Chapter 1

Preliminaries

1.1 Background

The section presents the needed background knowledge is introduced for clarifying the foundations of fuzzy logic. Lattice theory [**?**] is the key to bridge the corresponding relation between fuzzy set/relation and fuzzy logic, and it is illustrated in section 2.1.1. Fuzzy set/relation and fuzzy logic are represented in sections 2.1.2 and 2.1.3, respectively.

1.1.1 Lattice Theory

The definition of lattice is given by starting from set theory, followed by some important theorems and propositions as well as their proofs. Those theorems and propositions are used in chapter 3 to bridge fuzzy set/relation and fuzzy logic.

Definition 1.1.1. **Relation over multiple sets.** Let S_i be sets, where $i \in [1, n]$, a relation R over all the sets S_i is a subset of **Cartesian product** of the sets S_i, that is, $R \subseteq \Pi_{i \in [1,n]} S_i$.

Definition 1.1.2. **Relation over single set.** A relation R with arity n over a set S is a subset of **Cartesian product** S^n, that is, $R \subseteq S^n$.

Definition 1.1.3. **Binary relation over single set.** A binary relation R over a set S is subset of **Cartesian product** $S \times S$, that is, $R \subseteq S \times S$.

With respect to the binary relation, there are several interesting properties over it, such as reflexivity, antisymmetry, transitivity. Let R be a binary relation over S, and suppose x, y, z are elements in S.

Definition 1.1.4. **Partial order.** A binary relation R over S is called a partial order iff it satisfies the properties below,

- **Reflexivity** $\forall x$, if $x \in S$, then the pair $(x, x) \in R$

- **Antisymmetry** if $(x, y) \in R$, and $(y, x) \in R$, then $x = y$
- **Transitivity** if $(x, y) \in R$ and $(y, z) \in R$, then $(x, z) \in R$

Definition 1.1.5. **Partial ordered set.** A partial ordered set is a pair $(U, \preceq)$, where U is a set, and $\preceq$ is a partial order over U.

Example 1.1.6. $(\mathbb{N}, \leq)$ is partial order set, and so is $(\mathcal{P}(U), \subseteq)$, where $\mathcal{P}(U)$ is the power set of U, that is, $\mathcal{P}(U)$ is the set of all subsets of U.

Definition 1.1.7. **Supremum.** For subsets S of arbitrary partially ordered sets $(P, \preceq)$, a supremum or least upper bound of S is an element u in P such that

- u is a upper bound of S, notated as,

  $\forall x \in S.\ x \preceq u$
- u is a least upper bound of S, notated as,

  $\forall v \in P$ such that $x \preceq v$ for all x in S, it holds that $u \preceq v$

The special case is that a subset of $(P, \preceq)$ only with two elements, the supremum of such set is notated as $sup\{x, y\}$, which is used for the operations over fuzzy sets as their **join** and written as $x \vee y$.

Definition 1.1.8. **Infimum.** Formally, the infimum or greatest lower bound of a subset S of a partially ordered set $(P, \preceq)$ is an element m of P such that

- m is a lower bound of S, notated as,

  $\forall x \in S.\ m \preceq x$
- m is a greatest lower bound of S, notated as,

  $\forall v \in P$ such that $v \preceq x$ for all x in S, it holds that $v \preceq m$

The special case is that a subset of $(P, \preceq)$ only with two elements, the supremum of such set is notated as $inf\{x, y\}$, which is used for the operations over fuzzy sets as their **meet** and written as $x \wedge y$.

Definition 1.1.9. **Lattice.** A lattice is a partially ordered set $(P, \preceq)$ if every pair of elements of P has a sup and an inf in P.

Example 1.1.10. The partially ordered set $(\mathcal{P}(U), \subseteq)$ is a lattice. The *sup* (supremum) of two elements in $\mathcal{P}(U)$ is their union, and inf(infimum) is their intersection. The interval $[0, 1]$ is a lattice, with $inf\{x, y\} = min\{x, y\}$ and $sup\{x, y\} = max\{x, y\}$.

