$ with $T_{\tilde{B}}(y), I_{\tilde{B}}(y), F_{\tilde{B}}(y) \subseteq [0,1]$. Then, we do the neutrosophic type reduction to transform each interval into one number. There are many ways to do it, here, we give one method:

$$T'_{\tilde{B}}(y) = (\inf T_{\tilde{B}}(y) + \sup T_{\tilde{B}}(y))/2, \tag{2.43}$$

$$I'_{\tilde{B}}(y) = (\inf I_{\tilde{B}}(y) + \sup I_{\tilde{B}}(y))/2, \tag{2.44}$$

$$F'_{\tilde{B}}(y) = (\inf F_{\tilde{B}}(y) + \sup F_{\tilde{B}}(y))/2, \tag{2.45}$$

where $y \in Y$.

So, after neutrosophic type reduction, we will get an ordinary neutrosophic set (a type-1 neutrosophic set) $\tilde{B}$. Then we need to do the deneutrosophication to get a crisp output.

Step 4: Deneutrosophication

The purpose of deneutrosophication is to convert an ordinary neutrosophic set (a type-1 neutrosophic set) obtained by neutrosophic type reduction to a single real number which represents the real output. Similar to defuzzification [KY95], there are many deneutrosophication methods according to different applications. Here we give one method. The deneutrosophication process consists of two steps.

Step 4.1: *Synthesization*: It is the process to transform an ordinary neutrosophic set (a type-1 neutrosophic set) $\tilde{B}$ into a fuzzy set $\bar{B}$. It can be expressed using the following function:

$$f(T'_{\tilde{B}}(y), I'_{\tilde{B}}(y), F'_{\tilde{B}}(y)) : [0,1] \times [0,1] \times [0,1] \to [0,1] \tag{2.46}$$

Here we give one definition of f:

$$T_{\bar{B}}(y) = a * T'_{\tilde{B}}(y) + b * (1 - F'_{\tilde{B}}(y)) + c * I'_{\tilde{B}}(y)/2 + d * (1 - I'_{\tilde{B}}(y)/2), \tag{2.47}$$

where $0 \leq a, b, c, d \leq 1, a + b + c + d = 1$.

The purpose of synthesization is to calculate the overall truth degree according to three components: truth-membership function, indeterminacy-membership function and falsity-membership function. The component–truth-membership function gives the direct information about the truth-degree, so we use it directly in the formula; The component–falsity-membership function gives the indirect information about the truth-degree, so we use $(1-F)$ in the formula. To understand the meaning of indeterminacy-membership function I, we give an example: a statement is "The quality of service is good", now firstly a person has to select a decision among $\{T, I, F\}$, secondly he or she has to answer the degree of the decision in $[0,1]$. If he or she chooses $I = 1$, it means 100% "not sure" about the statement, i.e., 50% true and 50% false for the statement (100% balanced), in this sense, $I = 1$ contains the potential truth value 0.5. If he or she chooses $I = 0$, it means 100% "sure" about the statement, i.e., either 100% true or 100% false for the statement (0% balanced), in this sense, $I = 0$ is related to two extreme cases, but we do not know which one is in his or her mind. So we have to consider both at the same time: $I = 0$ contains the potential

truth value that is either 0 or 1. If I decreases from 1 to 0, then the potential truth value changes from one value 0.5 to two different possible values gradually to the final possible ones 0 and 1 (i.e., from 100% balanced to 0% balanced), since he or she does not choose either T or F but I, we do not know his or her final truth value. Therefore, the formula has to consider two potential truth values implicitly represented by I with different weights (c and d) because of lack of his or her final decision information after he or she has chosen I. Generally, $a > b > c, d$; c and d could be decided subjectively or objectively as long as enough information is available. The parameters a, b, c and d can be tuned using learning algorithms such as neural networks and genetic algorithms in the development of application to improve the performance of the INLS.

Step 4.2: *Calculation of a typical neutrosophic value*: Here we introduce one method of calculation of center of area. The method is sometimes called the *center of gravity method* or *centroid method*, the deneutrosophicated value, $dn(T_{\bar{B}}(y))$ is calculated by the formula

$$dn(T_{\bar{B}}(y)) = \frac{\int_\alpha^\beta T_{\bar{B}}(y)y\,dy}{\int_\alpha^\beta T_{\bar{B}}\,dy}. \tag{2.48}$$

END.

2.5   Conclusions

In this chapter, we give the formal definitions of interval neutrosophic logic which are extension of many other classical logics such as fuzzy logic, intuitionistic fuzzy logic and paraconsistent logics, etc. Interval neutrosophic logic include interval neutrosophic propositional logic and first order interval neutrosophic predicate logic. We call them classical (standard) neutrosophic logic. In the future, we also will discuss and explore the non-classical (non-standard) neutrosophic logic such as modal interval neutrosophic logic, temporal interval neutrosophic logic, etc. Interval neutrosophic logic can not only handle imprecise, fuzzy and incomplete propositions but also inconsistent propositions without the danger of trivialization. The chapter also give one application based on the semantic notion of interval neutrosophic logic – the Interval Neutrosophic Logic Systems (INLS) which is the generalization of classical FLS and interval valued fuzzy FLS. Interval neutrosophic logic will have a lot of potential applications in computational Web intelligence [ZKLY04]. For example, current fuzzy Web intelligence techniques can be improved by using more reliable interval neutrosophic logic methods because T, I and F are all used in decision making. In large, such robust interval neutrosophic logic methods can also be used in other applications such as medical informatics, bioinformatics and human-oriented decision-making under uncertainty. In fact, interval neutrosophic sets and interval neutrosophic logic could be applied in the fields that fuzzy sets and fuzz logic are suitable for, also the fields that paraconsistent logics are suitable for.

Chapter 3

Neutrosophic Relational Data Model

In this chapter, we present a generalization of the relational data model based on interval neutrosophic sets. Our data model is capable of manipulating incomplete as well as inconsistent information. Fuzzy relation or intuitionistic fuzzy relation can only handle incomplete information. Associated with each relation are two membership functions one is called truth-membership function T which keeps track of the extent to which we believe the tuple is in the relation, another is called falsity-membership function which keeps track of the extent to which we believe that it is not in the relation. A neutrosophic relation is inconsistent if there exists one tuple a such that $T(a) + F(a) > 1$. In order to handle inconsistent situation, we propose an operator called "split" to transform inconsistent neutrosophic relations into pseudo-consistent neutrosophic relations and do the set-theoretic and relation-theoretic operations on them and finally use another operator called "combine" to transform the result back to neutrosophic relation. For this model, we define algebraic operators that are generalisations of the usual operators such as interesection, union, selection, join on fuzzy relations. Our data model can underlie any database and knowledge-base management system that deals with incomplete and inconsistent information.

3.1   Introduction

Relational data model was proposed by Ted Codd's pioneering paper [Cod70]. Since then, relational database systems have been extensively studied and a lot of commercial relational database systems are currently available [EN00, SKS96]. This data model usually takes care of only well-defined and unambiguous data. However, imperfect information is ubiquitous – almost all the information that we have about the real world is not certain, complete and precise [Par96]. Imperfect information can be classified as: incompleteness, imprecision, uncertainty, inconsistency. Incompleteness arises from the absence of a value, imprecision from the existence of a value which cannot be measured with suitable precision, uncertainty from the fact that a person has given a subjective opinion about the truth of a fact which he/she does not know for certain, and inconsistency from the fact that there are two or more conflicting values for a variable.

In order to represent and manipulate various forms of incomplete information in relational databases,

several extensions of the classical relational model have been proposed [Bis83, BMS84, Cod79, Lip79, Lip81, Mai83]. In some of these extensions, a variety of "null values" have been introduced to model unknown or not-applicable data values. Attempts have also been made to generalize operators of relational algebra to manipulate such extended data models [Bis83, Cod79, Mai83, LS90, LS91]. The fuzzy set theory and fuzzy logic proposed by Zadeh [Zad65a] provide a requisite mathematical framework for dealing with incomplete and imprecise information. Later on, the concept of interval-valued fuzzy sets was proposed to capture the fuzziness of grade of membership itself [Tur86]. In 1986, Atanassov introduced the intuitionistic fuzzy set [Ata86] which is a generalization of fuzzy set and provably equivalent to interval-valued fuzzy set. The intuitionistic fuzzy sets consider both truth-membership T and falsity-membership F with $T(a), F(a) \in [0,1]$ and $T(a) + F(a) \leq 1$. Because of the restriction, the fuzzy set, interval-valued fuzzy set and intuitionistic fuzzy set cannot handle inconsistent information. Some authors [AR84, Bal83, BP82, CK78, KZ86, Pra84, RM88] have studied relational databases in the light of fuzzy set theory with an objective to accommodate a wider range of real-world requirements and to provide closer man-machine interactions. Probability, possibility and Dempster-Shafer theory have been proposed to deal with uncertainty. Possibility theory [Zad78] is built upon the idea of a fuzzy restriction. That means a variable could only take its value from some fuzzy set of values and any value within that set is a possible value for the variable. Because values have different degrees of membership in the set, they are possible to different degrees. Prade and Testemale [PT84] initially suggested using possibility theory to deal with incomplete and uncertain information in database. Their work is extended in [PT87] to cover multivalued attributes. Wong [Won82] proposes a method that quantifies the uncertainty in a database using probabilities. His method maybe is the simplest one which attached a probability to every member of a relation, and to use these values to provide the probability that a particular value is the correct answer to a particular query. Carvallo and Pittarelli [CP87] also use probability theory to model uncertainty in relational databases systems. Their method augmented projection an join operations with probability measures.

However, unlike incomplete, imprecise and uncertain information, inconsistent information has not enjoyed enough research attention. In fact, inconsistent information exists in a lot of applications. For example, in data warehousing application, inconsistency will appear when trying to integrate the data from many different sources. Another example is that in the expert system, there exist facts which are inconsistent with each other. Generally, two basic approaches have been followed in solving the inconsistency problem in knowledge bases: belief revision and paraconsistent logic. The goal of the first approach is to make an inconsistent theory consistent, either by revising it or by representing it by a consistent semantics. On the other hand, the paraconsistent approach allows reasoning in the presence of inconsistency, and contradictory information can be derived or introduced without trivialization [dACM02b]. Bagai and Sunderraman [BS95, SB95] proposed a paraconsistent relational data model to deal with incomplete and inconsistent information. The data model has been applied to compute the well-founded and fitting model of logic programming [BS96b, BS96a]. This data model is based on paraconsistent logics which were studied in detail by de Costa [Cos77a] and Belnap [Bel77b].

In this chapter, we present a new relational data model – neutrosophic relational data model (NRDM). Our model is based on the neutrosophic set theory which is an extension of intuitionistic fuzzy set

theory[GB93] and is capable of manipulating incomplete as well as inconsistent information. We use both truth-membership function grade α and falsity-membership function grade β to denote the status of a tuple of a certain relation with $\alpha, \beta \in [0, 1]$ and $\alpha + \beta \leq 2$. NRDM is the generalization of fuzzy relational data model (FRDM). That is, when $\alpha + \beta = 1$, neutrosophic relation is the ordinary fuzzy relation. This model is distinct with paraconsistent relational data model (PRDM), in fact it can be easily shown that PRDM is a special case of PIFRDM. That is when $\alpha, \beta = 0$ or 1, neutrosophic relation is just paraconsistent relation. We can use Figure 3.1 to express the relationship among FRDM, PRDM and PIFRDM.
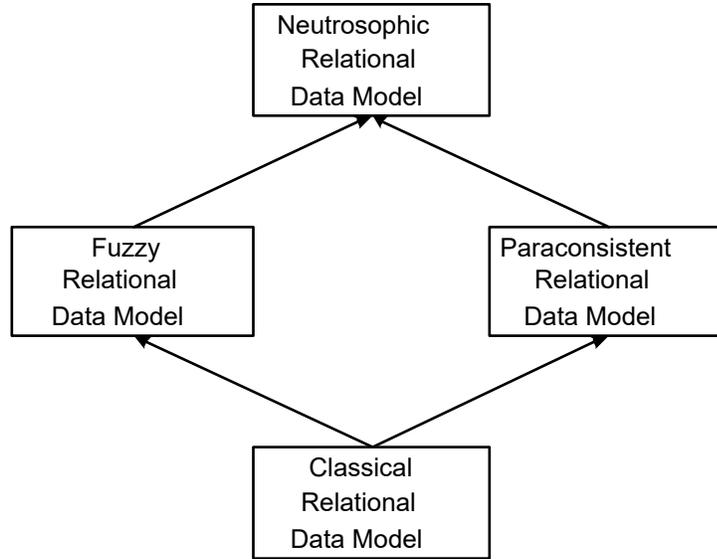
Figure 3.1: Relationship Among FRDM, PRDM, NRDM and RDM

We introduce neutrosophic relations, which are the fundamental mathematical structures underlying our model. These structures are strictly more general than classical fuzzy relations and intuitionistic fuzzy relations (interval-valued fuzzy relations), in that for any fuzzy relation or intuitionistic fuzzy relation (interval-valued fuzzy relation) there is a neutrosophic relation with the same information content, but not *vice versa*. The claim is also true for the relationship between neutrosophic relations and paraconsistent relations. We define algebraic operators over neutrosophic relations that extend the standard operators such as selection, join, union over fuzzy relations.

There are many potential applications of our new data model. Here are some examples:

(a) Web mining. Essentially the data and documents on the Web are heterogeneous, inconsistency is unavoidable. Using the presentation and reasoning method of our data model, it is easier to capture imperfect information on the Web which will provide more potentially value-added information.

(b) Bioinformatics. There is a proliferation of data sources. Each research group and each new exper-
imental technique seems to generate yet another source of valuable data. But these data can be
incomplete and imprecise and even inconsistent. We could not simply throw away one data in favor
of other data. So how to represent and extract useful information from these data will be a challenge
problem.

(c) Decision Support System. In decision support system, we need to combine the database with the
knowledge base. There will be a lot of uncertain and inconsistent information, so we need an efficient
data model to capture these information and reasoning with these information.

The chapter is organized as follow. Section 3.2 of the chapter deals with some of the basic definitions and
concepts of fuzzy relations and operations. Section 3.3 introduces neutrosophic relations and two notions
of generalising the fuzzy relational operators such as union, join, projection for these relations. Section 3.4
presents some actual generalised algebraic operators for neutrosophic relations. These operators can be
used for sepcifying queries for database systems built on such relations. Section 3.5 gives an illustrative
application of these operators. Finally, Section 3.8 contains some concluding remarks and directions for
future work.

3.2   Fuzzy Relations and Operations

In this section, we present the essential concepts of a fuzzy relational database. Fuzzy relations associate a
value between 0 and 1 with every tuple representing the degree of membership of the tuple in the relation.
We also present several useful query operators on fuzzy relations.

Let a *relation scheme* (or just *scheme*) Σ be a finite set of *attribute names*, where for any attribute name
$A \in \Sigma$, $dom(A)$ is a non-empty *domain* of values for A. A *tuple* on Σ is any map $t : \Sigma \to \cup_{A \in \Sigma} dom(A)$,
such that $t(A) \in dom(A)$, for each $A \in \Sigma$. Let $\tau(\Sigma)$ denote the set of all tuples on Σ.

Definition 45 *A* fuzzy relation *on scheme* Σ *is any map* $R : \tau(\Sigma) \to [0,1]$. *We let* $\mathcal{F}(\Sigma)$ *be the set of all
fuzzy relations on* Σ. $\square$

If Σ and Δ are relation schemes such that $\Delta \subseteq \Sigma$, then for any tuple $t \in \tau(\Delta)$, we let t^Σ denote the set
$\{t' \in \tau(\Sigma) \mid t'(A) = t(A),$ for all $A \in \Delta\}$ of all extensions of t. We extend this notion for any $T \subseteq \tau(\Delta)$ by
defining $T^\Sigma = \cup_{t \in T} t^\Sigma$.

