

Fuzzy Semantics of Contract Language

Hengyang Wu^{1,2}, Yixiang Chen^{2*}

¹Information Engineer College, Hangzhou Dianzi University, Hangzhou, China

²MoE Engineering Center for Software/Hardware Co-Design Technology and Application, East China Normal University, Shanghai, China

Email: wuhengy_1974@aliyun.com, *yxchen@sei.ecnu.edu.cn

Received 5 November 2015; accepted 15 March 2016; published 18 March 2016

Copyright © 2016 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

In this paper, we focus on investigation of the predicate transformer semantics of the contract language introduced by Back and von Wright in their book titled as “Refinement Calculus: A Systematic Introduction” (Springer-Verlag, New York, 1998) in the framework of fuzziness. In order to define fuzzy operations, *i.e.*, fuzzy logic connectives, we take into account implicator \rightarrow and its associated T-norm \otimes based on residuated lattice theory. Based on these basic fuzzy operations, we introduce the angelic and demonic updates of fuzzy relations. They are the basis of fuzzy predicate transformers in the sense of that any strongly monotone fuzzy predicate transformer can be represented as the sequential composition of the angelic and demonic updates. Together with the standard strong negation $\neg x = 1 - x$, we set up the duality between the angel and demon. The fuzzy predicate transformers semantics of contract statements is established and a simple example of contract statements is given.

Keywords

Contract Language, Formal Semantics, Residuated Lattice, Fuzzy Predicate Transformers

1. Introduction

There are four main approaches to describe the semantics of some language, *i.e.*, the denotational [1], the operational [2], the axiomatic [3] and the algebraic [4] approaches. Predicate transformers semantics [5] [6] is a kind of axiomatic approach to describe semantics of languages. Each statement in a language is characterized by the way it transforms postconditions to preconditions.

Back and von Wright in [7] introduce a contract language (a detailed introduction in Section 2) describing games that are played between two opponents, called the angel and the demon. The syntax of a contract state-

*Corresponding author.

ment S is given by (in BNF-form)

$$S ::= \{P\} \mid [P] \mid S_1; S_2 \mid (\bigcap_{i \in I} S_i) \mid (\bigcup_{i \in I} S_i).$$

Here P is a relation term and I is a set of higher-order logic which is simply thought of as an index set.

Back and von Wright investigate the transformer semantics of contract language in the Boolean case that the relation P is a classical relation P from X to Y . The angelic update $\{P\}$ in an initial state x establishes the postcondition $q \subseteq Y$ if there is at least one final state y with $(x, y) \in P$ satisfying q . The state y satisfying q means $y \in p$ or $q(y) = 1$. Therefore,

$$\{P\}(q)(x) = (\exists y \in Y. P(x, y) \wedge q(y)) = \sup \{P(x, y) \wedge q(y) \mid y \in Y\}.$$

Similarly, the demonic update $[P]$ in an initial state x establishes postcondition q if all final state y with $(x, y) \in P$ satisfy q , described by

$$[P](q)(x) = (\forall y \in Y. P(x, y) \rightarrow q(y)) = \inf \{P(x, y) \rightarrow q(y) \mid y \in Y\}.$$

For a given postcondition q , both angelic and demonic updates define preconditions over initial states, describing if initial states satisfy those updates (preconditions), then their outputs of program at these initial states will satisfy the postcondition q . Therefore, both updates define predicate transformers.

Ying in [8] investigates the probabilistic case of Back-Wright contract language. He generalizes the boolean relation to the probabilistic one. Ying sets up the probabilistic semantics of the contract statements based on the probabilistic logic through probabilistic predicate transformers [6] [9]. The probabilistic model for the guarded language is also studied in [10].

This paper focuses on the establishment of fuzzy semantics of contract language based on fuzzy predicate transformers [11]. Here, the relation P is a fuzzy relation [12] from initial states to final states. For initial state x and final state y , the value $P(x, y)$, the membership degree of state (x, y) belonging to the fuzzy set P , describes indeed the possibility degree of the y to be the output of this program that fuzzy relation P defines from the x . Our goal is twofold: (1) We set up the fuzzy predicate transformer semantics of contract statement S , which maps a postcondition α over final states, *i.e.*, a fuzzy predicate, into a precondition $\llbracket S \rrbracket(\alpha)$, a fuzzy predicate over initial states. For a given initial state x , the value $\llbracket S \rrbracket(\alpha)(x)$ is the possibility degree of that contract statement S establishes the postcondition α from the x . (2) We study the basis of contract languages such as the duality between angel and demon and the Normal Form Theorem that describes which contract statements are represented as the composition of the angelic and demonic updates.

In our investigation, the residuated lattice theory, which plays an extremely important role in modern fuzzy logic theory, is a foundation [13]. In a residuated lattice $\mathcal{L} = \langle L, \vee, \wedge, \otimes, \rightarrow, \mathbf{0}, \mathbf{1} \rangle$, $\vee, \wedge, \otimes, \rightarrow$ are used to model the fuzzy connectives: disjunction, conjunction, production and implication, respectively, as well as strong negation $\neg a = 1 - a$ to model fuzzy **not**.