From the definition of lattice, if $(U, \preceq)$ is a lattice, then it comes equipped with the two binary operations **join**$(\vee)$ and **meet**$(\wedge)$ described above. From either of these binary operations, $\preceq$ can be reconstructed. In fact, $a \preceq b$ if and only if $a \wedge b = a$ if and only if $a \vee b = b$. These two binary operations satisfy a number of properties exposed in Theorem 1.1.11.

Theorem 1.1.11. If $(U, \preceq)$ is a lattice, then for all $a, b, c \in U$.

- $a \vee a = a$ and $a \wedge a = a$. ($\vee$ and $\wedge$ are **idempotent**.)
- $a \vee b = b \vee a$ and $a \wedge b = b \wedge a$. ($\vee$ and $\wedge$ are **commutative**.)
- $(a \vee b) \vee c = a \vee (b \vee c)$ and $(a \wedge b) \wedge c = a \wedge (b \wedge c)$. ($\vee$ and $\wedge$ are **associative**.)
- $a \vee (a \wedge b) = a$ and $a \wedge (a \vee b) = a$. (These are the **absorption identities**.)

Theorem 1.1.12. If U is a set with binary operations $\vee$ and $\wedge$ which satisfy the properties of Theorem 1.1.11, then defining $a \preceq b$ if $a \wedge b = a$ makes $(U, \preceq)$ a lattice whose **sup** and **inf** operations are $\vee$ and $\wedge$.

Proof. We first show that $a \wedge b = a$ **iff** $a \vee b = b$. Thus defining $a \preceq b$ if $a \wedge b = a$ is equivalent to defining $a \preceq b$ if $a \vee b = b$.

- To prove "If $a \wedge b = a$, then $a \vee b = b$."

  If $a \wedge b = a$, then

  $$\begin{aligned} a \vee b &= (a \wedge b) \vee b \\ &= b \vee (b \wedge a) && (\textbf{commutative}) \\ &= b && (\textbf{absorption identities}) \end{aligned}$$

- To prove "If $a \vee b = b$, then $a \wedge b = a$."

  If $a \vee b = b$, then

  $$\begin{aligned} a \wedge b &= a \wedge (a \vee b) \\ &= a && (\textbf{absorption identities}) \end{aligned}$$

$(U, \vee, \wedge)$ with the definition of $\preceq$ that $a \preceq b$ iff $a \wedge b = a$ makes $(U, \preceq)$ into a partial order set. We specify this point from the definition of partial order set and describe $\preceq$ with three properties over set $U \times U$.

- *reflexivity*

  It shows that $a \wedge a = a$ by **idempotent** rule, then from the definition of $a \preceq b$, which is "if $a \wedge b = a$, then $a \preceq b$", $a \preceq a$ is drawn.

- *antisymmetry*

  If $a \preceq b$, then $a \vee b = a$ and $b \preceq a$ implies $b \vee a = b$ by the definition of $\preceq$. since $a \vee b = b \vee a$ by **commutative**, it is drawn that $a = b$. Thus, If $a \preceq b$ and $b \preceq a$, then $a = b$.

- *transitivity* If $a \preceq b$ and $b \preceq c$, since $a \preceq b$ implies $a \wedge b = a$, while $b \preceq c$ implies $b \wedge c = b$, Then,

$$\begin{aligned} a \wedge c &= (a \wedge b) \wedge c \\ &= a \wedge (b \wedge c) \quad (\textbf{associative}) \\ &= a \wedge b \\ &= a \end{aligned}$$

  $a \wedge c = a$ implies that $a \preceq c$. Therefore, if $a \preceq b$ and $b \preceq c$, then $a \preceq c$.

The binary relation $\preceq$ built on $\vee$ and $\wedge$ satisfies *reflexivity*, *antisymmetry* and *transitivity*, which makes $\preceq$ a partial order. Thus, $(U, \preceq)$ is partial order set.

Next, to show the existence of sups, we claim that $sup\{a, b\} = a \vee b$. The proof is as follows, since we define $a \preceq b$ if $a \wedge b = a$, and $a \wedge (a \vee b) = a$ by the absorption identities, then $a \preceq a \vee b$ is drawn. While, $b \wedge (a \vee b) = b \wedge (b \vee a) = b$ by commutative and absorption rules, therefore $b \preceq a \vee b$. Since $a \preceq a \vee b$ and $b \preceq a \vee b$, $a \vee b$ is an upper bound of a and b. For any upper bound x, $a \vee x = x$ and $b \vee x = x$. $x = x \vee x = (a \vee x) \vee (b \vee x) = (a \vee b) \vee x$, makes $a \vee b \preceq x$. Thus, $a \vee b$ is a smallest upper bound of a and b, that is, $sup\{a, b\} = a \vee b$.