3.2.1   Set-theoretic operations on Fuzzy relations

Definition 46 Union: *Let R and S be fuzzy relations on scheme* Σ. *Then, $R \cup S$ is a fuzzy relation on
scheme* Σ *given by*

$$(R \cup S)(t) = \max\{R(t), S(t)\}, for \ any \ t \in \tau(\Sigma).\square$$

Definition 47 Complement: *Let R be a fuzzy relation on scheme Σ. Then, $-R$ is a fuzzy relation on scheme Σ given by*

$$(-R)(t) = 1 - R(t), \text{for any } t \in \tau(\Sigma). \square$$

Definition 48 Intersection: *Let R and S be fuzzy relations on scheme Σ. Then, $R \cap S$ is a fuzzy relation on scheme Σ given by*

$$(R \cap S)(t) = \min\{R(t), S(t)\}, \text{for any } t \in \tau(\Sigma). \square$$

Definition 49 Difference: *Let R and S be fuzzy relations on scheme Σ. Then, $R - S$ is a fuzzy relation on scheme Σ given by*

$$(R - S)(t) = \min\{R(t), 1 - S(t)\}, \text{for any } t \in \tau(\Sigma). \square$$

3.2.2 Relation-theoretic operations on Fuzzy relations

Definition 50 *Let R and S be fuzzy relations on schemes Σ and Δ, respectively. Then, the* natural join *(or just* join*) of R and S, denoted $R \bowtie S$, is a fuzzy relation on scheme $\Sigma \cup \Delta$, given by*

$$(R \bowtie S)(t) = \min\{R(\pi_\Sigma(t)), S(\pi_\Delta(t))\}, \text{for any } t \in \tau(\Sigma \cup \Delta). \square$$

Definition 51 *Let R be a fuzzy relation on scheme Σ and let $\Delta \subseteq \Sigma$. Then, the projection of R onto Δ, denoted by $\Pi_\Delta(R)$ is a fuzzy relation on scheme Δ given by*

$$(\Pi_\Delta(R))(t) = \max\{R(u)|u \in t^\Sigma\}, \text{for any } t \in \tau(\Delta). \square$$

Definition 52 *Let R be a fuzzy relation on scheme Σ, and let F be any logic formula involving attribute names in Σ, constant symbols (denoting values in the attribute domains), equality symbol $=$, negation symbol $\neg$, and connectives $\vee$ and $\wedge$. Then, the selection of R by F, denoted $\dot{\sigma}_F(R)$, is a fuzzy relation on scheme Σ, given by*

$$(\dot{\sigma}_F(R))(t) = \begin{cases} R(t) & \text{if } t \in \sigma_F(\tau(\Sigma)) \\ 0 & \text{Otherwise} \end{cases}$$

where σ_F is the usual selection of tuples satisfying F. $\square$

3.3 Neutrosophic Relations

In this section, we generalize fuzzy relations in such a manner that we are now able to assign a measure of belief and a measure of doubt to each tuple. We shall refer to these generalized fuzzy relations as *neutrosophic relations.* So, a tuple in a neutrosophic relation is assigned a measure $\langle \alpha, \beta \rangle$, $0 \leq \alpha, \beta \leq 1$. α will be referred to as the *belief* factor and β will be referred to as the *doubt* factor. The interpretation of this measure is that we believe with confidence α and doubt with confidence β that the tuple is in the relation. The belief and doubt confidence factors for a tuple need not add to exactly 1. This allows for

incompleteness and inconsistency to be represented. If the belief and doubt factors add up to less than 1, we have incomplete information regarding the tuple's status in the relation and if the belief and doubt factors add up to more than 1, we have inconsistent information regarding the tuple's status in the relation.

In contrast to fuzzy relations where the grade of membership of a tuple is fixed, neutrosophic relations bound the grade of membership of a tuple to a subinterval $[\alpha, 1 - \beta]$ for the case $\alpha + \beta \leq 1$.

The operators on fuzzy relations can also be generalised for neutrosophic relations. However, any such generalization of operators should maintain the belief system intuition behind neutrosophic relations.

This section also develops two different notions of operator generalisations.

We now formalize the notion of a neutrosophic relation.

Recall that $\tau(\Sigma)$ denotes the set of all tuples on any scheme Σ.

Definition 53 *A **neutrosophic relation** R on scheme Σ is any subset of*

$$\tau(\Sigma) \times [0, 1] \times [0, 1].$$

For any $t \in \tau(\Sigma)$, we shall denote an element of R as $\langle t, R(t)^+, R(t)^- \rangle$, where $R(t)^+$ is the belief factor assigned to t by R and $R(t)^-$ is the doubt factor assigned to t by R. Let $\mathcal{V}(\Sigma)$ be the set of all neutrosophic relations on Σ.

Definition 54 *A neutrosophic relation R on scheme Σ is **consistent** if $R(t)^+ + R(t)^- \leq 1$, for all $t \in \tau(\Sigma)$. Let $\mathcal{C}(\Sigma)$ be the set of all consistent neutrosophic relations on Σ. R is said to be **complete** if $R(t)^+ + R(t)^- \geq 1$, for all $t \in \tau(\Sigma)$. If R is both consistent and complete, i.e. $R(t)^+ + R(t)^- = 1$, for all $t \in \tau(\Sigma)$, then it is a **total** neutrosophic relation, and let $\mathcal{T}(\Sigma)$ be the set of all total neutrosophic relations on Σ.*

Definition 55 *R is said to be **pseudo-consistent** if*
$\max\{b_i | (\exists t \in \tau(\Sigma))(\exists d_i)(\langle t, b_i, d_i \rangle \in R)\} + \max\{d_i | (\exists t \in \tau(\Sigma))(\exists b_i)(\langle t, b_i, d_i \rangle \in R)\} > 1$, *where for these* $\langle t, b_i, d_i \rangle, b_i + d_i = 1$. *Let $\mathcal{P}(\Sigma)$ be the set of all pseudo-consistent neutrosophic relations on Σ.*

Example 16 *Neutrosophic relation $R = \{\langle a, 0.3, 0.7 \rangle,$*
$\langle a, 0.4, 0.6 \rangle, \langle b, 0.2, 0.5 \rangle, \langle c, 0.4, 0.3 \rangle\}$ is pseudo-consistent. Because for $t = a$, $\max\{0.3, 0.4\} + \max\{0.7, 0.6\} = 1.1 > 1$.

It should be observed that total neutrosophic relations are essentially fuzzy relations where the uncertainty in the grade of membership is eliminated. We make this relationship explicit by defining a one-one correspondence $\lambda_\Sigma : \mathcal{T}(\Sigma) \to \mathcal{F}(\Sigma)$, given by $\lambda_\Sigma(R)(t) = R(t)^+$, for all $t \in \tau(\Sigma)$. This correspondence is used frequently in the following discussion.

Operator Generalisations

It is easily seen that neutrosophic relations are a generalization of fuzzy relations, in that for each fuzzy relation there is a neutrosophic relation with the same information content, but not *vice versa*. It is thus

natural to think of generalising the operations on fuzzy relations such as union, join, projection etc. to neutrosophic relations. However, any such generalization should be intuitive with respect to the belief system model of neutrosophic relations. We now construct a framework for operators on both kinds of relations and introduce two different notions of the generalization relationship among their operators.

An n-ary *operator on fuzzy relations with signature* $\langle \Sigma_1, \ldots, \Sigma_{n+1} \rangle$ is a function $\Theta : \mathcal{F}(\Sigma_1) \times \cdots \times \mathcal{F}(\Sigma_n) \to \mathcal{F}(\Sigma_{n+1})$, where $\Sigma_1, \ldots, \Sigma_{n+1}$ are any schemes. Similarly, an n-ary *operator on neutrosophic relations with signature* $\langle \Sigma_1, \ldots, \Sigma_{n+1} \rangle$ is a function $\Psi : \mathcal{V}(\Sigma_1) \times \cdots \times \mathcal{V}(\Sigma_n) \to \mathcal{V}(\Sigma_{n+1})$.

Definition 56 *An operator Ψ on neutrosophic relations with signature $\langle \Sigma_1, \ldots, \Sigma_{n+1} \rangle$ is* totality preserving *if for any total neutrosophic relations $R_1, \ldots, R_n$ on schemes $\Sigma_1, \ldots, \Sigma_n$, respectively, $\Psi(R_1, \ldots, R_n)$ is also total.* $\square$

Definition 57 *A totality preserving operator Ψ on neutrosophic relations with signature*

$$\langle \Sigma_1, \ldots, \Sigma_{n+1} \rangle$$

is a weak generalization *of an operator Θ on fuzzy relations with the same signature, if for any total neutrosophic relations $R_1, \ldots, R_n$ on schemes $\Sigma_1, \ldots, \Sigma_n$, respectively, we have*

$$\lambda_{\Sigma_{n+1}}(\Psi(R_1, \ldots, R_n)) = \Theta(\lambda_{\Sigma_1}(R_1), \ldots, \lambda_{\Sigma_n}(R_n)).\square$$

The above definition essentially requires Ψ to coincide with Θ on total neutrosophic relations (which are in one-one correspondence with the fuzzy relations). In general, there may be many operators on neutrosophic relations that are weak generalisations of a given operator Θ on fuzzy relations. The behavior of the weak generalisations of Θ on even just the consistent neutrosophic relations may in general vary. We require a stronger notion of operator generalization under which, at least when restricted to consistent intuitionistic fuzzy relations, the behavior of all the generalised operators is the same. Before we can develop such a notion, we need that of 'representations' of a neutrosophic relation.

We associate with a consistent neutrosophic relation R the set of all (fuzzy relations corresponding to) total neutrosophic relations obtainable from R by filling in the gaps between the belief and doubt factors for each tuple. Let the map $\mathbf{reps}_\Sigma : \mathcal{C}(\Sigma) \to 2^{\mathcal{F}(\Sigma)}$ be given by

$$\mathbf{reps}_\Sigma(R) = \{Q \in \mathcal{F}(\Sigma) \mid \bigwedge_{t_i \in \tau(\Sigma)} (R(t_i)^+ \leq Q(t_i) \leq 1 - R(t_i)^-)\}.$$

The set $\mathbf{reps}_\Sigma(R)$ contains all fuzzy relations that are 'completions' of the consistent neutrosophic relation R. Observe that $\mathbf{reps}_\Sigma$ is defined only for consistent neutrosophic relations and produces sets of fuzzy relations. Then we have following observation.

Proposition 1 *For any consistent neutrosophic relation R on scheme Σ, $\mathbf{reps}_\Sigma(R)$ is the singleton $\{\lambda_\Sigma(R)\}$ iff R is total.* $\square$

Proof It is clear from the definition of consistent and total neutrosophic relations and from the definition of **reps** operation.

We now need to extend operators on fuzzy relations to sets of fuzzy relations. For any operator $\Theta : \mathcal{F}(\Sigma_1) \times \cdots \times \mathcal{F}(\Sigma_n) \to \mathcal{F}(\Sigma_{n+1})$ on fuzzy relations, we let $\mathcal{S}(\Theta) : 2^{\mathcal{F}(\Sigma_1)} \times \cdots \times 2^{\mathcal{F}(\Sigma_n)} \to 2^{\mathcal{F}(\Sigma_{n+1})}$ be a map on sets of fuzzy relations defined as follows. For any sets $M_1, \ldots, M_n$ of fuzzy relations on schemes $\Sigma_1, \ldots, \Sigma_n$, respectively,

$$\mathcal{S}(\Theta)(M_1, \ldots, M_n) = \{\Theta(R_1, \ldots, R_n) \mid R_i \in M_i, \text{ for all } i, 1 \leq i \leq n\}.$$

In other words, $\mathcal{S}(\Theta)(M_1, \ldots, M_n)$ is the set of Θ-images of all tuples in the cartesian product $M_1 \times \cdots \times M_n$. We are now ready to lead up to a stronger notion of operator generalization.

Definition 58 *An operator Ψ on neutrosophic relations with signature $\langle \Sigma_1, \ldots, \Sigma_{n+1} \rangle$ is* consistency preserving *if for any consistent neutrosophic relations $R_1, \ldots, R_n$ on schemes $\Sigma_1, \ldots, \Sigma_n$, respectively, $\Psi(R_1, \ldots, R_n)$ is also consistent.* $\square$

Definition 59 *A consistency preserving operator Ψ on neutrosophic relations with signature $\langle \Sigma_1, \ldots, \Sigma_{n+1} \rangle$ is a* strong generalization *of an operator Θ on fuzzy relations with the same signature, if for any consistent neutrosophic relations $R_1, \ldots, R_n$ on schemes $\Sigma_1, \ldots, \Sigma_n$, respectively, we have*

$$\mathbf{reps}_{\Sigma_{n+1}}(\Psi(R_1, \ldots, R_n)) = \mathcal{S}(\Theta)(\mathbf{reps}_{\Sigma_1}(R_1), \ldots, \mathbf{reps}_{\Sigma_n}(R_n)).\square$$

Given an operator Θ on fuzzy relations, the behavior of a weak generalization of Θ is 'controlled' only over the total neutrosophic relations. On the other hand, the behavior of a strong generalization is 'controlled' over all consistent neutrosophic relations. This itself suggests that strong generalization is a stronger notion than weak generalization. The following proposition makes this precise.

Proposition 2 *If Ψ is a strong generalization of Θ, then Ψ is also a weak generalization of Θ.*$\square$

Proof Let $\langle \Sigma_1, \ldots, \Sigma_{n+1} \rangle$ be the signature of Ψ and Θ, and let $R_1, \ldots, R_n$ be any total neutrosophic relations on schemes $\Sigma_1, \ldots, \Sigma_n$, respectively. Since all total relations are consistent, and Ψ is a strong generalization of Θ, we have that

$$\mathbf{reps}_{\Sigma_{n+1}}(\Psi(R_1, \ldots, R_n)) = \mathcal{S}(\Theta)(\mathbf{reps}_{\Sigma_1}(R_1), \ldots, \mathbf{reps}_{\Sigma_n}(R_n)),$$

Proposition 1 gives us that for each i, $1 \leq i \leq n$, $\mathbf{reps}_{\Sigma_i}(R_i)$ is the singleton set $\{\lambda_{\Sigma_i}(R_i)\}$. Therefore, $\mathcal{S}(\Theta)(\mathbf{reps}_{\Sigma_1}(R_i), \ldots, \mathbf{reps}_{\Sigma_n}(R_n))$ is just the singleton set:

$$\{\Theta(\lambda_{\Sigma_1}(R_1), \ldots, \lambda_{\Sigma_n}(R_n))\}.$$

Here, $\Psi(R_1, \ldots, R_n)$ is total, and
$\lambda_{\Sigma_{n+1}}(\Psi(R_1, \ldots, R_n)) = \Theta(\lambda_{\Sigma_1}(R_1), \ldots, \lambda_{\Sigma_n}(R_n))$, i.e. Ψ is a weak generalization of Θ.

Though there may be many strong generalisations of an operator on fuzzy relations, they all behave the same when restricted to consistent neutrosophic relations. In the next section, we propose strong generalisations for the usual operators on fuzzy relations. The proposed generalised operators on neutrosophic relations correspond to the belief system intuition behind neutrosophic relations.

First we will introduce two special operators on neutrosophic relations called split and combine to transform inconsistent neutrosophic relations into pseudo-consistent neutrosophic relations and transform pseudo-consistent neutrosophic relations into inconsistent neutrosophic relations.

Definition 60 (Split) *Let R be a neutrosophic relation on scheme Σ. Then,*
$\triangle(R) = \{\langle t, b, d\rangle | \langle t, b, d\rangle \in R \text{ and } b + d \leq 1\} \cup \{\langle t, b', d'\rangle | \langle t, b, d\rangle \in R \text{ and } b + d > 1 \text{ and } b' = b \text{ and } d' = 1 - b\} \cup \{\langle t, b', d'\rangle | \langle t, b, d\rangle \in R \text{ and } b + d > 1 \text{ and } b' = 1 - d \text{ and } d' = d\}.$

It is obvious that $\triangle(R)$ is pseudo-consistent if R is inconsistent.

Definition 61 (Combine) *Let R be a neutrosophic relation on scheme Σ. Then,*
$\nabla(R) = \{\langle t, b', d'\rangle | (\exists b)(\exists d)((\langle t, b', d\rangle \in R \text{ and } (\forall b_i, d_i)(\langle t, b_i, d_i\rangle \rightarrow b' \geq b_i) \text{ and } \langle t, b, d'\rangle \in R \text{ and } (\forall b_i)(\forall d_i)(\langle t, b_i, d_i\rangle \rightarrow d' \geq d_i))\}.$

It is obvious that $\nabla(R)$ is inconsistent if R is pseudo-consistent.

Note that strong generalization defined above only holds for consistent or pseudo-consistent neutrosophic relations. For any arbitrary paraconsistent intuitionistic fuzzy relations, we should first use split operation to transform them into non inconsistent neutrosophic relations and apply the set-theoretic and relation-theoretic operations on them and finally use combine operation to transform the result into arbitrary neutrosophic relation. For the simplification of notation, the following generalized algebra is defined under such assumption.