The organization is as follows. In Section 2, we introduce the contract language in detail. We give an example to explain the contract language and explain why we need to deal with contract language in the framework of fuzziness. In Section 3, we recall some basic notions of residuated lattice. In Section 4, we recall the notion of fuzzy relations and introduce ones of fuzzy predicates and fuzzy predicate transformers. A decomposition of fuzzy predicate is given. In Section 5, we introduce two kinds of basic predicate transformers through updates: the angelic and demonic updates. The angelic update is defined via \otimes and the demonic update is defined via \rightarrow . Moreover, we analysis why it is better to use Wang's adjoint pair in defining the angelic and demonic update. In Section 6, we set up the fuzzy semantics of contract statements based on the fuzzy predicate transformers. An example is also given in this section. In Section 7, we introduce the duality and give the Duality Theorem which shows the duality between the angel and demon. In Section 8, we consider the monotonicity of fuzzy predicate transformers. We introduce the notion of strongly monotone fuzzy predicate transformers, and prove the Normal Form Theorem which shows that strongly monotone fuzzy predicate transformers can be represented as the sequential composition of the angelic and demonic updates. The followed is the section of conclusion and future work.

2. Contract Language

Back and von Wright introduce a contract language describing games that are played between two opponents, called the angel and the demon, in their book titled as “Refinement Calculus: A Systematic Introduction” (Springer-Verlag, New York, 1998) [7]. The syntax of a contract statement S is given by (in BNF-form)

$$S ::= \{P\} \mid [P] \mid S_1; S_2 \mid (\prod_{i \in I} S_i) \mid (\sqcup_{i \in I} S_i).$$

Here P is a relation term and I a set of higher-order logic which is simply thought of as an index set. We assume that there is a rule that associates a contract statement S_i with each index $i \in I$ in a meet $\prod_{i \in I} S_i$ (and similarly for a join).

We will interpret contract statements in terms of a game that is played between two opponents, called the angel and the demon. The game semantics describes how a contract statement S encodes the rules of a game. For a play (or execution) of the game (statement S , we also give the initial position (initial statement) $x \in X$ of the game and a goal (a postcondition) $q \subseteq Y$ that describes the set of final states that are winning positions.

For a given postcondition q , the angel tries to reach a final state that satisfies q starting from the initial state x . The demon tries to prevent this. Both follow the rules of the game, as described by the statement S . The statement determines the turns of the game, as well as the moves, where each move either leads to a new state or forces the player that is to make the move to quit (and hence lose the game). Each move is carried out by either the angel or the demon.

There are two basic moves, described by the demonic and the angelic update statements. The angelic update statement $\{P\}$ is executed by the angel in state x by choosing some state y such that $P(x, y)$ holds, *i.e.*, $P(x, y) = 1$. If no such state exists, then the angel can not carry out this move and quits (loses). The demonic update statement $[P]$ is similar, except that it is carried by the demon. It chooses some state y such that $P(x, y)$ holds. If no such state exists, then the demon can not carry out the move and quits.

The sequential composition $S_1; S_2$ is executed in initial state x by the first playing the game S_1 in initial state x . If this game leads to a final state y , then the game S_2 is played from this state. The final state of this game is the final state of the whole game $S_1; S_2$.

The demonic $\prod_{i \in I} S_i$ is executed in initial state x by the demon choosing some game $S_i, i \in I$, to be played in initial state x . If $I = \emptyset$, then there is no game that can be played in initial state x , and the demon has to quit. The angelic choice $\sqcup_{i \in I} S_i$ is executed similarly, but by the angel.

In the remaining of this section, we will give an example describing contract language and explain why we need to deal with contract language in the framework of fuzziness.

An example in [7], two opponents are components of a computer system, one could be a client, and the other could be server. The client issues requests to the server, who carries out the requests according to the specification that has been given for the possible requests. The server procedures that are invoked by the client’s requests have conditions associated with them that restrict the input parameters and the global system state. These conditions are assertions for the client, who has to satisfy them when calling a server procedure. The same conditions are assumptions for the server, who may assume that they hold when starting to comply with a request. The server has to achieve some final state (possibly returning some output values) when the assumptions are satisfied. This constitutes the contract between the client and the server.

There is an interesting form of duality between the client and the server. When we are programming the client, then any internal choices that the server can make increase our uncertainty about what can happen when a request is invoked. To achieve some desired effect, we have to guard ourselves against any possible choice that the server can make. Thus, the client is the angel and the server is the demon. On the other hand, when we are programming the server, then any choices that the client makes, *e.g.*, in choosing values for the input parameters, increases our uncertainty about the initial state in which the request is made, and all possibilities need to be considered when complying with the request. Now, the server is the angel and the client is the demon.