Claiming that $inf\{a, b\} = a \wedge b$, it will be proved as the existence of infs. $(a \wedge b) \wedge a = a \wedge (a \wedge b) = (a \wedge a) \wedge b = a \wedge b$, by the definition of $\preceq$, $a \wedge b \preceq a$. While $(a \wedge b) \wedge b = a \wedge (b \wedge b) = a \wedge b$, implies $a \wedge b \preceq b$. Since $a \wedge b \preceq a$ and $a \wedge b \preceq b$, $a \wedge b$ is a lower bound of a and b. For any lower bound x, $x \wedge a = x$ and $x \wedge b = x$. $x = x \wedge x = (x \wedge a) \wedge (x \wedge b) = x \wedge (a \wedge b)$ implies $x \preceq a \wedge b$, therefore, $a \wedge b$ is the largest lower bound of a and b, that is, $inf\{a, b\} = a \wedge b$.

□

The Theorem 1.1.12 shows different approaches to define lattice, as well as the point of view. A lattice could be seen as a partial order in which every pair of elements has an inf and a sup, or a set with a pair of operations which satisfy the properties mentioned in Theorem 1.1.11. In general case, $(U, \preceq)$ is a lattice and so is $(U, \vee, \wedge)$ with the properties in Theorem 1.1.11.

Some pertinent additional properties of a lattice $(U, \preceq)$ may have,

- *property 1:* 0 and 1 are **identities** for $\vee$ and $\wedge$ respectively.

  There is an element 0 in U such that $0 \vee a = a$ for all $a \in U$. There is an element $1 \in U$ such that $1 \wedge a = a$ for all $a \in U$.

- *property 2:* Each element in U has a **complement**

  Let U have identities, and for each element a in A, there is an element $a^{'}$ in U such that $a \wedge a^{'} = 0$ and $a \vee a^{'} = 1$.

- *property 3:* The binary operations $\vee$ and $\wedge$ **distribute** over each other.

  $a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$ and $a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$.

- *property 4:* Every subset T of U has a sup.

- *property 5:* Every subset T of U has an inf.

If a lattice has an identity for $\vee$ and an identity for $\wedge$, then it is a **bounded lattice**. A bounded lattice satisfying *property 2* is a **complemented lattice**. A lattice satisfying both distributive laws for $\vee$ and $\wedge$ in *property 3* is a **distributive lattice**. A bounded distributive lattice that is complemented is a **Boolean lattice**, or **Boolean Algebra**. A lattice satisfying *property 4* and *property 5* is a **complete lattice**.

The interval $[0, 1]$ is a complete lattice, and so is $(\mathcal{P}(U), \subseteq)$.

There is an operation denoted as $'$ with the following properties,

1. $(x^{'})^{'} = x$
2. $x \leq y$ implies that $y^{'} \leq x^{'}$

The operation " ' " on a bounded lattice is called an **involution**, or **duality**. If $'$ is an involution, the equations

$$(x \vee y)^{'} = x^{'} \wedge y^{'}$$
$$(x \wedge y)^{'} = x^{'} \vee y^{'}$$

are called the **De Morgan laws**, and may or may not hold. A system satisfying **De Morgan laws** $(V, \vee, \wedge, ', 0, 1)$ is a **De Morgan algebra**.

The lattice $([0, 1], \leq)$ is a De Morgan algebra, where its $'$ operation is $[0, 1] \to [0, 1] : x \to 1 - x$. The lattice $(\mathcal{P}(U), \subseteq, ', \emptyset, U)$ is also a De Morgan algebra, with $'$ as a complement operation on a subset of U.