3.4 Generalized Algebra on Neutrosophic Relations

In this section, we present one strong generalization each for the fuzzy relation operators such as union, join, projection. To reflect generalization, a hat is placed over a fuzzy relation operator to obtain the corresponding neutrosophic relation operator. For example, $\bowtie$ denotes the natural join among fuzzy relations, and $\hat{\bowtie}$ denotes natural join on neutrosophic relations. These generalized operators maintain the belief system intuition behind neutrosophic relations.

Set-Theoretic Operators

We first generalize the two fundamental set-theoretic operators, union and complement.

Definition 62 *Let R and S be neutrosophic relations on scheme Σ. Then,*

(a) *the* union *of R and S, denoted $R \,\widehat{\cup}\, S$, is a neutrosophic relation on scheme Σ, given by*

$$(R \,\widehat{\cup}\, S)(t) = \langle \max\{R(t)^+, S(t)^+\}, \min\{R(t)^-, S(t)^-\}\rangle, \ for\ any\ t \in \tau(\Sigma);$$

(b) *the* complement *of R, denoted $\widehat{-}\, R$, is a neutrosophic relation on scheme Σ, given by*

$$(\widehat{-}\, R)(t) = \langle R(t)^-, R(t)^+\rangle, \ for\ any\ t \in \tau(\Sigma).$$

$\square$

An intuitive appreciation of the union operator can be obtained as follows: Given a tuple t, since we believed that it is present in the relation R with confidence $R(t)^+$ and that it is present in the relation S with confidence $S(t)^+$, we can now believe that the tuple t is present in the "either-R-or-S" relation with confidence which is equal to the larger of $R(t)^+$ and $S(t)^+$. Using the same logic, we can now believe in the absence of the tuple t from the "either-R-or-S" relation with confidence which is equal to the smaller (because t must be absent from both R and S for it to be absent from the union) of $R(t)^-$ and $S(t)^-$. The definition of *complement* and of all the other operators on neutrosophic relations defined later can (and should) be understood in the same way.

Proposition 3 *The operators $\widehat{\cup}$ and unary $\widehat{-}$ on neutrosophic relations are strong generalisations of the operators $\cup$ and unary $-$ on fuzzy relations.*

Proof Let R and S be consistent neutrosophic relations on scheme Σ. Then $\mathbf{reps}_\Sigma(R \,\widehat{\cup}\, S)$ is the set

$$\{Q \mid \bigwedge_{t_i \in \tau(\Sigma)} (\max\{R(t_i)^+, \ S(t_i)^+\} \le Q(t_i) \le 1 - \min\{R(t_i)^-, \ S(t_i)^-\})\}$$

This set is the same as the set

$$\{r \,\cup\, s \mid \bigwedge_{t_i \in \tau(\Sigma)} (R(t_i)^+ \le r(t_i) \le 1 - R(t_i)^-), \bigwedge_{t_i \in \tau(\Sigma)} (S(t_i)^+ \le s(t_i) \le 1 - S(t_i)^-)\}$$

which is $S(\cup)(\mathbf{reps}_\Sigma(R), \ \mathbf{reps}_\Sigma(S))$. Such a result for unary $\widehat{-}$ can also be shown similarly.

For sake of completeness, we define the following two related set-theoretic operators:

Definition 63 *Let R and S be neutrosophic relations on scheme Σ. Then,*

(a) *the* intersection *of R and S, denoted $R \,\widehat{\cap}\, S$, is a neutrosophic relation on scheme Σ, given by*

$$(R \,\widehat{\cap}\, S)(t) = \langle \min\{R(t)^+, S(t)^+\}, \max\{R(t)^-, S(t)^-\}\rangle, \ for\ any\ t \in \tau(\Sigma);$$

(b) *the* difference *of R and S, denoted $R \,\widehat{-}\, S$, is a neutrosophic relation on scheme Σ, given by*

$$(R \,\widehat{-}\, S)(t) = \langle \min\{R(t)^+, S(t)^-\}, \max\{R(t)^-, S(t)^+\}\rangle, \ for\ any\ t \in \tau(\Sigma);$$

$\square$

The following proposition relates the intersection and difference operators in terms of the more fundamental set-theoretic operators union and complement.

Proposition 4 *For any neutrosophic relations R and S on the same scheme, we have*

$$R \mathbin{\widehat{\cap}} S = \widehat{-}(\widehat{-}R \mathbin{\widehat{\cup}} \widehat{-}S), \text{ and}$$
$$R \mathbin{\widehat{-}} S = \widehat{-}(\widehat{-}R \mathbin{\widehat{\cup}} S).$$

Proof

$$
\begin{aligned}
\text{By definition, } \widehat{-}R(t) &= \langle R(t)^-, R(t)^+ \rangle \\
\widehat{-}S(t) &= \langle S(t)^-, S(t)^+ \rangle \\
\text{and } (\widehat{-}R \mathbin{\widehat{\cup}} \widehat{-}S)(t) &= \langle \max(R(t)^-, S(t)^-), \ \min(R(t)^+, S(t)^+) \rangle \\
\text{so, } (\widehat{-}(\widehat{-}R \mathbin{\widehat{\cup}} \widehat{-}S))(t) &= \langle \min(R(t)^+, S(t)^+), \max(R(t)^-, S(t)^-) \rangle \\
&= R \mathbin{\widehat{\cap}} S(t).
\end{aligned}
$$

The second part of the result can be shown similarly.

Relation-Theoretic Operators

We now define some relation-theoretic algebraic operators on neutrosophic relations.

Definition 64 *Let R and S be neutrosophic relations on schemes Σ and Δ, respectively. Then, the* natural join *(further for short called* join*) of R and S, denoted $R \mathbin{\widehat{\bowtie}} S$, is a neutrosophic relation on scheme $\Sigma \cup \Delta$, given by*

$$(R \mathbin{\widehat{\bowtie}} S)(t) = \langle \min\{R(\pi_\Sigma(t))^+, S(\pi_\Delta(t))^+\}, \max\{R(\pi_\Sigma(t))^-, S(\pi_\Delta(t))^-\} \rangle,$$

where π is the usual projection of a tuple. $\qquad\square$

It is instructive to observe that, similar to the intersection operator, the minimum of the belief factors and the maximum of the doubt factors are used in the definition of the join operation.

Proposition 5 $\widehat{\bowtie}$ *is a strong generalization of* $\bowtie$.

Proof Let R and S be consistent neutrosophic relations on schemes Σ and Δ, respectively. Then $\mathbf{reps}_{\Sigma \cup \Delta}(R \mathbin{\widehat{\bowtie}} S)$ is the set $\{Q \in \mathcal{F}(\Sigma \cup \Delta) \mid \bigwedge_{t_i \in \tau(\Sigma \cup \Delta)}(\min\{R_{\pi_\Sigma}(t_i)^+, S_{\pi_\Delta}(t_i)^+\} \leq Q(t_i) \leq 1 - \max\{R_{\pi_\Sigma}(t_i)^-, S_{\pi_\Delta}(t_i)^-\})\}$ and $S(\bowtie)(\mathbf{reps}_\Sigma(R), \mathbf{reps}_\Delta(S)) = \{r \bowtie s \mid r \in \mathbf{reps}_\Sigma(R), s \in \mathbf{reps}_\Delta(S)\}$

Let $Q \in \mathbf{reps}_{\Sigma \cup \Delta}(R \mathbin{\widehat{\bowtie}} S)$. Then $\pi_\Sigma(Q) \in \mathbf{reps}_\Sigma(R)$, where π_Σ is the usual projection over Σ of fuzzy relations. Similarly, $\pi_\Delta(Q) \in \mathbf{reps}_\Delta(S)$. Therefore, $Q \in S(\bowtie)(\mathbf{reps}_\Sigma(R), \mathbf{reps}_\Delta(S))$.

Let $Q \in S(\bowtie)(\mathbf{reps}_\Sigma(R), \mathbf{reps}_\Delta(S))$. Then $Q(t_i) \geq \min\{R_{\pi_\Sigma}(t_i)^+, S_{\pi_\Delta}(t_i)^+\}$ and $Q(t_i) \leq \min\{1 - R\pi_\Sigma(t_i)^-, 1 - S_{\pi_\Delta}(t_i)^-\} = 1 - \max\{R_{\pi_\Sigma}(t_i)^-, S_{\pi_\Delta}(t_i)^-\}$, for any $t_i \in \tau(\Sigma \cup \Delta)$, because R and S are consistent.

Therefore, $Q \in \mathbf{reps}_{\Sigma \cup \Delta}(R \mathbin{\widehat{\bowtie}} S)$.

We now present the projection operator.

Definition 65 *Let R be a neutrosophic relation on scheme Σ, and $\Delta \subseteq \Sigma$. Then, the* projection *of R onto Δ, denoted $\widehat{\pi}_\Delta(R)$, is a neutrosophic relation on scheme Δ, given by*

$$(\widehat{\pi}_\Delta(R))(t) = \langle \max\{R(u)^+ | u \in t^\Sigma\}, \min\{R(u)^- | u \in t^\Sigma\}\rangle.$$

$\square$

The belief factor of a tuple in the projection is the maximum of the belief factors of all of the tuple's extensions onto the scheme of the input neutrosophic relation. Moreover, the doubt factor of a tuple in the projection is the minimum of the doubt factors of all of the tuple's extensions onto the scheme of the input neutrosophic relation.

We present the selection operator next.

Definition 66 *Let R be a neutrosophic relation on scheme Σ, and let F be any logic formula involving attribute names in Σ, constant symbols (denoting values in the attribute domains), equality symbol $=$, negation symbol $\neg$, and connectives $\vee$ and $\wedge$. Then, the* selection *of R by F, denoted $\widehat{\sigma}_F(R)$, is a neutrosophic relation on scheme Σ, given by*

$$(\widehat{\sigma}_F(R))(t) = \langle \alpha, \beta\rangle, \text{ where}$$

$$\alpha = \begin{cases} R(t)^+ & \text{if } t \in \sigma_F(\tau(\Sigma)) \\ 0 & \text{otherwise} \end{cases} \qquad \text{and} \qquad \beta = \begin{cases} R(t)^- & \text{if } t \in \sigma_F(\tau(\Sigma)) \\ 1 & \text{otherwise} \end{cases}$$

where σ_F is the usual selection of tuples satisfying F from ordinary relations. $\square$

If a tuple satisfies the selection criterion, it's belief and doubt factors are the same in the selection as in the input neutrosophic relation. In the case where the tuple does not satisfy the selection criterion, its belief factor is set to 0 and the doubt factor is set to 1 in the selection.

Proposition 6 *The operators $\widehat{\pi}$ and $\widehat{\sigma}$ are strong generalisations of π and σ, respectively.*

Proof Similar to that of Proposition 5.

Example 17 *Relation schemes are sets of attribute names, but in this example we treat them as ordered sequences of attribute names (which can be obtained through permutation of attribute names), so tuples can be viewed as the usual lists of values. Let $\{a, b, c\}$ be a common domain for all attribute names, and let R and S be the following neutrosophic relations on schemes $\langle X, Y\rangle$ and $\langle Y, Z\rangle$ respectively.*

t	$R(t)$
(a, a)	$\langle 0, 1\rangle$
(a, b)	$\langle 0, 1\rangle$
(a, c)	$\langle 0, 1\rangle$
(b, b)	$\langle 1, 0\rangle$
(b, c)	$\langle 1, 0\rangle$
(c, b)	$\langle 1, 1\rangle$

t	$S(t)$
(a, c)	$\langle 1, 0\rangle$
(b, a)	$\langle 1, 1\rangle$
(c, b)	$\langle 0, 1\rangle$

For other tuples which are not in the neutrosophic relations $R(t)$ and $S(t)$, their $\langle \alpha, \beta \rangle = \langle 0, 0 \rangle$ which means no any information available. Because R and S are inconsistent, we first use split operation to transform them into pseudo-consistent and apply the relation-theoretic operations on them and transform the result back to arbitrary neutrosophic set using combine operation. Then, $T_1 = \nabla(\triangle(R) \mathbin{\widehat{\bowtie}} \triangle(S))$ is a neutrosophic relation on scheme $\langle X, Y, Z \rangle$ and $T_2 = \nabla(\widehat{\pi}_{\langle X, Z \rangle}(\triangle(T_1)))$ and $T_3 = \widehat{\sigma}_{X \neg = Z}(T_2)$ are neutrosophic relations on scheme $\langle X, Z \rangle$. T_1, T_2 and T_3 are shown below:

t	$T_1(t)$
(a,a,a)	$\langle 0,1 \rangle$
(a,a,b)	$\langle 0,1 \rangle$
(a,a,c)	$\langle 0,1 \rangle$
(a,b,a)	$\langle 0,1 \rangle$
(a,b,b)	$\langle 0,1 \rangle$
(a,b,c)	$\langle 0,1 \rangle$
(a,c,a)	$\langle 0,1 \rangle$
(a,c,b)	$\langle 0,1 \rangle$
(a,c,c)	$\langle 0,1 \rangle$
(b,b,a)	$\langle 1,1 \rangle$
(b,c,b)	$\langle 0,1 \rangle$
(c,b,a)	$\langle 1,1 \rangle$
(c,b,b)	$\langle 0,1 \rangle$
(c,b,c)	$\langle 0,1 \rangle$
(c,c,b)	$\langle 0,1 \rangle$

t	$T_2(t)$
(a,a)	$\langle 0,1 \rangle$
(a,b)	$\langle 0,1 \rangle$
(a,c)	$\langle 0,1 \rangle$
(b,a)	$\langle 1,0 \rangle$
(c,a)	$\langle 1,0 \rangle$

t	$T_3(t)$
(a,a)	$\langle 0,1 \rangle$
(a,b)	$\langle 0,1 \rangle$
(a,c)	$\langle 0,1 \rangle$
(b,a)	$\langle 1,0 \rangle$
(b,b)	$\langle 0,1 \rangle$
(c,a)	$\langle 1,0 \rangle$
(c,c)	$\langle 0,1 \rangle$

$\square$

3.5 An Application

Consider the target recognition example presented in [Sub94]. Here, an autonomous vehicle needs to identify objects in a hostile environment such as a military battlefield. The autonomous vehicle is equipped with a number of sensors which are used to collect data, such as speed and size of the objects (tanks) in the battlefield. Associated with each sensor, we have a set of rules that describe the type of the object based on the properties detected by the sensor.

Let us assume that the autonomous vehicle is equipped with three sensors resulting in data collected about radar readings, of the tanks, their gun characteristics and their speeds. What follows is a set of rules that associate the type of object with various observations.

Radar Readings:

- Reading r_1 indicates that the object is a T-72 tank with belief factor 0.80 and doubt factor 0.15.

- Reading r_2 indicates that the object is a T-60 tank with belief factor 0.70 and doubt factor 0.20.

- Reading r_3 indicates that the object is not a T-72 tank with belief factor 0.95 and doubt factor 0.05.

- Reading r_4 indicates that the object is a T-80 tank with belief factor 0.85 and doubt factor 0.10.

Gun Characteristics:

- Characteristic c_1 indicates that the object is a T-60 tank with belief factor 0.80 and doubt factor 0.20.

- Characteristic c_2 indicates that the object is not a T-80 tank with belief factor 0.90 and doubt factor 0.05.

- Characteristic c_3 indicates that the object is a T-72 tank with belief factor 0.85 and doubt factor 0.10.

Speed Characteristics:

- Low speed indicates that the object is a T-60 tank with belief factor 0.80 and doubt factor 0.15.

- High speed indicates that the object is not a T-72 tank with belief factor 0.85 and doubt factor 0.15.

- High speed indicates that the object is not a T-80 tank with belief factor 0.95 and doubt factor 0.05.

- Medium speed indicates that the object is not a T-80 tank with belief factor 0.80 and doubt factor 0.10.

These rules can be captured in the following three neutrosophic relations:

Radar Rules

Reading	Object	Confidence Factors
r_1	T-72	$\langle 0.80, 0.15 \rangle$
r_2	T-60	$\langle 0.70, 0.20 \rangle$
r_3	T-72	$\langle 0.05, 0.95 \rangle$
r_4	T-80	$\langle 0.85, 0.10 \rangle$

Gun Rules

Reading	Object	Confidence Factors
c_1	T-60	$\langle 0.80, 0.20 \rangle$
c_2	T-80	$\langle 0.05, 0.90 \rangle$
c_3	T-72	$\langle 0.85, 0.10 \rangle$

Speed Rules

Reading	Object	Confidence Factors
low	T-60	$\langle 0.80, 0.15 \rangle$
high	T-72	$\langle 0.15, 0.85 \rangle$
high	T-80	$\langle 0.05, 0.95 \rangle$
medium	T-80	$\langle 0.10, 0.80 \rangle$

The autonomous vehicle uses the sensors to make observations about the different objects and then uses the rules to determine the type of each object in the battlefield. It is quite possible that two different sensors may identify the same object as of different types, thereby introducing inconsistencies.