In order to formalize this contract, R.-J. Back and J. von Wright introduce the notion of angelic and demonic update. Let P be a relation from X to Y , $q \subseteq Y$ a predicate and $x \in X$. Then the angelic and demonic update are defined respectively,

$$\{P\}(q)(x) = (\exists y \in Y. P(x, y) \wedge q(y)) = \sup \{P(x, y) \wedge q(y) \mid y \in Y\}$$

and

$$[P](q)(x) = (\forall y \in Y. P(x, y) \rightarrow q(y)) = \inf \{P(x, y) \rightarrow q(y) \mid y \in Y\}$$

Example. Let $X = \{x_1, x_2, x_3, x_4, x_5\}$ be the set of clients and $Y = \{y_1, y_2, y_3, y_4, y_5\}$ the set of servers. The relation P is defined as follows:

$$P(x, y) = \begin{cases} 1 & \text{if the client } x \text{ connects the server } y, \\ 0 & \text{otherwise.} \end{cases}$$

Assume P is given in **Table 1**.

Let q is a predicate which means the servers that can be free to download the film “Gone with the wind”. We assume $p = \{y_1, y_2, y_5\}$. In fact, p can be seen as a Boolean predicate

$$p(y) = \begin{cases} 1 & \text{if } y = y_1, y_2, y_5, \\ 0 & \text{otherwise.} \end{cases}$$

In the sense of this, we call the set to predicate. It is easy to get $\{P\}(q) = \{x_1, x_2, x_3, x_5\}$ and $[P](q) = \{x_3\}$. Hence $\{P\}$ and $[P]$ transform the postcondition (goal) p to some precondition, *i.e.*, they are predicate transformers. $\{P\}(q)(x_1) = 1$ means that the client x_1 connects some server y and can be free to download the film “Gone with the wind” from y by adopting angelic approach. At this time, we call the initial state x_1 can establish the postcondition p by the angelic approach. Thus, initial state x_2 can not establish the postcondition p by the demonic approach.

We can see that the underlying logic of this example is two-valued since starting from an initial state, the game reaches a final state or not and final state satisfies given condition (predicate) or not.

However, we can usually say how many possibilities a client connects a server since the tolerated connection time which is caused by the network; how many possibilities the servers can be free to download the film “Gone with the wind” since the properties of servers; thus how many possibilities a client can establish the postcondition (*i.e.* how many possibilities this client can be free to download this film). That is, it is reasonable to generalize the classical relation into fuzzy relation and the classical predicate into fuzzy predicate. Thus, we generalize the underlying two-valued logic of contract language into fuzzy logic.

3. Residuated Lattice Theory

We note, in the classical two-valued logic, \wedge and \rightarrow in the angelic and demonic update of Back and von Wright, are an adjoint pair. This also motivates us to consider the angelic and demonic update in the framework of residuated lattice. Residuated lattices, introduced by Dilworth and Ward in [13], are a common structure among algebras associated with logical systems. As an important and ideal structure, residuated lattices play an extremely important role in modern fuzzy logic theory.

Definition 3.1 [14] *A residuated lattice on L is an algebra*

$$\mathcal{L} = \langle L, \vee, \wedge, \otimes, \rightarrow, \mathbf{0}, \mathbf{1} \rangle$$

with four binary operations and two constants such that

Table 1. Relation between clients and servers.

$x \setminus y$	y_1	y_2	y_3	y_4	y_5
x_1	1	0	1	0	1
x_2	0	1	0	1	0
x_3	1	1	0	0	1
x_4	0	0	1	1	0
x_5	1	0	0	1	0

- $\mathcal{L} = \langle L, \vee, \wedge, \otimes, \rightarrow, \mathbf{0}, \mathbf{1} \rangle$ is a lattice with the largest element $\mathbf{1}$ and the least element $\mathbf{0}$ w.r.t. the lattice ordering \leq ,
- $\mathcal{L} = \langle L, \otimes, \mathbf{1} \rangle$ is a commutative semigroup with the unit element $\mathbf{1}$, i.e. \otimes is commutative, associative, and $\mathbf{1} \otimes x = x$ for all $x \in L$,
- \otimes and \rightarrow form an adjoint pair, i.e.

$$z \leq x \rightarrow y \text{ iff } x \otimes z \leq y \text{ for all } x, y, z \in L.$$

Given a residuated lattice $\mathcal{L} = \langle L, \vee, \wedge, \otimes, \rightarrow, \mathbf{0}, \mathbf{1} \rangle$, let us define a unary operator \neg , referred to as the precomplement operator, by $\neg a = a \rightarrow \mathbf{0}$ [15].

In what follows, \otimes is sometimes called generalized triangular norm (some literature also call it product) and \rightarrow is called the residuum of \otimes . We denote $x \rightarrow y$ by $I(x, y)$, an implicator I is called left monotonic (resp. right monotonic) iff for every $x \in L$, $I(\cdot, x)$ is decreasing (resp. $I(x, \cdot)$ is increasing). If I is both left monotonic and right monotonic, then it is called hybrid monotonic [15]. Clearly, when implicator I is hybrid monotonic, the corresponding adjoint \otimes is monotonic, i.e., $a \otimes c \leq b \otimes c$ if $a \leq b$, at this time, \otimes is called triangular norm.