Theorem 1.1.13. Let $(V, \vee, \wedge, ', 0, 1)$ be a De Morgan algebra and let U be any set. Let f and g be mappings from U into V. We define

- *Rule 1:* $(f \vee g)(x) = f(x) \vee g(x)$
- *Rule 2:* $(f \wedge g)(x) = f(x) \wedge g(x)$

- *Rule 3:* $f^{'}(x) = (f(x))^{'}$
- *Rule 4:* $0(x) = 0$
- *Rule 5:* $1(x) = 1$

Let V^U be the set of all mappings from U into V. Then $(V^U, \vee, \wedge, ', 0, 1)$ is a De Morgan algebra. If V is a complete lattice, then so is V^U.

Proof. Let f, g and h be arbitrary mappings from U to V, so they are elements in V^U. The lattice could be considered in two different ways, which are introduced in Theorem 1.1.12. One is, a lattice could be seen as a partial order in which every pair of elements has a **sup** and a **inf**. The other is, a lattice is viewed as a set with a pair of operations satisfying the properties in Theorem 1.1.11, where $\vee$, $\wedge$ have properties **idempotency**, **commutativity**, **associativity** and **identity absorption**. In order to prove that V^U is a complete lattice, there are two points to be demonstrated, one is that $(V^U, \vee, \wedge)$ is a lattice and the other is that each subset of V^U has an inf and a sup.

- $(V^U, \vee, \wedge)$ is a lattice.
  - **idempotency**
    $(f \vee f)(x) = f(x) \vee f(x)$ by **rule 1**, $f(x) \in V$ since $f : U \to V$, and $(V, \vee, \wedge)$ is a complete lattice. Therefore, $f(x) \vee f(x) = f(x)$ by using idempotency over V. Thus, $(f \vee f)(x) = f(x)$, that is idempotent holds in V^U. $(f \wedge f)(x) = f(x)$ could be proved in a similar way.
  - **commutativity**
    $$\begin{aligned} (f \vee g)(x) &= f(x) \vee g(x) && (\textbf{Rule 1}) \\ &= g(x) \vee f(x) && (\textbf{Commutative Rule} \text{ over } V) \\ &= (g \vee f)(x) && (\textbf{Rule 1}) \end{aligned}$$
    $(f \wedge g)(x) = f(x) \wedge g(x)$ could be proved in the same way.
  - **associativity**
    $$\begin{aligned} ((f \vee g) \vee h)(x) &= (f \vee g)(x) \vee h(x) && (\textbf{Rule 1}) \\ &= (f(x) \vee g(x)) \vee h(x) && (\textbf{Rule 1}) \\ &= f(x) \vee (g(x) \vee h(x)) \\ & \quad (\ \textbf{Associative Rule} \text{ over } V) \\ &= f(x) \vee ((g \vee h)(x)) && (\textbf{Rule 1}) \\ &= (f \vee (g \vee h))(x) && (\textbf{Rule 1}) \end{aligned}$$
    $((f \wedge g) \wedge h)(x) = (f \wedge (g \wedge h))(x)$ could be proved in the same way.
  - **absorption identities**
    $$\begin{aligned} (f \vee (f \wedge g))(x) &= f(x) \vee (f(x) \wedge g(x)) && (\textbf{Rule 1}) \\ &= f(x) \\ & \quad (\ \textbf{Absorption identities} \text{ over } V) \end{aligned}$$
    $(f \wedge (f \vee g))(x) = f(x)$ could be proved in the same way.

- Each subset of V^U has an inf and a sup.

  Since $(V^U, \vee, \wedge)$ is a lattice, then every pair of elements in V^U has an inf and a sup. It implies that each subset of V^U also has an inf and a sup, the proof can be done inductively, and can be found in [**?**].

$\square$

1.1.2 Fuzzy Set and Fuzzy Relation

The mathematical modeling of fuzzy concepts was presented by Zadeh in 1965, and his approach is described here. His intention is to represent the meaning in natural language as a matter of degree. For instance, for a proposition such as "Mary is young.", it is not always possible to be asserted as either true or false. It depends on the definition of concept "young", if "young" is given as a set

$$Young = \{x | x \in [0, \infty), x < 23\}$$

and Mary's age is given as 21, then the proposition "Mary is young" is true under the definition of "young" above. It seems that classical set theory solves the problem of representing the imprecise information such as 'young' in the the example above. However, 18 and 20 year olds are young, but with different degrees: 18 is younger than 20. This suggests that the membership in 'Young' concept should not be on a 0 or 1 basis, but rather on 0 to 1 scale, that is, the membership should be an element of the interval [0,1].