Let us now consider three objects o_1, o_2 and o_3 which need to be identified by the autonomous vehicle. Let us assume the following observations made by the three sensors about the three objects. Once again, we assume certainty factors (maybe derived from the accuracy of the sensors) are associated with each observation.

Radar Data

Object-id	Reading	Confidence Factors
o_1	r_3	$\langle 1.00, 0.00 \rangle$
o_2	r_1	$\langle 1.00, 0.00 \rangle$
o_3	r_4	$\langle 1.00, 0.00 \rangle$

Gun Data

Object-id	Reading	Confidence Factors
o_1	c_3	$\langle 0.80, 0.10 \rangle$
o_2	c_1	$\langle 0.90, 0.10 \rangle$
o_3	c_2	$\langle 0.90, 0.10 \rangle$

Speed Data

Object-id	Reading	Confidence Factors
o_1	high	$\langle 0.90, 0.10 \rangle$
o_2	low	$\langle 0.95, 0.05 \rangle$
o_3	medium	$\langle 0.80, 0.20 \rangle$

Given these observations and the rules, we can use the following algebraic expression to identify the three objects:

$$\widehat{\pi}_{\text{Object-id,Object}}(\text{Radar Data} \mathbin{\widehat{\bowtie}} \text{Radar Rules}) \mathbin{\widehat{\cap}}$$
$$\widehat{\pi}_{\text{Object-id,Object}}(\text{Gun Data} \mathbin{\widehat{\bowtie}} \text{Gun Rules}) \mathbin{\widehat{\cap}}$$
$$\widehat{\pi}_{\text{Object-id,Object}}(\text{Speed Data} \mathbin{\widehat{\bowtie}} \text{Speed Rules})$$

Table 3.1: EVAL

I_1	q_1	$\langle 0.9, 0.2 \rangle$
I_1	q_2	$\langle 1.0, 0.0 \rangle$
I_1	q_3	$\langle 0.1, 0.8 \rangle$
I_2	q_1	$\langle 1.0, 1.0 \rangle$
I_2	q_3	$\langle 0.8, 0.3 \rangle$

The intuition behind the intersection is that we would like to capture the common (intersecting) information among the three sensor data. Evaluating this expression, we get the following neutrosophic relation:

Object-id	Object	Confidence Factors
o_1	T-72	$\langle 0.05, 0.0 \rangle$
o_2	T-80	$\langle 0.0, 0.05 \rangle$
o_3	T-80	$\langle 0.05, 0.0 \rangle$

It is clear from the result that by the given information, we could not infer any useful information that is we could not decide the status of objects o_1, o_2 and o_3.

3.6   An Infinite-Valued Tuple Relational Calculus

As an example, suppose in the e-shopping environment, there are two items I_1 and I_2, which are evaluated by customers for some categories of quality q_1, q_2 and q_3. Let the evaluation results be captured by the following neutrosophic relation EVAL on scheme $\{I, Q\}$:

The above neutrosophic relation contains the information that the confidence of item I_1 was evaluated "good" for category q_1 is 0.9 and the doubt is 0.2. The confidence of item I_1 was evaluated "good" for category q_2 is 1.0 and the doubt is 0.0. The confidence of item I_1 was evaluated "poor" for category q_3 is 0.8 and the doubt is 0.1. Also, the confidence of item I_2 was evaluated "good" for category q_1 is 1.0 and the doubt is 1.0 (similarly, the confidence of item I_2 was evaluated "poor" for category q_1 is 1.0 and the doubt is 1.0). The confidence of I_2 was evaluated "good" for category q_3 is 0.8 and the doubt is 0.3. Note that the evaluation results of item I_2 for category q_2 is unknown. The above information contains fuzziness, incompleteness and inconsistency. Such information may be due to various reasons, such as evaluation not conducted, or evaluation results not yet available, the evaluation is not reliable, and different evaluation results for the same category producing different results, etc.

We define a infinite-valued membership function of a neutrosophic relation, which maps tuples to the pair of values $\langle \alpha, \beta$, with $0 \leq \alpha + \beta \leq 2$. We use the symbol $\mathbf{I}$ to denote the set of these values, i.e. $\mathbf{I} = \{\langle \alpha, \beta \rangle\}$. Now, for a neutrosophic relation $R = \langle t, R(t)^+, R(t)^- \rangle$ on scheme Σ, its membership function is a *infinite-valued predicate* $\Phi_R : \tau(\Sigma) \to \mathbf{I}$, given by

$$\Phi_R(t) = \langle R(t)^+, R(t)^- \rangle.$$

In [Bag00], it proposed a 4-valued characteristic function of neutrosophic relation, which maps tuples to one of the following values: $\top$ (for contradiction), t (for true), f(for false) and $\perp$ (for unknown). It can be easily verified that when $R(t)^+ = R(t)^- = 1$, it corresponds to $\top$; when $R(t)^+ = 1, R(t)^- = 0$, it corresponds to t; when $R(t)^+ = 0, R(t)^- = 1$, it corresponds to f; and when $R(t)^+ = R(t)^- = 0$, it corresponds to $\perp$.

The tuple relational calculus provides a very natural, set-theoretic, declarative notation for querying ordinary relational database management systems. A tuple calculus expression has the form:

$$\{t \text{ of } \Sigma | P(t)\},$$

where t is a tuple variable, Σ a scheme, and P is some 2-valued predicate on tuples in $\tau(\Sigma)$. The expression denotes the set of all tuple values T (from $\tau(\Sigma)$) of the variable t for which the predicate $P(T)$ is true.

We retain the above simple syntax in the generalised tuple calculus expression for neutrosophic databases. However, the predicate P is now interpreted as a *infinite-valued* predicate on tuples. Moreover, the entire expression now denotes a neutrosophic relation (of which P is the membership function).

In this section we define the syntax and semantics of legal infinite-valued predicate expressions. They are defined in relation to a given set of binary comparators on domains associated with the attribute names appearing in schemes. Most intuitive binary comparators, like $<$ and $\leq$, produce 2-valued results, but in principle infinite-valued comparators are possible. The basic building blocks of formulas are *atoms*, of which there are four kinds:

1. For any tuple variable t and relation R on the same scheme, $t \,\tilde{\in}\, R$ is an atom. For any tuple value T for the variable t, the atom $t \,\tilde{\in}\, R$ denotes the value $\Phi_R(T)$.

2. For any tuple variable t_1 and t_2, attribute names A and B in the schemes of t_1 and t_2 respectively, and binary comparator Θ such that A and B are Θ-comparable, $t_1.A \,\Theta\, t_2.B$ is an atom. For any tuple values T_1 and T_2 for the variables t_1 and t_2 respectively, the atom $t_1.A \,\Theta\, t_2.B$ denotes the value $T_1(A) \,\Theta\, T_2(B)$.

3. For any tuple variable t, constant c, and attribute names A and B such that A is in the scheme of t, $c \in dom(B)$, and A and B are Θ-comparable, $t.A \,\Theta\, c$ is an atom. For any tuple value T for the variable t, the atom $t.A \,\Theta\, c$ denotes the value $T(A) \,\Theta\, c$.

4. For any constant c, tuple variable t, and attribute names A and B such that $c \in dom(A)$, B is in the scheme of t, and A and B are Θ-comparable, $c \,\Theta\, t.B$ is an atom. For any tuple value T for the variable t, the atom $c \,\Theta\, t.B$ denotes the value $c \,\Theta\, T(B)$.

We use infinite-valued connectives $\tilde{\neg}$ (not), $\tilde{\wedge}$ (and), $\tilde{\vee}$ (or), $\tilde{\exists}$ (there exists) and $\tilde{\forall}$ (for all) to recursively build *formulas* from atoms. Any atom is a formula, where the formula denotes the same value as the atom.

If f and g are formulas, and f^+ is truth-degree of the f, f^- is falsity-degree of f, then $\tilde{\neg} \, f$, $f \, \tilde{\wedge} \, g$ and $f \, \tilde{\vee} \, g$ are also formulas. The values of such formulas are given as the following:

$$\tilde{\neg} \, f = \langle f^-, f^+ \rangle \tag{3.1}$$

$$f \, \tilde{\wedge} \, g = \langle \min(f^+, g^+), \max(f^-, g^-) \rangle \tag{3.2}$$

$$f \, \tilde{\vee} \, g = \langle \max(f^+, g^+), \min(f^-, g^-) \rangle \tag{3.3}$$

An intuitive appreciation of the disjunctive connective can be obtained as follows: Given a tuple t, since we believed that it is present in the relation R with confidence $R(t)^+$ and that it is present in the relation S with confidence $S(t)^+$, we can now believe that the tuple t is present in the "either-R-or-S" relation with confidence which is equal to the larger of $R(t)^+$ and $S(t)^+$. Using the same logic, we can now believe in the absence of the tuple t from the "either-R-or-S" relation with confidence which is equal to the smaller (because t must be absent from both R and S for it to be absent from the disjunction) of $R(t)^-$ and $S(t)^-$. The definition of *negation* and *conjunction* can be understood in the same way.

The duality of $\tilde{\wedge}$ and $\tilde{\vee}$ is evident from the above formulas. It is interesting to note the algebraic laws shown in Table 3.2 that are exhibited by these connectives.

Table 3.2: Albegraic Properties of Infinite-Valued Propositional Connectives

(commutative laws)	$f \, \tilde{\vee} \, g = g \, \tilde{\vee} \, f$
	$f \, \tilde{\wedge} \, g = g \, \tilde{\wedge} \, f$
(associative laws)	$(f \, \tilde{\vee} \, g) \, \tilde{\vee} \, h = f \, \tilde{\vee} \, (g \, \tilde{\vee} \, h)$
	$(f \, \tilde{\wedge} \, g) \, \tilde{\wedge} \, h = f \, \tilde{\wedge} \, (g \, \tilde{\wedge} \, h)$
(distributive laws)	$f \, \tilde{\vee} \, (g \, \tilde{\wedge} \, h) = (f \, \tilde{\vee} \, g) \, \tilde{\wedge} \, (f \, \tilde{\vee} \, h)$
	$f \, \tilde{\wedge} \, (g \, \tilde{\vee} \, h) = (f \, \tilde{\wedge} \, g) \, \tilde{\vee} \, (f \, \tilde{\wedge} \, h)$
(idempotent laws)	$f \, \tilde{\vee} \, f = f$
	$f \, \tilde{\wedge} \, f = f$
(identity laws)	$f \, \tilde{\vee} \, \mathbf{f} = f$
	$f \, \tilde{\wedge} \, \mathbf{t} = f$
(double complementation)	$\tilde{\neg} \, (\tilde{\neg} \, f) = f$
(De Morgan laws)	$\tilde{\neg} \, (f \, \tilde{\vee} \, g) = \tilde{\neg} \, f \, \tilde{\wedge} \, \tilde{\neg} \, g$
	$\tilde{\neg} \, (f \, \tilde{\wedge} \, g) = \tilde{\neg} \, f \, \tilde{\vee} \, \tilde{\neg} \, g$

If t is a tuple variable, Σ a scheme, and P an infinite-valued predicate on tuples in $\tau(\Sigma)$, then $\tilde{\exists} t$ of $\Sigma | P(t)$ and $\tilde{\forall} t$ of $\Sigma | P(t)$ are formulas. If P is the membership function of the neutrosophic rela-

tion R, then the values denoted by these formulas are given by

$$\tilde{\exists} t \text{ of } \Sigma | P(t) = \langle t_{\tilde{\exists}}, f_{\tilde{\exists}} \rangle, \tag{3.4}$$

where $t_{\tilde{\exists}} = \max\{R(t)^+\}$, for all $t \in \tau(\Sigma)$, $f_{\tilde{\exists}} = \min\{R(t)^-\}$, for all $t \in \tau(\Sigma)$.

$$\tilde{\forall} t \text{ of } \Sigma | P(t) = \langle t_{\tilde{\forall}}, f_{\tilde{\forall}} \rangle, \tag{3.5}$$

where $t_{\tilde{\forall}} = \min\{R(t)^+\}$, for all $t \in \tau(\Sigma)$, $f_{\tilde{\exists}} = \max\{R(t)^+\}$, for all $t \in \tau(\Sigma)$.

The extended De Morgan laws can be verified to continue to hold for our generalized infinite-valued semantics for quantifiers, i.e. the following pairs of formulas are equivalent:

$$\tilde{\exists} t \text{ of } \Sigma | P(t) \equiv \tilde{\neg} \left(\tilde{\forall} t \text{ of } \Sigma | \tilde{\neg} P(t) \right)$$

$$\tilde{\forall} t \text{ of } \Sigma | P(t) \equiv \tilde{\neg} \left(\tilde{\exists} t \text{ of } \Sigma | \tilde{\neg} P(t) \right)$$

$$\tilde{\exists} t \text{ of } \Sigma | (P(t) \tilde{\wedge} Q(t)) \equiv \tilde{\neg} \left(\tilde{\forall} t \text{ of } \Sigma | \tilde{\neg} P(t) \tilde{\vee} \tilde{\neg} Q(t) \right)$$

$$\tilde{\exists} t \text{ of } \Sigma | (P(t) \tilde{\vee} Q(t)) \equiv \tilde{\neg} \left(\tilde{\forall} t \text{ of } \Sigma | \tilde{\neg} P(t) \tilde{\wedge} \tilde{\neg} Q(t) \right)$$

$$\tilde{\forall} t \text{ of } \Sigma | (P(t) \tilde{\wedge} Q(t)) \equiv \tilde{\neg} \left(\tilde{\exists} t \text{ of } \Sigma | \tilde{\neg} P(t) \tilde{\vee} \tilde{\neg} Q(t) \right)$$

$$\tilde{\forall} t \text{ of } \Sigma | (P(t) \tilde{\vee} Q(t)) \equiv \tilde{\neg} \left(\tilde{\exists} t \text{ of } \Sigma | \tilde{\neg} P(t) \tilde{\wedge} \tilde{\neg} Q(t) \right)$$

It is worth mentioning that in ordinary 2-valued relational calculus caution needs to be exercised in mixing negation and quantifiers in a *safe* manner as the resulting expressions have the potential of denoting infinite relations, even if all components denote finite relations. Fortunately, as neutrosophic databases are by nature capable of handling infinite relations, safety of expressions is not an issue in infinite-valued calculus.

3.6.1 An Example

Let us now consider an example illustrating some infinite-valued computations. We use the neutrosophic relation EVAL on scheme $\{I, Q\}$ of the item-category evaluation as example.

Consider the query:

What items showed contradictory evaluation results for some category?

In ordinary relational databases, it is impossible to store contradictory information, let alone entertaining queries about contradiction.

Let $\Delta = \{I\}$, and $\Sigma = \{I, Q\}$ be schemes. A tuple calculus expression for this query is:

$$\{d \text{ of } \Delta | (\tilde{\exists} t \text{ of } \Sigma | t.I = d.I \tilde{\wedge} t \tilde{\in} \text{EVAL} \ \tilde{\wedge} \ \tilde{\neg} \ t \tilde{\in} \text{EVAL}) \}$$

In the ordinary 2-valued logic the above query will produce an empty answer due to the condition for the tuple t to simultaneously be in EVAL as well as not be in EVAL. In infinite-valued logic, however, the query denotes that neutrosophic relation on scheme Δ whose membership function is denoted by the infinite-valued predicate expression

$$\tilde{\exists} t \text{ of } \Sigma | t.I = d.I \ \tilde{\wedge} \ t \ \tilde{\in} \ \text{EVAL} \ \tilde{\wedge} \ \tilde{\neg} \ t \ \tilde{\in} \ \text{EVAL} \tag{3.6}$$

That function can be computed by determining the value of the above expression for all possible values of its free variable d, namely I_1 and I_2.