Definition 3.2 [15] *Let $\mathcal{L} = \langle L, \vee, \wedge, \otimes, \rightarrow, \mathbf{0}, \mathbf{1} \rangle$ be a residuated lattice, \neg be defined as above. If $\neg\neg x = x$ holds for all $x \in L$, then $\mathcal{L} = \langle L, \vee, \wedge, \otimes, \rightarrow, \mathbf{0}, \mathbf{1} \rangle$ is called a regular residuated lattice.*

Example. Taking $L = [0, 1]$, for any $a, b \in [0, 1]$, the implication operator \rightarrow and the generalized triangular norm \otimes are defined as follows:

- Łukasiewicz's adjoint pair:

$$\begin{aligned} a \rightarrow b &= \min(1 - a + b, 1), \\ a \otimes b &= \max(a + b - 1, 0). \end{aligned}$$

- Gödel's adjoint pair:

$$\begin{aligned} a \rightarrow b &= \begin{cases} 1 & \text{if } a \leq b \\ b & \text{otherwise,} \end{cases} \\ a \otimes b &= \min(a, b). \end{aligned}$$

- Goguen's adjoint pair:

$$\begin{aligned} a \rightarrow b &= \begin{cases} 1 & \text{if } a = 0 \\ \min\left(\frac{b}{a}, 1\right) & \text{otherwise,} \end{cases} \\ a \otimes b &= a \times b. \end{aligned}$$

- Wang's adjoint pair:

$$\begin{aligned} a \rightarrow b &= \begin{cases} 1 & \text{if } a \leq b \\ \max(1 - a, b) & \text{otherwise,} \end{cases} \\ a \otimes b &= \begin{cases} \min(a, b) & \text{if } a + b > 1 \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

For the above four situations, one can verify that $\mathcal{L} = \langle [0, 1], \vee, \wedge, \otimes, \rightarrow, \mathbf{0}, \mathbf{1} \rangle$ are residuated lattices. However, residuated lattices based on Gödel and Goguen implication operators and the corresponding adjoint are not regular since $\neg\neg a = 1 \neq a$ for any $a \in (0, 1)$, the others are regular.

Remark. Ying in [8] investigates the probabilistic game semantics of contracts in Section 8. He uses the probabilistic logic based on Goguen implicator and the corresponding adjoint. Hence, Ying can not get the duality theorem in Section 8(Theorem 8.4).

The following properties are useful for our discussion. Their proofs are straightforward and can be found in, e.g., [16] [17].

Proposition 3.3 Let $\mathcal{L} = \langle L, \vee, \wedge, \otimes, \rightarrow, \mathbf{0}, \mathbf{1} \rangle$ be a regular residuated lattice. Then for any $a, b, c, b_i (i \in I) \in L$,

- (1) $a \rightarrow b = \mathbf{1}$ if and only if $a \leq b$;
- (2) $a \rightarrow b = \neg b \rightarrow \neg a$;
- (3) $a \rightarrow (b \rightarrow c) = b \rightarrow (a \rightarrow c)$;
- (4) $a \rightarrow \inf_{i \in I} b_i = \inf_{i \in I} (a \rightarrow b_i)$ and $(\sup_{i \in I} b_i) \rightarrow a = \inf_{i \in I} (b_i \rightarrow a)$;
- (5) $a \otimes b = \neg(a \rightarrow \neg b)$;
- (6) $(a \otimes b) \rightarrow c = a \rightarrow (b \rightarrow c)$;
- (7) $a \otimes (a \rightarrow b) \leq b$ and $a \rightarrow (a \otimes b) \geq b$;
- (8) $a \otimes (\sup_{i \in I} b_i) = \sup_{i \in I} a \otimes b_i$;
- (9) $\mathbf{1} \rightarrow a = a$. □

Note that, in this paper, we use a **regular residuated lattice** with the hybrid monotonic implicator.

4. Fuzzy Relation and Predicate Transformers

This section will recall three basic notions of fuzzy predicates, fuzzy relations and fuzzy predicate transformers, on which we define some operations.

4.1. Fuzzy Predicates

This paper focuses on the establishment of fuzzy semantics of contract statements based on the fuzzy predicate transformers. Now we will recall these notions of fuzzy predicate and fuzzy predicate transformers.

Definition 4.1 A fuzzy predicate on a state space X is any mapping from X to the unit interval $[0,1]$. We denote the set of fuzzy predicates on X by $\mathcal{F}(X)$. The partial order \sqsubseteq on $\mathcal{F}(X)$ is defined pointwise: For $\alpha, \beta \in \mathcal{F}(X)$,

$$\alpha \sqsubseteq \beta \text{ if and only if } \forall x \in X, \alpha(x) \leq \beta(x).$$

Remark. Indeed, fuzzy predicate on X in this paper is in fact fuzzy set on X , we call it fuzzy predicate in order to coincide with the notion in the computer science. On the other hand, usually the notion fuzzy predicate is different from that of fuzzy set. For example, in [11] [18] fuzzy predicate over a dcpo X , a directed complete poset, is Scott continuous function from this dcpo to the unit interval $[0,1]$ where Scott-continuous function f refers that f preserves the order (i.e., $a \leq b \Rightarrow f(a) \leq f(b)$) and suprema of directed sets (i.e., $f(\sup d_i) = \sup f(d_i)$ for every directed family $(d_i)_i$ in X) (see, Definition 3.1, [18]). When we only consider discrete state spaces X (i.e., a dcpo such that $x \leq y$ if and only if $x = y$ for all $x, y \in X$), fuzzy predicate is just fuzzy set.