For representation of membership with degree, fuzzy set is defined in the same way as the definition of crisp set.

Definition 1.1.14. **Crisp Set.** Let U be the universal set, an ordinary subset A of U is determined by its **indicator function**, or **characteristic function** χ_A defined by

$$\chi_A(x) = \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}$$

The indicator function of a subset A of the universal set U specifies whether an element is in A or not. Only two values can be taken for the characteristic function, 0 and 1. The characteristic function could be generalized as $\chi_A : U \to \{0, 1\}$.

According to the definition above, crisp set only has 0, 1 as there membership degree. However, in fuzzy set, the degree value is a real number in $[0, 1]$.

Definition 1.1.15. **Fuzzy Set.** Let U be the universal set, which includes all the elements in our interested domain. A **fuzzy subset** A of the universal set U is a function $\mu_A : U \to [0, 1]$. It is common to refer to a fuzzy subset simply as a **fuzzy set**.

The image of the fuzzy function μ_A contains two values 0 and 1, which correspond to the image of crisp subset of U. Therefore, crisp subsets are considered as special cases of fuzzy subsets.

It is customary in fuzzy literature to represent fuzzy set in both notations, A and μ_A. The first one is called 'linguistic label', which indicates the concept of interested domain. For example, Let A be a set of young people. Then, the $\mu_A : U \to [0, 1]$ is called fuzzy function, representing the degree of youngness that has been assigned to each member in U.

Suppose that all the numbers which are specified as ages belong to U, then a value in $[0, 1]$ is assigned to each number in U. For example, $\mu_{young}(18) = 0.85$, $\mu_{young}(20) = 0.81$, then "18 is younger then 20" has its semantic meaning under μ_{young} or concept 'young', which is $\mu_{young}(18) > \mu_{young}(20)$.

1.1.2.1 Operations on fuzzy set

Following the definition above, a crisp subset A of universal set U can be represented by a function $\chi_A : U \to \{0, 1\}$, while a fuzzy subset $\tilde{A}$ of universal set U is indicated as a function $\mu_{\tilde{A}} : U \to [0, 1]$. On the subsets of U, whether they are crisp or fuzzy, the operations of union, intersection, and complement are given as follows.

The original definition is given by the rules

$$A \cup B = \{x|x \in A \text{ or } x \in B\}$$

$$A \cap B = \{x|x \in A \text{ and } x \in B\}$$

$$A' = \{x \in U|x \notin A\}$$

From the point of view of function, the operations over crisp subsets are:

$$\chi_{A\cup B}(x) = max\{\chi_A(x), \chi_B(x)\} = \chi_A(x) \vee \chi_B(x)$$

$$\chi_{A\cap B}(x) = min\{\chi_A(x), \chi_B(x)\} = \chi_A(x) \wedge \chi_B(x)$$

$$\chi_{A'}(x) = 1 - \chi_A(x)$$

Naturally, by the membership function, these operations over fuzzy subsets of U are extended as,

$$\mu_{\tilde{A}\cup\tilde{B}}(x) = sup\{\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)\} = \mu_{\tilde{A}}(x) \vee \mu_{\tilde{B}}(x)$$

$$\mu_{\tilde{A}\cap\tilde{B}}(x) = inf\{\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)\} = \mu_{\tilde{A}}(x) \wedge \mu_{\tilde{B}}(x)$$

$$\mu_{\tilde{A}'}(x) = 1 - \mu_{\tilde{A}}(x)$$

The syntax $\vee$ and $\wedge$ are the operations over sets, which are mentioned in section 2.1 Lattice. Note that they are different from the logic notation used for 'or' and 'and'.

1.1.2.2 Fuzzy Relation

Relations, or associations among objects, are of fundamental importance in the analysis of real-world system. Since the real-world is full of vague and uncertain information, the fuzzy relation plays an important role to represent it.

The classical relation has been defined in the lattice section. In this subsection, the fuzzy relation will be generalized from the crisp one.