For the value $d = I_1$, the expression (3.6) can be seen to reduce to the value $\langle 0.2, 0.8 \rangle$. For the value $d = I_2$, the expression (3.6) can be seen to reduce to the value $\langle 1.0, 0.8 \rangle$. The result is the neutrosophic relation:

I_1	$\langle 0.2, 0.8 \rangle$
I_2	$\langle 1.0, 0.8 \rangle$

The result states that I_1 showed contradictory evaluation result for some category with confidence is 0.2 and doubt is 0.8, so it is safe to conclude that I_1 did not show contradictory evaluation resulti, but I_2 showed contradictory evaluation result for some category with confidence 1.0 and doubt is 0.8, the explanation is that I_2 did show contradictory result for some category and did not show contradictory for other category at the same times.

3.7   A Generalized SQL Query Construct for Neutrosophic Relations

The most popular construct for information retrieval from most commercial systems is the SQL's SELECT statement. While the statement has many options and extensions to its basic form, here we just present an infinite-valued generalization to the basic form, as generalizing the options then just becomes a trivial matter of detail. The basic form of the statement contains three clauses select, from and where, and has the following format:

$$\text{select } A_1, A_2, \ldots A_m \text{ from } R_1, R_2, \ldots R_n \text{ where } C$$

where

1. $A_1, A_2, \ldots A_m$ is a list of attribute names whose values are to be retrieved by the query,

2. $R_1, R_2, \ldots R_n$ is a list of relation names required to process the query, and

3. C is a boolean expression that identifies the tuples to be retrieved by the query.

Without loss of generality, we assume that each attribute name occurs in exactly one relation, because if some attribute A_i occurs in more than one relation, we require, instead of simply the attribute Ai, a pair of the form $R_j.A_i$ qualifying that attribute. The result of the SELECT statement is a relation with attributes $A_1, A_2, \ldots A_m$ chosen from the attributes of $R_1 \times R_2 \times \cdots \times R_n$ for tuples that satisfy the boolean condition C, i.e.

$$\pi_{A_1,A_2,\ldots A_m}(\sigma_C(R_1 \times R_2 \times \cdots \times R_n)),$$

where π, σ, and $\times$ are the projection, selection and product operations, respectively, on ordinary relations. We retain the above syntax in the generalized SELECT statement for the neutrosophic relations. However, the relation names $R_1, R_2, \ldots R_n$ now represent some neutrosophic relations and C is some infinite-valued condition. The result of the generalized SELECT statement is then the value of the algebraic expression:

$$\pi_{A_1,A_2,\ldots,A_m}(\widehat{\sigma}_C(R_1\widehat{\times}R_2\widehat{\times}\cdots\widehat{\times}R_n)),$$

where $\widehat{\pi}, \widehat{\sigma}$, and $\widehat{\times}$ are, respectively, the projection, selection and product operations on neutrosophic relations constructed in the next section. Furthermore, the result of the generalized SELECT statement is also a neutrosophic relation.

3.7.1 Infinite-Valued Conditions

In the generalized SELECT statement, we let the condition occurring in the where clause be infinite-valued. The infinite values, except $\langle 1, 0 \rangle$ and $\langle 0, 1 \rangle$, arise essentially due to any nested subqueries. For any arithmetic expressions E_1 and E_2, comparisons such as $E_1 \leq E_2$ are simply 2-valued conditions ($\langle 1, 0 \rangle$ or $\langle 0, 1 \rangle$). Let ξ be a subquery of the form

$$(\text{select} \ldots \text{from} \ldots \text{where} \ldots)$$

occurring in the where clause of a SELECT statement. And let R be the neutrosophic relation on scheme Σ that the subquery ξ evaluates to. Then, conditions involving the subquery ξ evaluate as follows.

1. The condition

$$\text{exists } \xi$$

   evaluates to $\langle \alpha, \beta \rangle$,
$$\alpha = \max\{a\}, a = R(t)^+, \text{ for all } t \in \tau(\Sigma),$$

   $\beta = \min\{b\}, b = R(t)^-, \text{ if } R(t)^+ + R(t)^- \leq 1, b = 1 - R(t)^+, \text{ if } R(t)^+ + R(t)^- > 1, \text{ for all } t \in \tau(\Sigma).$

2. For any tuple $t \in \tau(\Sigma)$, the condition

$$t \text{ in } \xi$$

evaluates to $\phi_R(t)$.

3. If Σ contains exactly one attribute, then for any (scalar value) $t \in \tau(\Sigma)$, the condition

$$t > \text{any } \xi$$

evaluates to $\langle \alpha, \beta \rangle$,
$\alpha = \max\{a\}, a = R(k)^+,$ if $t > k,$ for some $k \in R, (\beta = \min\{b\}, b = R(k)^-,$ if $R(k)^+ + R(k)^- \leq 1,$
$b = 1 - R(k)^+,$ if $R(k)^+ + R(k)^- > 1),$ if $t > k,$ for some $k \in R;$
$\alpha = 0, \beta = 1,$ otherwise.

An infinite-valued semantics for other operators, such as $\geq$any, =any, can be defined similarly. Note that conditions involving such operators never evaluate to the value $\alpha, \beta,$ such that $\alpha + \beta > 1.$

4. If Σ contains exactly one attribute, then for any (scalar value) $t \in \tau(\Sigma)$, the condition

$$t > \text{all } \xi$$

evaluates to $\langle \alpha, \beta \rangle$,
$(\alpha = \min\{a\}, a = R(k)^-,$ if $R(k)^+ + R(k)^- \leq 1, a = 1 - R(k)^+,$ if $R(k)^+ + R(k)^- > 1),$ if $t \leq k,$
for some $k \in R, \beta = \max\{b\}, b = R(k)^+,$ if $t \leq k,$ for some $k \in R;$
$\alpha = 1, \beta = 0,$ otherwise.

An infinite-valued semantics for other operators, such as $\geq$all, =all, can be defined similarly. Note that conditions involving such operators never evaluate to the value $\langle \alpha, \beta \rangle,$ such that $\alpha + \beta > 1.$

We complete our infinite-valued semantics for conditions by defining the **not, and** and **or** operators on them. Let C and D be any conditions, and value of $C = \langle t_c, f_c \rangle$ and value of $D = \langle t_d, f_d \rangle$. Then, the value of the condition **not** C is given by

$$\textbf{not } C = \langle f_c, t_c \rangle$$

while the value of the condition C **and** D is given by

$$C \textbf{ and } D = \langle \min t_c, t_d, \max f_c, f_d \rangle$$

and that of the condition C **or** D is given by

$$C \textbf{ or } D = \langle \max t_c, t_d, \min f_c, f_d \rangle$$

The duality of **and** and **or** is evident from their formulas. It is interesting to note the following algebraic laws exhibited by the above infinite-valued operators:

1. Double Complementation Law:
$$\textbf{not } (\textbf{ not } C) = C$$

2. Identity and Idempotence Laws:
$$C \textbf{ and } \langle 1, 0 \rangle = C \textbf{ and } C = C$$
$$C \textbf{ or } \langle 0, 1 \rangle = C \textbf{ or } C = C$$

3. Commutativity Laws:
$$C \textbf{ and } D = D \textbf{ and } C$$
$$C \textbf{ or } D = D \textbf{ or } C$$

4. Associativity Laws:
$$C \textbf{ and } (D \textbf{ and } E) = (C \textbf{ and } D) \textbf{ and } E$$
$$C \textbf{ or } (D \textbf{ or } E) = (C \textbf{ or } D) \textbf{ or } E$$

5. Distributivity Laws:
$$C \textbf{ and } (D \textbf{ or } E) = (C \textbf{ and } D) \textbf{ or } (C \textbf{and } E)$$
$$C \textbf{ or } (D \textbf{ and } E) = (C \textbf{ or } D) \textbf{ and } (C \textbf{or } E)$$

6. De Morgan Laws:
$$\textbf{not } (C \textbf{ and } D) = (\textbf{ not } C) \textbf{ or } (\textbf{ not } D)$$
$$\textbf{not } (C \textbf{ or } D) = (\textbf{ not } C) \textbf{ and } (\textbf{ not } D)$$

We are now ready to define the selection operator on neutrosophic relations.

Let R be a neutrosophic relation on scheme Σ, and C be an infinite-valued condition on tuples of Σ denoted $\langle t_C(t), f_C(t) \rangle$. Then, the selection of R by C, denoted $\widehat{\sigma}_C(R)$, is a neutrosophic relation on scheme Σ, given by

$$(\widehat{\sigma}_C(R))(t) = \langle \min R(t)^+, t_C(t), \max R(t)^-, f_C(t) \rangle.$$

The above definition is similar to that of the **and** operator given earlier.

Since performing a simple union is impossible within a SELECT statement, SQL provides a union operator among subqueries to achieve this. We end this section with an infinite-valued semantics of union.

Let ξ_1 and ξ_2 be subqueries that evaluate, respectively, to neutrosophic relations R_1 and R_2 on scheme Σ. Then, the subquery

$$\xi_1 \textbf{ union } \xi_2$$

evaluates to the neutrosophic relation R on scheme Σ given by

$$R(t) = \langle \max R1(t)^+, R2(t)^+, \min R1(t)^-, R2(t)^- \rangle$$

(I, Q)	$C = $ **not** $((I, Q)$ **in** $EVAL)$
(I_1, q_1)	$\langle 0.2, 0.9 \rangle$
(I_1, q_2)	$\langle 0.0, 1.0 \rangle$
(I_1, q_3)	$\langle 0.8, 0.1 \rangle$
(I_2, q_1)	$\langle 1.0, 1.0 \rangle$
(I_2, q_2)	$\langle 0.0, 0.0 \rangle$
(I_2, q_3)	$\langle 0.3, 0.8 \rangle$

An intuitive appreciation of the union operator can be obtained as follows: Given a tuple t, since we believed that it is present in the relation R_1 with confidence $R_1(t)^+$ and that it is present in the relation R_2 with confidence $R_2(t)^+$, we can now believe that the tuple t is present in the "either-R_1-or-R_2" relation with confidence which is equal to the larger of $R_1(t)^+$ and $R_2(t)^+$. Using the same logic, we can now believe in the absence of the tuple t from the "either-R_1-or-R_2" relation with confidence which is equal to the smaller (because t must be absent from both R_1 and R_2 for it to be absent from the union) of $R_1(t)^-$ and $R_2(t)^-$.

3.7.2   An Example

Let us now consider an example illustrating some infinite-valued computations. We use the neutrosophic relation EVAL on scheme $\{I, Q\}$ of the item-category evaluation as example.

Consider the query:

> What items showed contradictory evaluation of some category of quality?

A SELECT statement for this query is:

$$\text{select } I$$
$$\text{from EVAL where } \textbf{not} \ ((I, Q) \textbf{ in } \text{EVAL})$$

One possible evaluation method for the above query in ordinary 2-valued SQL is to produce the I attribute of those rows of EVAL that satisfy the **where** condition. Since the **where** condition in the above case is exactly that row not be in EVAL, in 2-valued logic the above query will produce an empty answer.

In infinite-valued logic, however, the where condition needs to be evaluated, to one of infinite possible values, for every possible row with attributes $\Sigma = (I, Q)$. The result is then combined with EVAL according to the semantics of p, on which $\widehat{\sigma}$ is performed to produce the resulting neutrosophic relation.

Therefore, for each of the 6 rows in $\tau(\Sigma)$, we first evaluate the where condition C:

Now, $\widehat{\sigma}(EVAL)$ according to the definition of $\widehat{\sigma}$ evaluates to the neutrosophic relation:

Finally, $\widehat{\pi}$ of the above is the neutrosophic relation:

I_1	q_1	$\langle 0.2, 0.9 \rangle$
I_1	q_2	$\langle 0.0, 0.1 \rangle$
I_1	q_3	$\langle 0.1, 0.8 \rangle$
I_2	q_1	$\langle 1.0, 1.0 \rangle$
I_2	q_3	$\langle 0.3, 0.8 \rangle$

I_1	$\langle 0.1, 0.8 \rangle$
I_2	$\langle 1.0, 0.0 \rangle$

3.8 Conclusions

We have presented a generalization of fuzzy relations, intuitionistic fuzzy relations (interval-valued fuzzy relations) and paraconsistent relations, called neutrosophic relations, in which we allow the representation of confidence (belief and doubt) factors with each tuple. The algebra on fuzzy relations is appropriately generalized to manipulate neutrosophic relations.

Various possibilities exist for further study in this area. Recently, there has been some work in extending logic programs to involve quantitative paraconsistency. Paraconsistent logic programs were introduced in [BS89] and probabilistic logic programs in [NS92]. Paraconsistent logic programs allow negative atoms to appear in the head of clauses (thereby resulting in the possibility of dealing with inconsistency), and probabilistic logic programs associate confidence measures with literals and with entire clauses. The semantics of these extensions of logic programs have already been presented, but implementation strategies to answer queries have not been discussed. We propose to use the model introduced in this chapter in computing the semantics of these extensions of logic programs. Exploring application areas is another important thrust of our research.

We developed two notions of generalising operators on fuzzy relations for neutrosophic relations. Of these, the stronger notion guarantees that any generalised operator is "well-behaved" for neutrosophic relation operands that contain consistent information.

For some well-known operators on fuzzy relations, such as union, join, projection, we introduced generalised operators on neutrosophic relations. These generalised operators maintain the belief system intuition behind neutrosophic relations, and are shown to be "well-behaved" in the sense mentioned above.

Our data model can be used to represent relational information that may be incomplete and inconsistent. As usual, the algebraic operators can be used to construct queries to any database systems for retrieving vague information.

Chapter 4

Soft Semantic Web Services Agent

Web services technology is critical for the success of business integration and other application fields such as bioinformatics. However, there are two challenges facing the practicality of Web services: (a) efficient location of the Web service registries that contain the requested Web services and (b) efficient retrieval of the requested services from these registries with high quality of service (QoS). The main reason for this problem is that current Web services technology is not semantically oriented. Several proposals have been made to add semantics to Web services to facilitate discovery and composition of relevant Web services. Such proposals are being referred to as Semantic Web services (SWS). However, most of these proposals do not address the second problem of retrieval of Web services with high QoS. In this chapter, we propose a framework called Soft Semantic Web Services Agent (soft SWS agent) for providing high QoS Semantic Web services using soft computing methodology. Since different application domains have different requirements for QoS, it is impractical to use classical mathematical modeling methods to evaluate the QoS of semantic Web services. We use neutrosophic neural networks with Genetic Algorithms (GA) as our study case. Simulation results show that the soft SWS agent methodology is extensible and scalable to handle fuzzy, uncertain and inconsistent QoS metrics effectively.

4.1   Introduction

Web services are playing an important role in e-business application integration and other application fields such as bioinformatics. So it is crucial for the success of both service providers as well as service consumers to provide and invoke the high quality of service (QoS) Web services. Unfortunately, current Web services technologies such as SOAP (Simple Object Access Protocol) [soa], WSDL (Web Services Description Language) [wsd], UDDI (Universal Description, Discovery and Integration) [udd], ebXML (Electronic Business XML Initiative) [ebx], XLANG [xla], WSFL (Web Services Flow Language) [wsf], BPEL4WS (Business Process Execution Language for Web Services) [bpe], and BSML (Bioinformatic Sequence Markup Language) [bsm] are all syntax-oriented with little or no semantics associated with them. Computer programs may read and parse them, but with little or no semantic information associated with these technologies, the computer programs can do little to reason and infer knowledge about the Web

services.

Current research trend is to add semantics to the Web services framework to facilitate the discovery, invocation, composition, and execution monitoring of Web services. Web services with explicit semantic annotation are called Semantic Web services (SWS). Several projects are underway to try to reach such a goal. For example, OWL-S (previously DAML-S [dama] from OWL Services Coalition [owla]) uses OWL based ontology for describing Web services. METEOR-S [SVSM03] follows the way that relates concepts in WSDL to DAML+OIL ontologies in Web services description, and then provides an interface to UDDI that allows querying based on ontological concepts. The Internet Reasoning Service (IRS-II) [MDCG03] is a Semantic Web services framework, which allows applications to semantically describe and execute Web services. IRS-II is based on the UPML framework [OCFB03]. The Web Service Modeling Framework (WSMF) [FB02] provides a model for describing the various aspects related to Web services. Its main goal is to fully enable e-commerce by applying Semantic Web technology to Web services.

In our vision, with the maturing of semantic Web services technologies, there will be a proliferation of public and/or private registries for hosting and querying semantic Web services based on specific ontologies. Currently, there are many public and private UDDI registries advertising numerous similar Web services with different QoS. For example, GenBank [gen], XEMBL [xem], and OmniGene [omn] all provide similar Web services with different quality of services. There are two challenges existing for automatic discovery and invocation of Web services. One is the efficient location of service registries advertising requested Web services and the another is the efficient retrieval of the requested services from these registries with the highest quality of service (QoS). The semantic Web services technologies that we mentioned above can be exploited to solve the first challenge. For the second challenge, we believe that the QoS of semantic Web services should cover both functional and non-functional properties. Functional properties include the input, output, conditional output, pre-condition, access condition, and the effect of service [mn]. These functional properties can be characterized as the capability of the service [ABH02]. Non-functional properties include the availability, accessibility, integrity, performance, reliability, regulatory, security, response time and cost [mn] of the Web service.