The join $\sqcup_{i \in I} \alpha_i$ and meet $\sqcap_{i \in I} \alpha_i$ of a family $\{\alpha_i \mid i \in I\}$ of fuzzy predicates are given by

$$\begin{aligned} (\sqcup_{i \in I} \alpha_i)(x) &= \sup_{i \in I} \alpha_i(x) \\ (\sqcap_{i \in I} \alpha_i)(x) &= \inf_{i \in I} \alpha_i(x). \end{aligned}$$

The fuzzy predicate space $\mathcal{F}(X)$ is a completely distributive lattice. The bottom element is *false*, where $false(x) = 0$, and its top element is *true*, where $true(x) = 1$ for any $x \in X$.

For any $r \in [0,1]$, we define the constant fuzzy predicate \mathbf{r} as the constant function \mathbf{r} from X to $[0,1]$, i.e., $\mathbf{r}(x) = r$, for all $x \in X$. For any $x \in X$, we define a point fuzzy predicate \bar{x} as $\bar{x}(x') = 1$ if $x' = x$ and 0 otherwise.

In addition to the greatest lower bound and the least upper bound, we introduce the logical connectives: negation \neg , the product \otimes and the corresponding adjoint \rightarrow on fuzzy predicates α, β pointwise as follows:

For all $\alpha, \beta \in \mathcal{F}(X)$ and $x \in X$,

$$\begin{aligned}(\neg\alpha)(x) &= 1 - \alpha(x) \\ (\alpha \otimes \beta)(x) &= \alpha(x) \otimes \beta(x) \\ (\alpha \rightarrow \beta)(x) &= \alpha(x) \rightarrow \beta(x)\end{aligned}$$

In particular, $(\mathbf{r} \otimes \alpha)(x) = r \otimes \alpha(x)$, where $r \in [0, 1]$.

It is easily verified that $\langle \mathcal{F}(X), \sqcup, \sqcap, \otimes, \rightarrow, \mathbf{false}, \mathbf{true} \rangle$ is a regular residuated lattice.

Proposition 4.2 (Decomposition of fuzzy predicates) For any $\alpha \in \mathcal{F}(X)$, we have

$$\alpha = \sqcup_{x \in X} \alpha \otimes \bar{x}$$

and

$$\alpha = \sqcap_{x \in X} (\bar{x} \rightarrow \alpha)$$

where (\otimes, \rightarrow) is an adjoint pair.

Proof. We have $(\sqcup_{x \in X} \alpha \otimes \bar{x})(y) = \max(\sup_{x \neq y} \alpha(y) \otimes \bar{x}(y), \alpha(y) \otimes \bar{y}(y)) = \max(0, \alpha(y) \otimes 1) = \alpha(y)$ by the definition of residuated lattice for every $y \in X$.

Since $\alpha \otimes \bar{x} = \neg(\alpha \rightarrow \neg\bar{x})$, we have that $\neg\alpha = \sqcap_{x \in X} (\bar{x} \rightarrow \neg\alpha)$ which proves the second equality. \square

4.2. Fuzzy Predicates Transformers

The notion of fuzzy predicate transformers comes from [18], the authors use it to set up the logical semantics of possibility computation.

Definition 4.3 (1) A fuzzy predicate transformer (FPT, for short) from the state space X to Y is any mapping from $\mathcal{F}(Y)$ to $\mathcal{F}(X)$, denoted by $t: X \mapsto_{\mathcal{F}} Y$.

(2) The refinement order \sqsubseteq between fuzzy predicate transformers is pointwise defined, *i.e.*, for two fuzzy predicate transformers t and t' , $t \sqsubseteq t'$ if and only if $t(\alpha) \sqsubseteq t'(\alpha)$ for any fuzzy predicate $\alpha \in \mathcal{F}(Y)$.

(3) A fuzzy predicate transformer $t: X \mapsto_{\mathcal{F}} Y$ is monotonic if for any $\alpha, \beta \in \mathcal{F}(Y)$, $\alpha \sqsubseteq \beta$ implies that $t(\alpha) \sqsubseteq t(\beta)$ holds.

For any $r \in [0, 1]$, we define a constant FPT $\mathbf{r}_T: X \mapsto_{\mathcal{F}} Y$ by setting $\mathbf{r}_T(\alpha) = r$ for any fuzzy predicate $\alpha \in \mathcal{F}(Y)$. And we denote the identity FPT by *skip* over state space X , defined as $skip(\alpha) = \alpha$ for any fuzzy predicate $\alpha \in \mathcal{F}(X)$.