Definition 1.1.16. **Fuzzy Relation.** An **n-ary fuzzy function relation in a set** $V = U_1 \times U_2 \times ... \times U_n$ is a fuzzy subset R of V, that is, a function $R : V \to [0, 1]$. If the sets U_i are identical, say U, then R is an **n-ary fuzzy relation on** U.

1.1.3 Fuzzy Logic

1.1.3.1 Syntax

Let Σ be a set of function symbols, V be a set of variables and Π be a set of predicate symbols. Terminology such as terms, atoms are formally defined.

Definition 1.1.17. **Term**

- t is a term, if $t \in V$
- $f(t_1, ..., t_n)$ is a term, if $f \in \Sigma$ and t_i are terms, where $i \in [1, n]$.
- nothing else is term.

A special case is that f's arity is 0, then the term is called *constant*. The set of all the terms is called **term universe**, notated as $TU_{\Sigma,V}$.

Definition 1.1.18. **Atom** $p(t_1, ..., t_n)$ is an atom if $p \in \Pi$ and t_i are terms.

A special case is that p's arity is 0, then the atom is called *proposition*. The set of all the atoms built under V, Σ, Π is called **term base**, notated as $TB_{\Pi,\Sigma,V}$.

Terms and atoms are called *ground* if they do not contain variables. The **Herbrand universe** $\mathbb{HU}$ is a set of all ground terms, and the **Herbrand base** $\mathbb{HB}$ is a set of all ground atoms.

1.1.3.2 Semantics

In classical logic, the interpretation of formula only could be 0 or 1, but in fuzzy interpretation, the real number $v \in [0, 1]$ is assigned to an *atom*.

Let $\mathbb{T}$ be an interval $[0, 1]$, which is a value pool the atoms will be assigned to. $(\mathbb{T}, \leq)$ is a complete lattice as we know from lattice section. A *valuation* $\sigma : V \rightarrow \mathbb{HU}$ is an assignment of ground terms to variables. Each valuation σ uniquely constitutes a mapping $\hat{\sigma} : TU_{\Pi,\Sigma,V} \rightarrow \mathbb{HB}$ that is defined in the obvious way. A *fuzzy Herbrand interpretation* of a fuzzy logic is a mapping $I : \mathbb{HB} \rightarrow \mathbb{T}$ that assigns truth values to ground atoms.

For two interpretations I and J, we say I *is less than or equal to* J, written $I \sqsubseteq J$, iff $I(A) \preceq J(A)$ for all $A \in \mathbb{HB}$. Two interpretations I and J are equal, written $I = J$, iff $I \sqsubseteq J$ and $J \sqsubseteq I$. Minimum and maximum for interpretations are defined from $\preceq$ as usual. Accordingly, the infimum and supremum of interpretation are, for all $A \in \mathbb{HB}$, defined as $(I \sqcap J)(A) = min(I(A), J(A))$ and $(I \sqcup J)(A) = max(I(A), J(A))$.

1.2 RFuzzy Framework

RFuzzy framework is a Prolog-based tool for representing and reasoning with fuzzy information. In this section we present a brief overlook for the RFuzzy framework [**?**] in this chapter to familiarize the reader with the syntax and semantics which we work on in similarity and quantification.

1.2.1 Introduction

Logic programming [**?**] has been successfully used in knowledge representation and reasoning for decades. Indeed, world data in not always perceived in a crisp way. Information that we gather might be imperfect, uncertain, or fuzzy in some other way. Hence the management of uncertainty and fuzziness is very important in knowledge representation.

Introducing Fuzzy Logic into Logic Programming has provided the development of several fuzzy systems over Prolog. These systems replace its inference mechanism, SLD-resolution, with a fuzzy variant that is able to handle partial truth. Most of these systems implement the fuzzy resolution introduced by Lee in [**?**], as the Prolog-Elf system, the FRIL Prolog system [**?**] and the F-Prolog language [**?**]. However, there is no common method for fuzzifying Prolog, as noted in [**?**].