Several matchmaking schemes have already been proposed to match the service requestor's requirements with service provider's advertisement [GCTB, SKWL99, PPS]. These schemes basically try to solve the capability matching problem. Here, we must be aware that on the one hand, the degree of capability matching and non-functional properties are all fuzzy, and on the other hand, different application domains have different requirements on non-functional properties. As a consequence, it is not flexible to use classical mathematical modeling methods to evaluate the QoS of semantic Web services. Although there are several existing QoS models [CSZ92, FK98, Gar88, GGPS96, HSUW00, Rom95, SH90, SCMK02, ZBS97], none of them are suitable for the requirements considered in this chapter. These QoS models are based on precise QoS metrics and specific application domains. They cannot handle fuzzy and uncertain QoS metrics.

In this chapter, we propose a framework called *soft semantic Web services agent* (soft SWS agent) to provide high QoS semantic Web services based on specific domain ontology such as gnome. The soft SWS agent could solve the forementioned two challenges effectively and efficiently. The soft SWS agent itself is implemented as a semantic Web service and comprises of six components: (a) Registries Crawler, (b) Repository, (c) Inquiry Server, (d) Publish Server, (e) Agent Communication Server, and (f) Intelligent In-

ference Engine. The core of the soft SWS agent is Intelligent Inference Engine (IIE). It uses soft computing technologies to evaluate the entire QoS of semantic Web services using both functional and non-functional properties. In this chapter, we use semantic Web services for bioinformatics as a case study. We employ neutrosophic neural networks with Genetic Algorithms (GA) for the IIE component of our soft SWS agent. The case study illustrates the flexibility and reliability of soft computing methodology for handling fuzzy and uncertain linguistic information. For example, capability of a Web service is fuzzy. It is unreasonable to use crisp values to describe it. So we can use several linguistic variables such as a "little bit low" and "a little bit high" to express the capability of services.

The chapter is organized as follows. In section 2, we present the necessary background of the QoS model, semantic Web services, and soft computing methodology. In section 3, we provide the architecture of the extensible soft SWS agent. In section 4, we present the design of the neutrosophic neural network with GA and simulation results. In section 5, we present related work, and finally, in section 6, we present conclusions and possibilities for future research.

4.2 Background

This section details the background material related to this research. We cover traditional Web services, semantic Web, semantic Web services, soft computing methodology, and the QoS model.

4.2.1 Traditional Web services

Web services are modular, self-describing, and self-contained applications that are accessible over the internet [CNW01]. The core components of the Web services infrastructure are XML based standards like SOAP, WSDL, and UDDI. SOAP is the standard messaging protocol for Web services. SOAP messages consist of three parts: an envelope that defines a framework for describing what is in a message and how to process it, a set of encoding rules for expressing instances of application-defined datatypes, and a convention for representing remote procedure calls and responses. WSDL is an XML format to describe Web services as collections of communication endpoints that can exchange certain messages. A complete WSDL service description provides two pieces of information: an application-level service description (or abstract interface), and the specific protocol-dependent details that users must follow to access the service at a specified concrete service endpoint. The UDDI specifications offer users a unified and systematic way to find service providers through a centralized registry of services that is roughly equivalent to an automated online "phone directory" of Web services. UDDI provides two basic specifications that define a service registry's structure and operation. One is a definition of the information to provide about each service and how to encode it and the other is a publish and query API for the registry that describes how this information can be published and accessed.

4.2.2   Semantic Web

The current Web is just a collection of documents which are human readable but not machine processable. In order to remedy this disadvantage, the concept of semantic Web is proposed to add semantics to the Web to facilitate the information finding, extracting, representing, interpreting and maintaining. "The semantic Web is an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation" [BLHL01]. The core concept of semantic Web is ontology. "Ontology is a set of knowledge terms, including the vocabulary, the semantic interconnections, and some simple rules of inference and logic for some particular topic" [Hen01]. There are many semantic Web technologies available today, such as RDF [rdfb], RDFS [rdfa], DAML+OIL [damb] and OWL [owlb]. The description logics are used as the inference mechanism for current semantic Web technologies. There are some drawbacks in the description logics [SRT05]. It cannot handle fuzziness and uncertainty associated with concept membership. The current research trend is to combine soft computing with semantic Web [Str98, Str04, KLP97, DP04].

4.2.3   Semantic Web Services

The industry is proposing Web services to transform the Web from "passive state"–repository of static documents to "positive state"–repository of dynamic services. Unfortunately, the current Web services standards are not semantic-oriented. They are awkward for service discovery, invocation, composition, and monitoring. So it is natural to combine the semantic Web with Web services, the so-called semantic Web services. Several projects have been initiated to design the framework for semantic Web services such as OWL-S, IRS-II, WSMF and METEOR-S.

For example, OWL-S 1.0 which is based on OWL is the upper ontology for services. It has three subontologies: ServiceProfile, ServiceModel and ServiceGrounding. The service profile tells "what the service does"; this is, it gives the types of information needed by a service-seeking agent to determine whether the service meets its needs. The service model tells "how the service works"; that is, it describes what happens when the service is carried out. A service grounding specifies the details of how an agent can access a service. Typically a grounding will specify a communication protocol, message formats, and other service-specific details such as port numbers used in contacting the service. In addition, the grounding must specify, for each abstract type specified in the ServiceModel, an unambiguous way of exchanging data elements of that type with the service.

4.2.4   Soft Computing Methodology

"Soft computing differs from conventional (hard) computing in that, unlike hard computing, it is tolerant of imprecision, uncertainty, partial truth, and approximations" [Zad94]. The principal constituents of soft computing are fuzzy logic, neural networks, and generic algorithms. More and more technologies will join into the soft computing framework in the near future. Fuzzy logic is primarily concerned with handling imprecision and uncertainty, neural computing focuses on simulating human being's learning process, and genetic algorithms simulate the natural selection and evolutionary processes to perform randomized global

search. Each component of soft computing is complementary to each other. Using combinations of several technologies such as fuzzy-neural systems will generally get better solutions.

4.2.5 QoS Model

Different applications generally have different requirements of QoS dimensions. Rommel [Rom95] and Stalk and Hout [SH90] investigate the features with which successful companies assert themselves in the competitive world markets. Their result showed that success is based on three essential dimensions: time, cost and quality. [Gar88] associates eight dimensions with quality, including performance and reliability. Software systems quality of service has been extensively studied in [CSZ92, GGPS96, HSUW00, ZBS97]. For middleware systems, Frlund and Koisinen [FK98] present a set of practical dimensions for distributed object systems reliability and performance, which include TTR (time to repair), TTF (time to failure), availability, failure masking, and server failure. Gardaso, Miller, Sheth and Arnold [GMSA] propose a QoS model for workflows and Web services processes based on four dimensions: time, cost, reliability and fidelity.

In this paper, we construct a QoS model for semantic Web services. It is composed of the following dimensions: capability, response time, and trustworthiness. In order to be more precise, we give our definitions of the three dimensions as follows:

1. The *capability* of a semantic Web service can be defined as the degree to which its functional properties match with the required functional properties of the semantic Web service requestor;

2. The *response time* of a semantic Web service represents the time that elapses between service requests arrival and the completion of that service request. Response time is the sum of waiting time and actual processing time;

3. The *trustworthiness* of a semantic Web services is the extent to which it is consistent, reliabile, competent, and honest.

4.3 Architecture of Extensible Soft SWS Agent

The extensible soft SWS agent can provide high QoS semantic Web services based on specific ontology. The extensible SWS agent uses centralized client/server architecture internally. But itself can also be and should be implemented as a semantic Web service based on specific service ontology. The extensible soft SWS agent comprises of six components: (a) Registries Crawler; (b) SWS Repository; (c) Inquiry Server; (d) Publish Server; (e) Agent Communication Server; (f) Intelligent Inference Engine. The high level architecture of the extensible soft SWS agent is shown in Figure 1. Each of the components is described next.
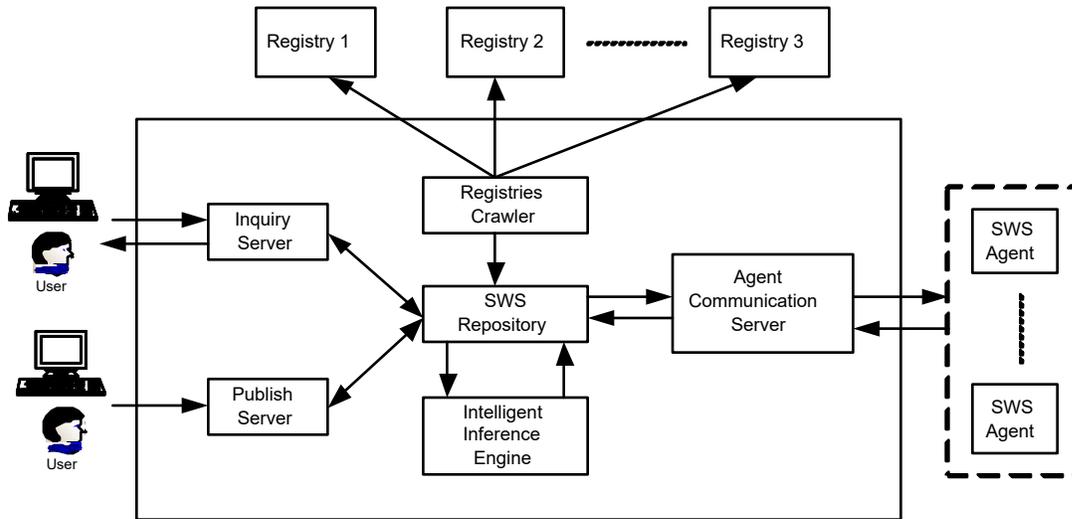
Figure 4.1: Architecture of the Extensible Soft SWS Agent

4.3.1   Registries Crawler

As we pointed out before, the current UDDI registry only supports keyword based search for the Web services description. Under the Semantic Web environment, UDDI registry must be extended to be ontology-compatible which supports semantic matching of semantic Web services' capabilities. One possible way is to map the OWL-S service profiles into current UDDI registry's data structure. Semantic Web service providers will publish the service profiles of semantic Web services in the public or private specific service ontology-oriented UDDI registries or directly on their semantic Web sites. The specific ontology based semantic Web services registries crawler has two tasks:

1. Accessing these public and private specific service ontology-oriented UDDI registries using UDDI query API to fetch the service profiles, transforming them into the format supported by our repository, and storing them into the repository using the publish API of our repository;

2. Crawling the semantic Web sites hosting the specific ontology based semantic Web services directly to get the service profiles, transforming them into the format supported by the repository, and storing them into repository using the publish API for the repository.

The registries crawler should be multithreaded and should be available 24x7. The registries crawler must also be provided the information of highest level specific service ontology before its execution.

4.3.2 SWS Repository

The specific ontology based semantic Web servcies repository will store service profiles of semantic Web services. The architecture of repository is shown in Figure 2.
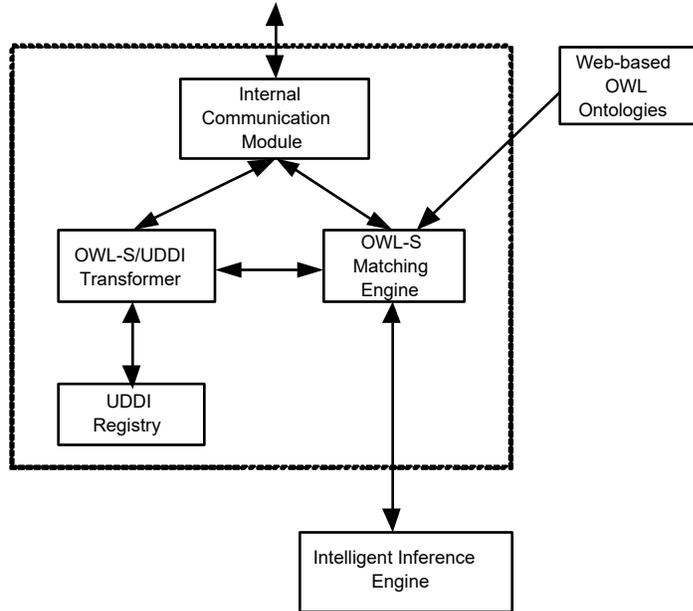

Figure 4.2: Architecture of Repository

The internal communication module provides the communication interface between the repository and the registries crawler, inquiry server, publish server, and the agent communication server. If a message is an advertisement, the internal communication module sends it to the OWL-S/UDDI transformer that constructs a UDDI service description using information about the service provider and the service name. The result of publishing with the UDDI is a reference ID of the service. This ID combined with the capability description and non-functional properties of the advertisement are sent to the OWL-S matching engine that stores the advertisement for capability matching. If a message is a query, the internal communication module sends the request to the OWL-S matching engine that performs the capability matching. After calculating the degree of capability, the OWL-S matching engine will feed the degree of capability and non-functional properties to the intelligent inference engine to get the entire Quality of Servie (QoS). The service with highest QoS will be selected. The result of the selection is the advertisement of the providers selected and a reference to the UDDI service record. The combination of UDDI records and advertisements is then sent to the inquiry server. If the required service does not exist, OWL-S matching engine will transfer the query to the agent communication server through the internal communication module. The matching algorithm used by OWL-S matching engine is based on the modified algorithm described in

[PKPS02]. The modified algorithm considers not only the inputs, outputs, preconditions and effects, but also service name.

4.3.3   Inquiry Server

The specific ontology based semantic Web services inquiry server provides two kinds of query interface: a programmatic API to other semantic Web services or agents and a Web-based interface for the human user. Both interfaces support keyword oriented query as well as capability oriented searches.

For capability oriented query, the inquiry server transforms the service request profile into the format supported by the repository such as OWL-S service profile and sends the query message to the internal communication module of the repository. The internal communication module sends the service profile to the OWL-S matching engine and returns back the requested advertisement to the inquiry server and then on to the service requestor. The process is shown in Figure 4.3:
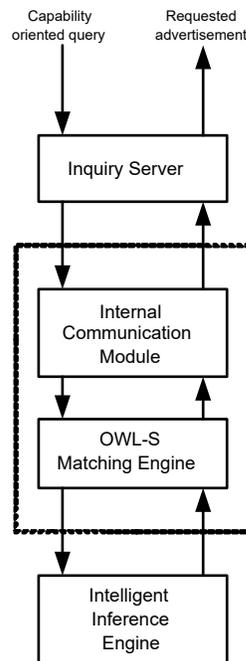

Figure 4.3: Capability oriented query

For the keyword oriented queries, the inquiry server will directly send the query string to the internal communication module as a query message and the internal communication module sends the query string to the UDDI Registry and returns back the requested UDDI records to the inquiry server and then on to the service requestor. The process is shown in Figure 4.4:
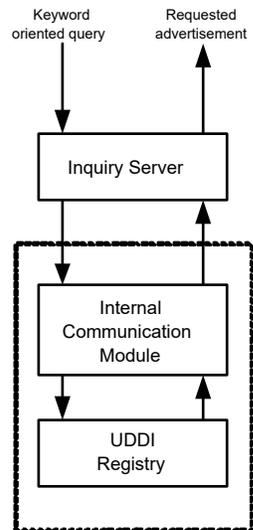
Figure 4.4: Keyword oriented query

We use SOAP as a communication protocol between service requestors and the inquiry server.