All operations on fuzzy predicates may be pointwise extended to fuzzy predicate transformers:

$$\begin{aligned}(\neg t)(\alpha) &= \neg t(\alpha) \\ (t \otimes t')(\alpha) &= t(\alpha) \otimes t'(\alpha) \\ (t \rightarrow t')(\alpha) &= t(\alpha) \rightarrow t'(\alpha) \\ (\sqcup_{i \in I} t_i)(\alpha) &= \sqcup_{i \in I} t_i(\alpha) \\ (\sqcap_{i \in I} t_i)(\alpha) &= \sqcap_{i \in I} t_i(\alpha)\end{aligned}$$

in particular,

$$(\mathbf{r}_T \otimes t)(\alpha) = \mathbf{r} \otimes t(\alpha)$$

for all $t, t', t_i: X \mapsto_{\mathcal{F}} Y$, $\alpha \in \mathcal{F}(Y)$.

Moreover, we define the sequential composition of fuzzy predicate transformers. For $t: X \mapsto_{\mathcal{F}} Y$ and $t': Y \mapsto_{\mathcal{F}} Z$ the sequential composition $t; t': X \mapsto_{\mathcal{F}} Z$ of t and t' is defined by $(t; t')(\alpha) = t(t'(\alpha))$ for all $\alpha \in \mathcal{F}(Z)$.

4.3. Fuzzy Relations

Relations appear in many fields of mathematics and computer science. In classical mathematics these relations

are usually crisp, *i.e.*, two objects are related or they are not. However, many relations in real-word applications are intrinsically fuzzy, *i.e.*, objects can be related to each other to a certain degree [19].

Definition 4.4 For given two state spaces X and Y , a fuzzy relation P from X to Y is a fuzzy set on the product $X \times Y$. We write $X \leftrightarrow_{\mathcal{F}} Y$ for the space of fuzzy relations from X to Y . The order between fuzzy relations is defined as $P \sqsubseteq Q$ if and only if $P(x, y) \leq Q(x, y)$ for all $(x, y) \in X \times Y$ for given two fuzzy relations P and Q .

The join $\sqcup_{i \in I} P_i$ and meet $\sqcap_{i \in I} P_i$ of a family $\{P_i \mid i \in I\}$ of fuzzy relations on $X \times Y$ are given by

$$\begin{aligned} (\sqcup_{i \in I} P_i)(x, y) &= \sup_{i \in I} P_i(x, y) \\ (\sqcap_{i \in I} P_i)(x, y) &= \inf_{i \in I} P_i(x, y). \end{aligned}$$

For the algebraic structure of $X \leftrightarrow_{\mathcal{F}} Y$, we have that $X \leftrightarrow_{\mathcal{F}} Y$ is a completely distributive lattice. Its bottom (the empty relation), denoted by *False*, and top (the complete relation), denoted by *True* are defined by

$$False(x, y) = 0 \quad \text{and} \quad True(x, y) = 1$$

for $x \in X$ and $y \in Y$. The identity function over the state space X is represented by the relation Id_X defined as $Id_X(x, x') = 1$ if $x = x'$ and 0 otherwise. We often omit the subscript X .

Definition 4.5 [12] [20] For fuzzy relations $P: X \leftrightarrow_{\mathcal{F}} Y$ and $Q: Y \leftrightarrow_{\mathcal{F}} Z$, *sup-T* composition where T is a triangular norm: $P \bullet Q: X \leftrightarrow_{\mathcal{F}} Z$ is defined by

$$(P \bullet Q)(x, z) = \sup_{y \in Y} P(x, y) \otimes Q(y, z)$$

for all $x \in X$ and $z \in Z$.

Some basic properties of composition are collected in the next proposition. We omit the straightforward proofs.

Proposition 4.6 Let $P, P_i (i \in I): X \leftrightarrow_{\mathcal{F}} Y$ and $Q, Q_i (i \in I): Y \leftrightarrow_{\mathcal{F}} Z$. Then

- (1) $Id_Y \bullet Q = Q; P \bullet Id_X = P$;
- (2) $False \bullet Q = False, P \bullet False = False$;
- (3) $P \bullet (\sqcup_{i \in I} Q_i) = \sqcup_{i \in I} (P \bullet Q_i)$; and
- (4) $(\sqcup_{i \in I} P_i) \bullet Q = \sqcup_{i \in I} (P_i \bullet Q)$. □

Remark 2. $P \bullet (\sqcap_{i \in I} Q_i) \sqsubseteq \sqcap_{i \in I} (P \bullet Q_i)$ holds, but $\sqcap_{i \in I} (P \bullet Q_i) \sqsubseteq P \bullet (\sqcap_{i \in I} Q_i)$ does not hold in general. The following is an example.

Let $P(x, y) = \frac{1}{3}$ for all $x \in X$ and $y \in Y$ and $Q_n(y, z) = \frac{2}{3} + \frac{1}{3n} (n = 1, 2, \dots)$ for all $y \in Y$ and $z \in Z$.