One of the most promising fuzzy tools for Prolog is RFuzzy Framework. The most important advantages against the other approaches are:

1. A truth value is represented as a real number in unit interval $[0, 1]$, satisfying certain constrains.
2. A truth value is propagated through the rules by means of an aggregation operator. The definition of this aggregation operator is general and is subsumes conjunctive operators (triangular norms [**?**] like min, prod, etc.), disjunctive operators [**?**] (triangular co-norms, like max, sum, etc.), average operators (average as arithmetic average, quasi-linear average, etc.) and hybrid operators (combinations of the above operators [**?**]).
3. Crisp and fuzzy reasoning are consistently combined
4. It provides some interesting improvements with respect to FLOPER [**?**, **?**]: default values, partial default values, typed predicates and useful syntactic sugar (for presenting facts, rules and functions).

1.2.2 Syntax

From Σ a set of function symbols and a set of variables V, the *term universe* $TU_{\Sigma,V}$ is built, whose elements are *terms*. It is the minimal set such that each variable is a term and terms are closed under Σ operations. In particular, constant symbols are terms, which is a function with arity 0. The creation procedure of *term universe* is described in definition 2.3.1.

Term base $TB_{\Pi,\Sigma,V}$ is defined from a set of predicate symbols and *term universe* $TU_{\Sigma,V}$. The elements in the *term base* are called *atoms*, which are predicates whose arguments are elements of $TU_{\Sigma,V}$. The building procedure is described in definition 2.3.2.

Atoms and terms are called *ground* if they do not contain variables. The *Herbrand universe* $\mathbb{HU}$ is the set of all ground terms, and the *Herbrand base* $\mathbb{HB}$ is the set of all atoms with arguments from the *Herbrand universe*.

Ω is a set of *many-valued connectives* and includes,

1. conjunctions $\&_1$, $\&_2$, ..., $\&_k$
2. disjunctions $\vee_1$, $\vee_2$, ..., $\vee_l$
3. implications $\leftarrow_1$, $\leftarrow_2$, ..., $\leftarrow_m$
4. aggregations $@_1$, $@_2$, ..., $@_n$
5. real numbers $v \in [0, 1] \subset \mathbb{R}$. These connectives are of arity 0, that is, $v \in \Omega^{(0)}$.

While Ω denotes the set of connectives symbols, $\hat{\Omega}$ denotes a set of associated truth functions. Instances of connective symbols and their truth functions are denoted by F and $\hat{F}$ respectively, so $F \in \Omega$, and $\hat{F} \in \hat{\Omega}$.

Definition 1.2.1. **(fuzzy clauses).** A fuzzy clause is written as,

$$A \overset{c,F_c}{\leftarrow} F(B_1, ..., B_n)$$

where $A \in TB_{\Pi,\Sigma,V}$ is called head, and $B_i \in TB_{\Pi,\Sigma,V}$ is called body, $i \in [1, n]$. $c \in [0, 1]$ is credibility value, and $F_c \in \{\&_1, ..., \&_k\} \subset \Omega^{(2)}$ and $F \in \Omega^{(n)}$ are connective symbols for the credibility value and the body, respectively.

A *fuzzy fact* is a special case of a clause where $c = 1$, F_c is the usual multiplication of real numbers "." and $n = 0$. It is written as $A \leftarrow v$, where the c and F_c are omitted.

Example 1.2.2.

$$good - destination(X) \overset{1.0,.}{\leftarrow} .(nice - weather(X), many - sight(X)).$$

This fuzzy clause models to what extent cities can be deemed good destinations - the quality of the destination depends on the weather and the availability of sights. The credibility value of the rule is 1.0, which means that we have no doubt about this relationship. The connectives used here in both cases is the usual multiplication of real numbers.

Definition 1.2.3. **(default value declaration).** A *default value declaration* for a predicate $p \in \Pi^{(n)}$ is written as

$$default(p/n) = [\delta_1 \text{ if } \varphi_1, ..., \delta_m \text{ if } \varphi_m]$$

where $\delta_i \in [0, 1]$ for all i. The φ_i are FOL(First-Order Logic) formulas restricted to terms from TU_{Σ,V_p}, the predicates $=$ and $\neq$, the symbol true and the junctors $\vee$ and $\wedge$ in their usual meaning.

Example 1.2.4.

$$default(nice - weather(X)) = 0.5$$

Definition 1.2.5. **(type declaration).** Types are built from terms $t \in \mathbb{HU}$. A *term type declaration* assigns a type $\tau \in \mathcal{T}$ to a term $t \in \mathbb{HU}$ and is written as $t : \tau$. A *predicate type declaration* assigns a type $(\tau_1, ..., \tau_n) \in \mathcal{T}^n$ to a predicate $p \in \Pi^n$ and is written as $p : (\tau_1, ..., \tau_n)$, where τ_i is the type of p's $i - th$ argument. The set composed by all term type declarations and predicate type declarations is denoted by T.