4.3.4 Publish Server

The specific ontology based semantic Web services publish server provides the publishing service for other agents and human users. It has two kinds of interface. One is the programmatic API to other semantic Web services or agents and another is for the human user which is Web-based. The publish server will transform the service advertisement into the format supported by the repository such as OWL-S service profile and sends the publish message to the internal communication module. The internal communication module sends the transformed OWL-S service profile to the OWL-S/UDDI transformer. The OWL-S/UDDI transformer will map the OWL-S service profile into UDDI registries data structure, and store the OWL-S service profile and reference ID of service into OWL-S matching engine. The process is shown in Figure 4.5:

If the advertised semantic Web services are not in the domain of the soft SWS agent, the internal communication server will transfer the advertisements to the agent communication server which will try to publish the advertisements into other soft SWS agents. SOAP is used as a communication protocol between service publisher and the publisher server.
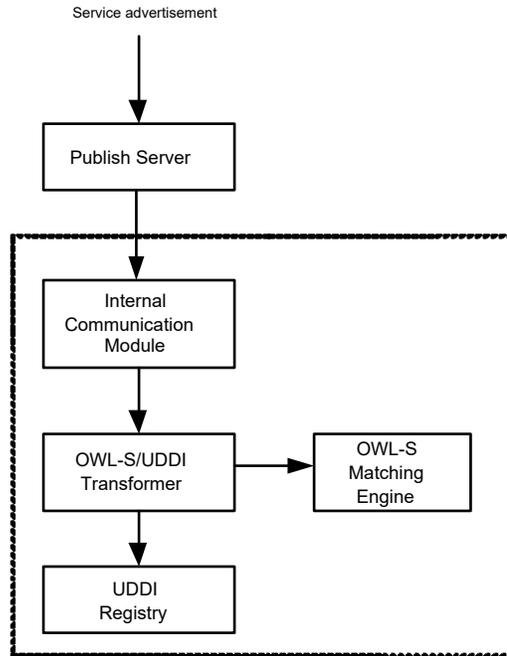
Figure 4.5: Publish service advertisement

4.3.5   Agent Communication Server

The soft semantic Web services agent communication server uses a certain communication protocol such as Knowledge Query and Manipulation Language (KQML) and Agent Communication Language (ACL) to communicate with other soft SWS agents. If the current soft SWS agent cannot fulfill the required services (query and publish), the agent communication server is responsible for transfering the requirements to other soft SWS agents, getting results back, and conveying the results back to the service requestors. The current KQML and ACL should be extended to be ontology-compatible to facilitate the semantic oriented communication.

4.3.6   Intelligent Inference Engine

The intelligent inference engine (IIE) is the core of the soft SWS agent. The soft SWS agent is extensible because IIE uses soft computing methodology to calculate the QoS of the semantic Web services with multidimensional QoS metrics. IIE gets the degree of capability matching and non-functional properties' values from OWL-S matching engine and returns back the whole QoS to OWL-S matching engine. In the next section, we show the design of an IIE using neutrosophic logic, neural networks, and genetic algorithms.
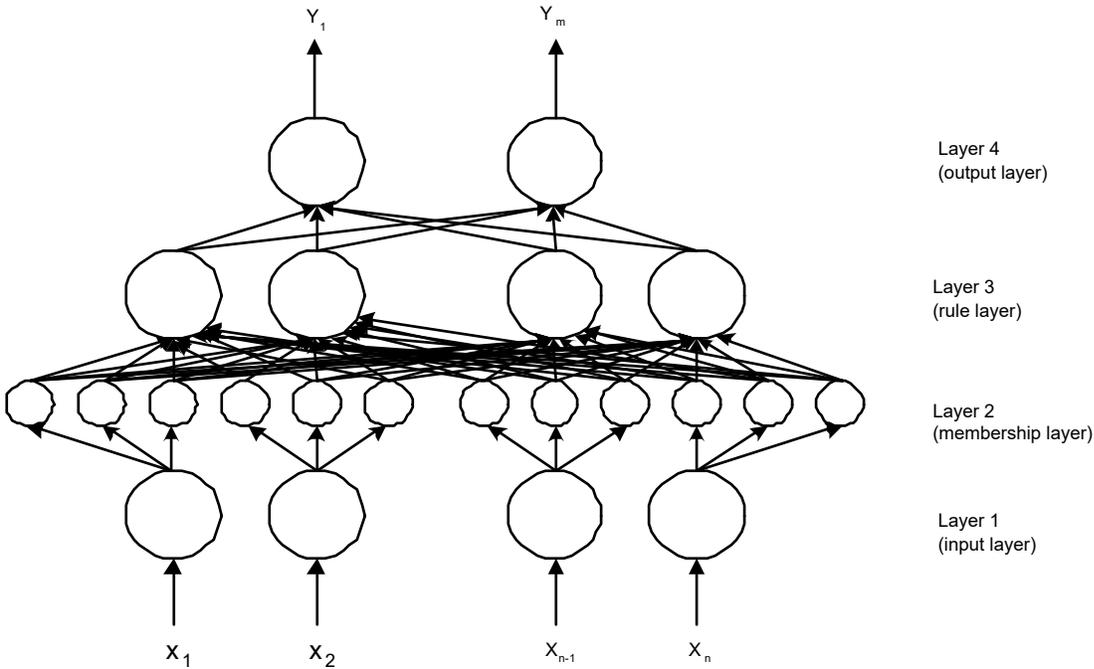
Figure 4.6: Schematic diagram of Neutrosophic Neural Network

4.3.7   Design of Intelligent Inference Engine

This section shows one implementation of IIE based on neutrosophic logic, neural network and genetic algorithm. A schematic diagram of the four-layered neutrosophic neural network is shown in Figure 3. Nodes in layer one are input nodes representing input linguistic variables. Nodes in layer two are membership nodes. Membership nodes are truth-membership node, indeterminacy-membership node and falsity-membership node, which are responsible for mapping an input linguistic variable into three possibility distributions for that variable. The rule nodes reside in layer three. The last layer contains the output variable nodes [LHL03].

As we mentioned before, the metrics of QoS of Semantic Web services are multidimensional. For illustration of specific ontology based Semantic Web services for bioinformatics, we decide to use capability, response time and trustworthiness as our inputs and whole QoS as output. The neutrosophic logic system is based on TSK model.

4.3.8   Input neutrosophic sets

Let x represent capability, y represent response time and z represent trustworthiness. We scale the capability, response time and trustworthiness to [0,10] respectively. The graphical representation of membership functions of x, y, and z are shown in Figure 4.

4.3.9   Neutrosophic rule bases

Here, we design the neutrosophic rule base based on the TSK model. A neutrosophic rule is shown below:

IF x is I_1 and y is I_2 and z is I_3 THEN O is $a_{i,1} * x + a_{i,2} * y + a_{i,3} * Z + a_{i,4}$.

where, I_1, I_2 and I_3 are in low, middle, and high respectively and i in [1,27]. There are totally 27 neutrosophic rules. The $a_{i,j}$ are consequent parameters which will be obtained by training phase of neutrosophic neural network using genetic algorithm.

4.3.10   Design of deneutrosophication

Suppose, for certain inputs x, y and z, there are m fired neutrosophic rules. To calculate the firing strength of jth rule, we use the formula:

$$W^j = W_x^j * W_y^j * W_z^j, \tag{4.1}$$

where
$W_x^j = (0.5 * t_x(x) + 0.35 * (1 - f_x(x)) + 0.025 * i_x(x) + 0.05),$
$W_y^j = (0.5 * t_y(y) + 0.35 * (1 - f_y(y)) + 0.025 * i_y(y) + 0.05),$
$W_z^j = (0.5 * t_z(z) + 0.35 * (1 - f_z(z)) + 0.025 * i_z(z) + 0.05),$
where $t_x, f_x, i_x, t_y, f_y, i_y, t_z, f_z, i_z$, are the truth-membership, falsity-membership, indeterminacy-membership of neutrosophic inputs x, y, z, respectively.

So the crisp output is:

$$O = \sum_{j=1}^{m} W^j * (a_{j,1} * x + a_{j,2} * y + a_{j,3} * z + a_{j,4})/(\sum_{j=1}^{m} W^j) \tag{4.2}$$

4.3.11   Genetic algorithms

GA is a model of machine learning which derives its behavior form a metaphor of the processes of evolution in nature. This is done by creation within a machine of a population of individuals represented by chromosomes. Here we use real-coded scheme. Given the range of parameters (coefficients of linear equations in TSK model), the system uses the derivate-free random search-GA to learn to find the near optimal solution by the fitness function through the training data.
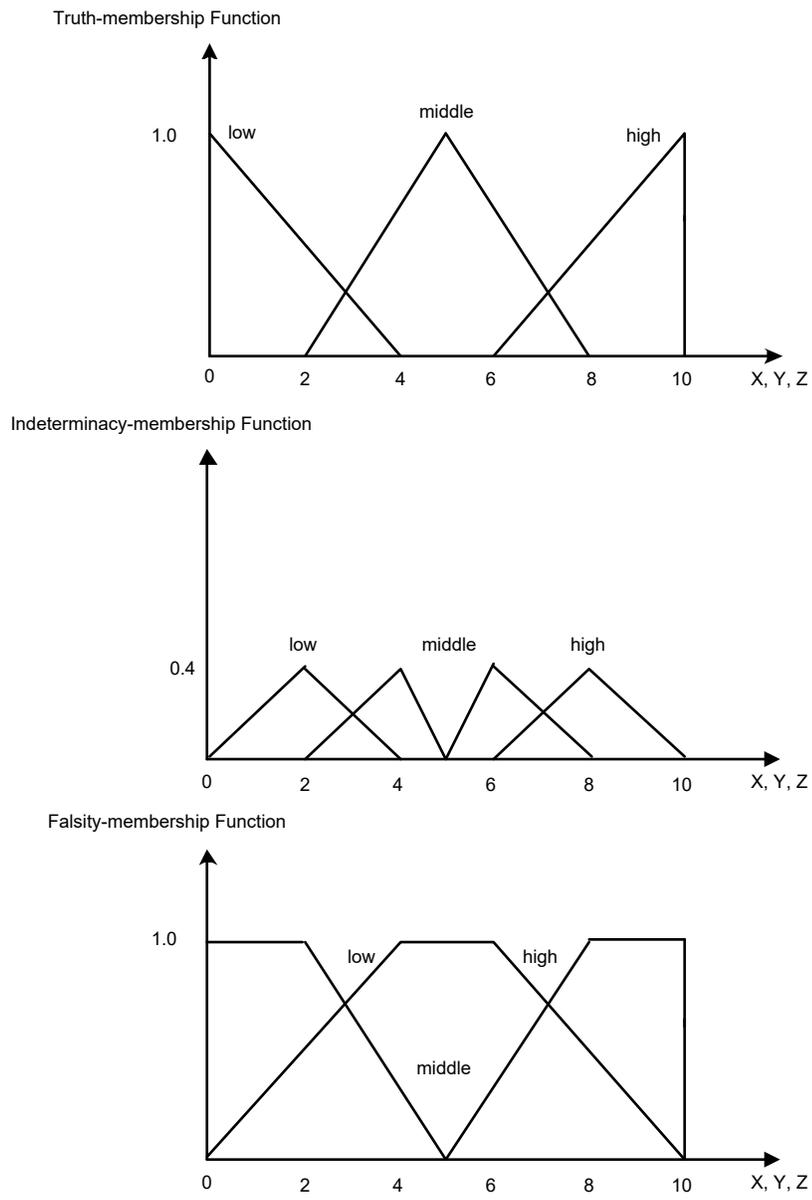
Figure 4.7: Membership functions of inputs

1. Chromosome: The genes of each chromosome are 108 real numbers (there are 108 parameters in the neutrosophic rule base) which are initially generated randomly in the given range. So each chromosome is a vector of 108 real numbers.

2. Fitness function: The fitness function is defined as

$$E = 1/2 \sum_{j=1}^{m} (d_i - o_i)^2 \tag{4.3}$$

3. Elitism: The tournament selection is used in the elitism process.

4. Crossover: The system will randomly select two parents among the population, then randomly select the number of cross points, and simply exchange the corresponding genes among these two parents to generate a new generation.

5. Mutation: For each individual in the population, the system will randomly select genes in the chromosome and replace them with randomly generated real numbers in the given range.

4.3.12 Simulations

There are two phases for applying a fuzzy neural network: training and predicting. In the training phase, we use 150 data entries as training data set. Each entry consists of three inputs and one expected output. We tune the performance of the system by adjusting the size of population, the number of generation and probability of crossover and mutation. Table 1 gives the part of prediction results with several parameters for output o.

In Table 1, No. of generation = 10000, No. of population = 100, probability of crossover = 0.7, probability of mutation = 0.3. The maximum error of prediction result is 1.64. The total prediction error for 150 entries of testing dataset is 19%. By our observation, designing reasonable neutrosophic membership functions and choosing reasonable training data set which is based on specific application domain can reduce the prediction error a lot. Here the example is just for illustration.

4.4 Related Work

MWSDI (METEOR-S Web Service Discovery Infrastructure) is an infrastructure of registries for semantic publication and discovery of Web services [VSSP04]. MWSDI supports creating registry federation by grouping registries that are mapped to the same node in Registries Ontology. MSWDI is based on the P2P model, so the registries are considered as peers. In our work, the soft SWS agents also can be regarded as peers. MWSDI uses the Registries Ontology to maintain a global view of the registries, associated domains and uses this information during Web service publication and discovery. The limitation of MWSDI is that it supports only capability matching of Web services and does not consider non-functional properties of

Table 4.1: Prediction Result of Neutrosophic Neural Network

Input x	Input y	Input z	Desired output	Real output o
1	0	1	0	0.51
1	2	5	1	1.71
1	4	7	2	2.59
3	2	9	3	3.52
3	6	7	4	3.81
3	10	7	5	4.92
5	8	9	6	5.43
7	10	7	7	5.90
7	10	9	8	6.45
9	10	9	9	7.36

Web services. The soft SWS agent can be viewed as an enhancement over MWSDI as it provides the service for discovering semantic Web services with the highest whole QoS.

The MWSDI approach annotates WSDL by associating its input and output types to domain specific ontologies and uses UDDI structures to store the mapping of input and output types in WSDL files to domain specific ontologies. It is similar to our work where we use OWL-S ontology directly to enable the semantic description of Web services.

SWWS (Semantic Web enabled Web Services) proposes a semantic-oriented service Registry which is similar to our idea [sww]. It has five components: Profile Crawler, UDDI Integration Engine, Registry API, Ontology Server and Query Interface. The service modelling ontology is stored in the ontology server. All individual service descriptions are stored as instances of the service description ontology and are also managed by the ontology server. SWWS does not support quality based semantic Web services discovery.

OASIS/ebXML describes an architecture of service registry [oas]. The registry provides a stable store where information submitted by a submitting organization is made persistent. Such information is used to facilitate ebXML based B2B partnerships and transactions. Submitted content may be XML schema and documents, process descriptions, ebXML Core Components, context descriptions, UML models, etc. It focuses mainly on the registry information model and discusses issues like object replication, object relocation and lifecycle management for forming registry federation. It does not use semantic Web and semantic Web services technologies.

4.5 Conclusions

In this chapter, we discussed the design of an extensible soft SWS agent and gave one implementation of Intelligent Inference Engine. The soft SWS agent supports both keyword based discovery as well as

capability based discovery of semantic Web services. The primary motivation of our work is to solve two challenges facing current Web services advertising and discovery techniques. One is how to locate the registry hosting required Web service description and another is how to find the required Web service with highest QoS in the located registry. The soft SWS agent solves both these problems efficiently and effectively. The soft SWS agent is built upon semantic Web, Web services, and soft computing technologies. The soft SWS agent could be used in WWW, P2P, or Grid infrastructures. The soft SWS agent is flexible and extensible. With the evolution of soft computing, more and more technologies can be integrated into the soft SWS agent. We used specific ontology based semantic Web services for bioinformatics and neutrosophic neural network with genetic algorithm as our study case. The training time is short and training results are satisfactory. The soft SWS agent will return the desired semantic Web services based on the entire QoS of semantic Web services. In the future, we plan to extend the architecture of the soft SWS agent to compute the entire QoS workflow of semantic Web services to facilitate the composition and monitoring of complex semantic Web services and apply it to semantic Web-based bioinformatics applications.

Bibliography

[ABH02]   A. Ankolekar, M. Burstein, and J. Hobbs, *Daml-s: Web service description for the semantic web*, The First International Semantic Web Conference (ISWC), 2002.

[AG90]   K. Atanassov and G. Gargov, *Intuitionistic fuzzy logic*, Compt. Rend. Acad. Bulg. Sci. **43** (1990), 9–12.

[AG98]   K. Atanassov and George Gargov, *Elements of intuitionistic fuzzy logic. part i*, Fuzzy Sets and Systems **95** (1998), 39–52.

[AR84]   M. Anvari and G. F. Rose, *Fuzzy relational databases*, Proceedings of the 1st International Conference on Fuzzy Information Processing (Kuaui, Hawaii), CRC Press, 1984.

[Ata86]   K. Atanassov, *Intuitionistic fuzzy sets*, Fuzzy Sets and Systems **20** (1986), 87–96.