And let \otimes be the corresponding adjoint of implicator of Wang's. Then

$$\begin{aligned} & (\sqcap_{n \in \mathbb{N}} (P \bullet Q_n))(x, z) \\ &= \inf_{n \in \mathbb{N}} (P \bullet Q_n)(x, z) = \inf_{n \in \mathbb{N}} \sup_{y \in Y} P(x, y) \otimes Q_n(y, z) \\ &= \inf_{n \in \mathbb{N}} \sup_{y \in Y} \left[\frac{1}{3} \otimes \left(\frac{2}{3} + \frac{1}{3n} \right) \right] = \frac{1}{3} \quad \left(\text{since } \frac{1}{3} + \frac{2}{3} + \frac{1}{3n} > 1 \right). \end{aligned}$$

However,

$$\begin{aligned} & (P \bullet \sqcap_{n \in \mathbb{N}} Q_n)(x, z) \\ &= \sup_{y \in Y} \left[P(x, y) \otimes \inf_{n \in \mathbb{N}} Q_n(y, z) \right] = \sup_{y \in Y} \left(\frac{1}{3} \otimes \frac{2}{3} \right) = 0. \end{aligned}$$

This shows that $\sqcap_{i \in I} (P \bullet Q_i) \sqsubseteq P \bullet (\sqcap_{i \in I} Q_i)$ does not hold.

Further, the following proposition is easily gotten.

Proposition 4.7 Let $P_1 : X \leftrightarrow_{\mathcal{F}} Y$, $P_2 : Y \leftrightarrow_{\mathcal{F}} Z$ and $P_3 : Z \leftrightarrow_{\mathcal{F}} L$, then $P_1 \bullet (P_2 \bullet P_3) = (P_1 \bullet P_2) \bullet P_3$. \square

5. Basic Predicate Transformers

This section will introduce two basic fuzzy predicate transformers through an update method applied to fuzzy relations and analysis them through adjoint pairs.

5.1. Angelic and Demonic Update

As documented in [7], for a given Boolean relation P from X to Y and a Boolean predicate $p \subseteq Y$. The angelic update $\{P\}$ and the demonic update $\langle P \rangle$ are formally defined as follows:

$$\begin{aligned} \{P\}(q)(x) &= (\exists y \in Y. P(x, y) \wedge q(y)) = \sup_{y \in Y} P(x, y) \wedge q(y) \\ \langle P \rangle(q)(x) &= (\forall y \in Y. P(x, y) \rightarrow q(y)) = \inf_{y \in Y} (P(x, y) \rightarrow q(y)). \end{aligned}$$

Ying in [8] defines the notions of angelic and demonic updates in the probabilistic case by translating the two logical formulae above into probabilistic logic (see Definition 6, page 334).

Here, we define these updates in the possibility case. Since the existential and universal quantifiers can be interpreted by supremum and infimum, respectively, one arrives at the following definition.

Definition 5.1 Let X and Y be two state spaces and $P : X \leftrightarrow_{\mathcal{F}} Y$. For all $\alpha \in \mathcal{F}(Y)$ and $x \in X$, the angelic update $\{P\} : X \mapsto_{\mathcal{F}} Y$ and the demonic update $\langle P \rangle \mapsto_{\mathcal{F}} Y$ are respectively defined by

$$\begin{aligned} \{P\}(\alpha)(x) &= \sup_{y \in Y} P(x, y) \otimes \alpha(y) \\ \langle P \rangle(\alpha)(x) &= \inf_{y \in Y} (P(x, y) \rightarrow \alpha(y)). \end{aligned}$$

where (\otimes, \rightarrow) is an adjoint pair.

Both angelic and demonic updates define fuzzy predicate transformers from state space X to state space Y . That is, for any fuzzy predicate (postcondition) $\alpha \in \mathcal{F}(Y)$, we get a fuzzy predicate (precondition) $\{P\}(\alpha)$ (resp. $\langle P \rangle(\alpha)$) $\in \mathcal{F}(X)$. We call in initial state x , the angelic update $\{P\}$ establishes the postcondition α with the degree $\{P\}(\alpha)(x)$ if the state x satisfies fuzzy predicate $\{P\}(\alpha)$ with the degree $\{P\}(\alpha)(x)$, which is just meaning of membership degree of the state x belonging to fuzzy set $\{P\}(\alpha)$.

Remark. The angelic and demonic updates are similar to the definition of upper and lower fuzzy rough approximation in [19], but a little difference.

$$\begin{aligned} \overline{FAS}^T(A)(y) &= \sup_{x \in X} P(x, y) \otimes A(x) \\ \underline{FAS}_T(A)(y) &= \inf_{x \in X} (P(x, y) \rightarrow A(x)). \end{aligned}$$

where A is a fuzzy set in X . We can see that if P is fuzzy relation in $X \times Y$, then \overline{FAS}^T and \underline{FAS}_T are functions from $\mathcal{F}(X)$ to $\mathcal{F}(Y)$. However, $\{P\}$ and $\langle P \rangle$ are functions from $\mathcal{F}(Y)$ to $\mathcal{F}(X)$.