Example 1.2.6. Suppose that the set of types is $\mathcal{T} = \{City, Continent\}$, here are some examples of *term type declaration* and *predicate type declaration*.

$$madrid : City, sydney : City$$

$$africa : Continent, america : Continent$$

$$nice - weather : (City)$$

$$city - continent : (City, Continent)$$

In this example, "madrid", "sydney", "africa" and "america" are ground terms. "madrid" and "sydney" have type City. "africa" and "america" have type Continent. "nice-weather" and "city-continent" are fuzzy predicates. "nice-weather" is of type (City) and "city-continent" is of type (City, Continent).

Definition 1.2.7. **(well-typed).** A ground atom $A = p(t_1, ..., t_n) \in \mathbb{HB}$ is *well_typed* with respect to T **iff** $p : (\tau_1, ..., \tau_n) \in T$ implies that the term type declaration $(t_i : \tau_i) \in T$ for $1 \leq i \leq n$.

A ground clause $A \overset{c,F_c}{\leftarrow} F(B_1, ..., B_n)$ is *well_typed* w.r.t T **iff** all B_i are *well_typed* for $1 \leq i \leq m$ implies that A is *well_typed*. A non-ground clause is *well_typed* **iff** all its ground instances are well_typed.

Definition 1.2.8. **(well-defined RFuzzy program).** A fuzzy program $P = (R, D, T)$ is called *well-defined* **iff**

- for each predicate symbol p/n, there exist both a predicate type declaration and a default value declaration.
- all clauses in R are *well_typed*.
- for each default value declaration

  $$default(p(X_1, ..., X_n)) = [\delta_1 \text{ if } \varphi_1, ..., \delta_m \text{ if } \varphi_m].$$

  the formulas φ_i are pairwise contradictory and $\varphi_1 \vee ... \vee \varphi_m$ is a tautology.

1.2.3 Semantics

A *valuation* $\sigma : V \rightarrow \mathbb{HB}$ is an assignment of ground terms to variables. Each valuation σ uniquely constitutes a mapping $\hat{\sigma} : TB_{\Pi,\Sigma,V} \rightarrow \mathbb{HB}$ that is defined in the obvious way.

A *fuzzy Herbrand interpretation* of a fuzzy logic program is a mapping $I : \mathbb{HB} \rightarrow \mathbb{T}$ that assigns truth values to ground atoms.

Definition 1.2.9. **(Model).** Let $P = (R, D, T)$ be a fuzzy logic program.

For a clause $r \in R$ of the form $A \overset{c,F_c}{\longleftarrow} F(B_1, ..., B_n)$, then I is a model of the clause r and is written as

$$I \models A \overset{c,F_c}{\longleftarrow} F(B_1, ..., B_n)$$

iff for all valuations σ:

$$\text{if } I(\sigma(B_i)) = v_i > 0 \text{ for all } i \text{ then } I(\sigma(A)) \geq v'$$

where $v' = \hat{F}_c(c, \hat{F}(v_1, ..., v_n))$.

For a clause $r \in R$ of the form $A \leftarrow v$, then I is a model of a clause r and is written as,

$$I \models A \leftarrow v$$

iff for all valuations σ:

$$I(\sigma(A)) \geq v$$

For a default value declaration $d \in D$, then I is a model of the default value declaration d and is written as,

$$I \models default(p/n) = [\delta_1 \text{ if } \varphi_1,\ ...,\ \delta_m \text{ if } \varphi_m]$$

iff for all valuation σ :

$$I(\sigma(p(X_1, ..., X_n))) \geq \delta_i$$

where φ_j is the only formula that holds for $\sigma(\varphi_j)(1 \leq j \leq m)$.

$I \models R$ **iff** $I \models r$ for all $r \in R$ and similarly, $I \models D$ **iff** $I \models d$ for all $d \in D$. Finally, $I \models P$ **iff** $I \models R$ and $I \models D$.