[Ata88]   _____, *Two variants of intuitionistic fuzzy propositional calculus*, Preprint IM-MFAIS-5-88, 1988.

[Ata89]   _____, *More on intuitionistic fuzzy sets*, Fuzzy Sets and Systems **33** (1989), 37–46.

[Bag00]   R. Bagai, *Tuple relational calculus for paraconsistent databases*, Lecture Notes in Artificial Intelligence, vol. 1952, Springer-Verlag, 2000, pp. 409–416.

[Bal83]   J. F. Baldwin, *A fuzzy relational inference language for expert systems*, Proceedings of the 13th IEEE International Symposium on Multivalued Logic (Kyoto, Japan), 1983, pp. 416–423.

[Bel77a]   N. D. Belnap, *A useful four-valued logic*, Modern Uses of Many-valued Logic (G. Eppstein and J. M. Dunn, eds.), Reidel, Dordrecht, 1977, pp. 8–37.

[Bel77b]   _____, *A useful four-valued logic*, Modern Uses of Many-valued Logic (G. Eppstein and J. M. Dunn, eds.), Reidel, Dordrecht, 1977, pp. 8–37.

[Bis83]   J. Biskup, *A foundation of codds relational maybe–operations*, ACM Trans. Database Syst. 8 **4** (1983), 608–636.

[BLHL01]   T. Berners-Lee, J. Hendler, and O. Lassila, *The semantic web*, Scientific American **284** (2001), no. 5, 34–43.

[BMS84]   M. L. Brodie, J. Mylopoulous, and J. W. Schmidt, *On the development of data models*, On Conceptual Modelling (1984), 19–47.

[BP82]   B. P. Buckles and F. E. Petry, *A fuzzy representation for relational databases*, Fuzzy Sets Syst **7** (1982), 213–226.

[bpe]   *Business process execution language for web service (bpel4ws) 1.1 (may 2003)*.

[BS89]   H. A. Blair and V. S. Subrahmanian, *Paraconsistent logic programming*, Theoretical Computer Science **68** (1989), 135–154.

[BS95]   R. Bagai and R. Sunderraman, *A paraconsistent relational data model*, International Journal of Computer Mathematics **55** (1995), no. 1–2, 39–55.

[BS96a]   _____, *A bottom-up approach to compute the fitting model of general deductive databases*, Journal of Intelligent Information Systems **6** (1996), no. 1, 59–75.

[BS96b]   _____, *Computing the well-founded model of deductive databases*, The International Journal of Uncertainty, Fuzziness and Knowledge-based Systems **4** (1996), no. 2, 157–176.

[bsm]   *The bioinformatic sequence markup language (bsml) 3.1*.

[CK78]   S. K. Chang and J. S. Ke, *Database skeleton and its application to fuzzy query translation*, IEEE Trans. Softw. Eng. SE–4 (1978), 31–43.

[CNW01]   F. Curbera, W. Nagy, and S. Weerawarana, *Web services: Why and how*, Workshop on Object-Oriented Web Services-OOPSLA 2001 (Tampa, FL), 2001.

[Cod70]   E.F. Codd, *A relational model for large shared data banks*, Communications of the ACM **13** (1970), no. 6, 377–387.

[Cod79]   _____, *Extending the database relational model to capture more meaning*, ACM Transactions of Database Systems **4** (1979), no. 4, 397–434.

[Cos77a]   N. C. A. Da Costa, *On the theory of inconsistent formal systems*, Notre Dame Journal of Formal Logic **15** (1977), 621–630.

[Cos77b]   N.C.A.D. Costa, *On the theory of inconsistent formal systems*, Notre Dame Journal of Formal Logic **15** (1977), 621–630.

[CP87]   R. Cavallo and M. Pottarelli, *The theory of probabilistic databases*, Proceedings of the 13th Very Large Database Conference, 1987, pp. 71–81.

[CSZ92]   D. Clark, S. Shenker, and L. Zhang, *Supporting real-time applications in an integrated services packet network: Architecture and mechanism*, Proceedings of ACM SIGCOMM, 1992.

[dACM02a]   S. de Amo, W. Carnielli, and J. Marcos, *A logical framework for integrating inconsistent information in multiple databases*, Proc. PoIKS'02, LNCS 2284, 2002, pp. 67–84.

[dACM02b]   _____, *A logical framework for integrating inconsistent information in multiple databasses*, Proc. PoIKS'02, LNCS 2284, 2002, pp. 67–84.

[dama]   *Daml-s 0.9 draft release.*

[damb]   *Daml+oil (march 2001).*

[DP04]   Z. Ding and Y. Peng, *A probabilistic extension to ontology language owl*, Proceedings of the Hawai' International Conference on System Sciences, 2004.

[ebx]   *Electronic business xml initiative (ebxml).*

[EN00]   Elmasri and Navathe, *Fundamentals of database systems*, third ed., Addison–Wesley, New York, 2000.

[FB02]   D. Fensel and C. Bussler, *The web service modelling framework wsmf*, Electronic Commerce: Research and Application **1** (2002), 113–137.

[FK98]   S. Frlund and J. Koisinen, *Quality-of-service specification in distributed object systems*, Distributed System Engineering Journal **5** (1998), no. 4, 179–202.

[Gar88]   David A. Garvin, *Managing quality: The strategic and competitive edge*, Free Press, New York, 1988.

[GB93]   Wen-Lung Gau and Daniel J. Buehrer, *Vague sets*, IEEE Transactions on Systems, Man, and Cybernetics **23** (1993), no. 2, 610–614.

[GCTB]   J. Gonzalez-Castillo, D. Trastour, and C. Bartonili, *Description logics for matchmaking of services.*

[gen]   *Genebank database.*

[GGPS96]   L. Georgiadis, R. Guerin, V. Peris, and K. Sivarajan, *Efficient network qos provisioning based on per node traffic shaping*, IEEE/ACM Transactions on Networking **4** (1996), no. 4, 482–501.

[GMSA]   J. Gardoso, J. Miller, A. Sheth, and J. Arnold, *Modelling quality of service for workflows and web service processes.*

[Hen01]   J. Hendler, *Agents and the semantic web*, IEEE Intelligent Systems **16** (2001), no. 2, 30–37.

[HSUW00]   M. Hiltunen, R. Schlichting, C. Ugarte, and G. Wong, *Survivability through customization and adaptability: The cactus approach*, DARPA Information Survivability Conference and Exposition (DISCEX 2000), 2000.

[KLP97]   D. Koller, A. Levy, and A. Pfeffer, *P-classic: A tractable probabilistic description logic*, Proceedings of AAAI-97, 1997, pp. 390–397.

[KM98]   N. Karnik and J. Mendel, *Introduction to type-2 fuzzy logic systems*, Proc. 1998 IEEE Fuzz Conf., 1998, pp. 915–920.

[KY95]   George J. Klir and Bo Yuan, *Fuzzy sets and fuzzy logic: Theory and applications*, Prentice Hall, Upper Saddle River, New Jersey, 1995.

[KZ86]   J. Kacprzyk and A. Ziolkowski, *Database queries with fuzzy linguistic quantifiers*, IEEE Trans. Syst. Man Cybern. SMC–16 **3** (1986), 474–479.

[LHL03]   C.-H. Lee, J.-L. Hong, and Y.-C. Lin, *Type-2 fuzzy neural network systems and learning*, International Journal of Computational Cognition **1** (2003), no. 4, 79–90.

[Lip79]   W. Lipski, *On semantic issues connected with incomplete information databases*, ACM Trans. Database Syst. 4 **3** (1979), 262–296.

[Lip81]   ——, *On databases with incomplete information*, Journal of the Association for Computing Machinery **28** (1981), 41–70.

[LM00]   Qilian Liang and Jerry M. Mendel, *Interval type-2 fuzzy logic systems: Theory and design*, IEEE Transactions On Fuzzy Systems **8** (2000), no. 5, 535–550.

[LS90]   K.C. Liu and R. Sunderraman, *Indefinite and maybe information in relational databases*, ACM Transaction on Database Systems **15** (1990), no. 1, 1–39.

[LS91]   ——, *A generalized relational model for indefinite and maybe information*, IEEE Transaction on Knowledge and Data Engineering **3** (1991), no. 1, 65–77.

[Mai83]   D. Maier, *The theory of relational databases*, Computer Science Press, Rockville, Maryland, 1983.

[MDCG03]   E. Motta, J. Domingue, L. Cabral, and M. Gaspari, *A framework and infrastructure for semantic web services*, The Semantic Web-ISWC 2003, 2003.

[Men87]   E. Mendelson, *Introduction to mathematical logic*, Van Nostrand, Princeton, NJ, 1987, Third edition.

[MJ02]   J. Mendel and R. John, *Type-2 fuzzy sets made simple*, IEEE Transactions on Fuzzy Systems **10** (2002), 117–127.

[mn] *Understanding quality of service for web services.*

[NS92] R. Ng and V.S. Subrahmanian, *Probabilistic logic programming*, Information and Computation **101** (1992), no. 2, 150–201.

[oas] *Oasis/ebxml registry services specification v2.5 (june 2003).*

[OCFB03] B. Omelayenko, M. Crubezy, D. Fensel, and R. Benjamins, *Upml: The language and tool support for making the semantic web alive*, Spinning the Semantic Web: Bringing the WWW to its Full Potential, MIT Press, 2003, pp. 141–170.

[omn] *Omnigene: standardizing biological data interchange through web services technology.*

[owla] *Owl-s 1.0 release.*

[owlb] *Web ontology language (owl).*

[Par96] S. Parsons, *Current approaches to handling imperfect information in data and knowledge bases*, IEEE Trans. Knowledge and Data Engineering **3** (1996), 353–372.

[PKPS02] M. Paolucci, T. Kawamura, T. Payne, and K. Sycara, *Semantic matching of web services capabilities*, Proceedings of ISWC 2002, 2002.

[PPS] T. Payne, M. Paolucci, and K. Sycara, *Advertising and matching daml-s service descriptions.*

[Pra84] H. Prade, *Lipskiś approach to incomplete information databases restated and generalised in the setting of zadehś possibility theory*, Inf. Syst. 9 **1** (1984), 27–42.

[PT84] H. Prade and C. Testemale, *Generalizing database relational algebra for the treatment of incomplete or uncertain information and vague queries*, Information Sciences **34** (1984), 115–143.

[PT87] ———, *Representation of soft constraints and fuzzy attribute values by means of possibility distributions in databases*, Analysis of Fuzzy Information, Volume II, Artificial Intelligence and Decision Systems (1987), 213–229.

[rdfa] *Rdf vocabulary description language 1.0: Rdf schema.*

[rdfb] *Resource description framework.*

[RM88] K. V. S. V. N. Raju and A. K. Majumdar, *Fuzzy functional dependencies and lossless join decomposition of fuzzy relational database systems*, ACM Trans. on Database Syst. 13 **2** (1988), 129–166.

[Rom95] G. Rommel, *Simplicity wins: how germany's mid-sized industrial companies succeed*, Harvard Business School Press, Boston, Mass, 1995.

[SB95]   R. Sunderraman and R. Bagai, *Uncertainty and inconsistency in relational databases*, Advances in Data Management (S. Chaudhuri, A. Deshpande, and R. Krishnamurthy, eds.), Tata McGraw Hill, 1995, pp. 206–220.

[SCMK02]   A. Sheth, J. Cardoso, J. Miller, and K. Kochut, *Qos for service-oriented middleware*, Proceedings of COnference on Systemics, Cybernetics and Informatics (Orlando, FL), 2002.

[SH90]   G. Stalk and T. Hout, *Competing against time: how timebased competition is reshaping global markets*, Free Press, New York, 1990.

[SKS96]   A. Silberschatz, H. F. Korth, and S. Sudarshan, *Database system concepts*, third ed., McGraw–Hill, Boston, 1996.

[SKWL99]   K. Sycara, M. Klusch, S. Widoff, and J. Lu, *Dynamic service matchmaking among agents in open information environmen ts*, SIGMOD Record **28** (1999), no. 1, 47–53.

[Sma99]   Florentin Smarandache, *A unifying field in logics. neutrosophy: Neutrosophic probability, set and logic*, American Research Press, Rehoboth, 1999.

[Sma03]   _____, *A unifying field in logics: Neutrosophic logic. neutrosophy, neutrosophic set, neutrosophic probability and statistics*, Xiquan, Phoenix, 2003, third edition.

[soa]   *Simple object access protocol (soap) 1.2.*

[SRT05]   A. Sheth, C. Ramakrishnan, and C. Thomas, *Semantics for the semantic web: The implicit, the formal and the powerful*, International Journal on Semantic Web and Information Systems **1** (2005), no. 1, 1–18.

[Str98]   U. Straccia, *A fuzzy description logic*, Proceedings of AAAI-98, 1998.

[Str04]   _____, *Uncertainty and description logic program: A proposal for expressing rules and uncertainty on top of ontologies*, Tech. Report 2004-TR-14, ISTI-CNR, 2004.

[Sub94]   V. S. Subrahmanian, *Amalgamating knowledge bases*, ACM Transactions on Database Systems **19** (1994), no. 2, 291–331.

[SVSM03]   K. Sivashanmugam, K. Verma, A. Sheth, and J. Miller, *Adding semantics to web services standards*, International Conference on Web Services (ICWS'03), 2003, pp. 395–401.

[sww]   *Report on development of web service discovery framework (october 2003).*

[Tur86]   I. Turksen, *Interval valued fuzzy sets based on normal forms*, Fuzzy Sets and Systems **20** (1986), 191–210.

[udd]   *Universal description, discovery and integration (uddi) 3.0.1.*

[VSSP04] K. Verma, K. Sivashanmugam, A. Sheth, and A. Patil, *Meteor-s wsdi: A scalable p2p infrastructure of registries for semantic publication and discovery of web services*, Journal of Information Technology and Management (2004), In print.

[Won82] E. Wong, *A statistical approach to incomplete information in database systems*, ACM Trans. on Database Systems **7** (1982), 470–488.

[wsd] *Web services description language (wsdl) 1.1.*

[wsf] *Web services flow language (wsfl) 1.0.*

[WZS04] H. B. Wang, Y. Q. Zhang, and R. Sunderraman, *Soft semantic web services agent*, The Proceedings of NAFIPS 2004, 2004, pp. 126–129.

[xem] *Xml web services for embl (xembl).*

[xla] *Xlang: Web services for business process design (2001).*

[Zad65a] L. A. Zadeh, *Fuzzy sets*, Inf. Control 8 (1965), 338–353.

[Zad65b] L.A. Zadeh, *Fuzzy sets*, Inform. and Control **8** (1965), 338–353.

[Zad78] L. A. Zadeh, *Fuzzy sets as the basis for a theory of possibility*, Fuzzy Sets and Systems **1** (1978), 1–27.

[Zad94] L. Zadeh, *Fuzzy logic, neural networks, and soft computing*, Communications of the ACM **37** (1994), 77–84.

[ZBS97] J. Zinky, D. Bakken, and R. Schantz, *Architectural support for quality of service for corba objects*, Theory and Practice of Object Systems **3** (1997), no. 1.

[ZKLY04] Y.-Q. Zhang, A. Kandel, T.Y. Lin, and Y.Y. Yao, *Computational web intelligence: Intelligent technology for web applications, series in machine perception and artificial intelligence*, World Scientific, 2004, Volume 58.

This book presents the advancements and applications of neutrosophics. Chapter 1 first introduces the interval neutrosophic set which is an instance of neutrosophic set. In this chapter, the definition of interval neutrosophic sets and set-theoretic operators are given and various properties of interval neutrosophic sets are proved. Chapter 2 defines the interval neutrosophic logic based on interval neutrosophic sets including the syntax and semantics of first order interval neutrosophic propositional logic and first order interval neutrosophic predicate logic. The interval neutrosophic logic can reason and model fuzzy, incomplete and inconsistent information. In this chapter, we also design an interval neutrosophic inference system based on first order interval neutrosophic predicate logic. The interval neutrosophic inference system can be applied to decision making. Chapter 3 gives one application of interval neutrosophic sets and logic in the field of relational databases. Neutrosophic data model is the generalization of fuzzy data model and paraconsistent data model. Here, we generalize various set-theoretic and relation-theoretic operations of fuzzy data model to neutrosophic data model. Chapter 4 gives another application of interval neutrosophic logic. A soft semantic Web Services agent framework is proposed to facilitate the registration and discovery of high quality semantic Web Services agent. The intelligent inference engine module of soft semantic Web Services agent is implemented using interval neutrosophic logic.

US $25.00