In next section, we will establish the fuzzy predicate transformer semantics of contract statements. Here, we discuss some properties of these basic updates.

Proposition 5.2 $\{Id\} = \langle Id \rangle = skip$. \square

Proposition 5.3 Let $P, Q : X \leftrightarrow_{\mathcal{F}} Y$. Then

$P \sqsubseteq Q$ if and only if $\{P\} \sqsubseteq \{Q\}$ if and only if $\langle P \rangle \sqsupseteq \langle Q \rangle$. \square

Proof. We only prove $\{P\} \sqsubseteq \{Q\}$ implies $P \sqsubseteq Q$. Let $\{P\} \sqsubseteq \{Q\}$. Then for any $\alpha \in \mathcal{F}(Y)$ and $x \in X$, $\{P\}(\alpha)(x) \leq \{Q\}(\alpha)(x)$. That is,

$$\sup_{y \in Y} P(x, y) \otimes \alpha(y) \leq \sup_{y \in Y} Q(x, y) \otimes \alpha(y).$$

In particular, we take $\alpha = \overline{y_0}$, $y_0 \in Y$, then we have $P(x, y_0) \leq Q(x, y_0)$. By the arbitrariness of y_0 , we get $P \sqsubseteq Q$. \square

This proposition tells us that the order is preserved by the angelic update and reversed by the demonic update.

The following two propositions present homomorphic properties of the angelic and demonic update operators on fuzzy relations.

Proposition 5.4 *Let X, Y, Z be three state spaces, $P: X \leftrightarrow_{\mathcal{F}} Y$ and $Q: Y \leftrightarrow_{\mathcal{F}} Z$. Then*

$$(1) \ \{\!\{P \bullet Q\}\!\} = \{\!\{P\}\!\}; \{\!\{Q\}\!\},$$

$$(2) \ \langle\!\langle P \bullet Q \rangle\!\rangle = \langle\!\langle P \rangle\!\rangle; \langle\!\langle Q \rangle\!\rangle.$$

Proof. We only prove (2), the proof of (1) being similar. For $\alpha \in \mathcal{F}(Z)$ and $x \in X$, we have:

$$\begin{aligned} & (\langle\!\langle P \rangle\!\rangle; \langle\!\langle Q \rangle\!\rangle)(\alpha)(x) = \langle\!\langle P \rangle\!\rangle(\langle\!\langle Q \rangle\!\rangle(\alpha))(x) \\ & = \inf_{y \in Y} P(x, y) \rightarrow \inf_{z \in Z} (Q(y, z) \rightarrow \alpha(z)) \\ & = \inf_{y \in Y} \inf_{z \in Z} [P(x, y) \rightarrow (Q(y, z) \rightarrow \alpha(z))] \quad (\text{by Proposition 3.3(6)}) \\ & = \inf_{y \in Y} \inf_{z \in Z} [(P(x, y) \otimes Q(y, z)) \rightarrow \alpha(z)] \quad (\text{by Proposition 3.3(4)}) \\ & = \inf_{z \in Z} \inf_{y \in Y} [(P(x, y) \otimes Q(y, z)) \rightarrow \alpha(z)] \\ & = \inf_{z \in Z} [\sup_{y \in Y} (P(x, y) \otimes Q(y, z)) \rightarrow \alpha(z)] \quad (\text{by Proposition 3.3(4)}) \\ & = \inf_{z \in Z} [(P \bullet Q)(x, z) \rightarrow \alpha(z)] \quad (\text{by the definition of } \bullet) \\ & = \langle\!\langle P \bullet Q \rangle\!\rangle \cdot \alpha \cdot x \end{aligned}$$

\square

The above proposition shows that the updates and the composition of fuzzy relations commute, whereas the next proposition indicates that the angelic update preserves unions of fuzzy relations, and the demonic update changes a union of fuzzy relations into a meet.

Proposition 5.5 *Let X, Y be state spaces and $P_i (i \in I): X \leftrightarrow_{\mathcal{F}} Y$. Then*

$$(1) \ \{\!\{\sqcup_{i \in I} P_i\}\!\} = \sqcup_{i \in I} \{\!\{P_i\}\!\};$$

$$(2) \ \langle\!\langle \sqcup_{i \in I} P_i \rangle\!\rangle = \sqcup_{i \in I} \langle\!\langle P_i \rangle\!\rangle.$$

Proof. The straightforward proofs use Definition 5.1 and properties of \otimes and \rightarrow . \square

5.2. Analysis

In this subsection, we will analysis angelic and demonic update through different adjoint pairs. We wish to choose more appropriate those to apply contract language.

Usually, in some game, we always assume ‘‘I’’ am angel and opponent is demon. I always wants to win and prevents opponent to win. For fuzzy situation, I always tries to maximize the possibility of wining and minimize possibility of winning of the opponent. Since, in this paper, we discuss contract language only under regular residuated lattice, we compare Łukasiewicz’s adjoint pair with Wang’s adjoint pair for the sake of familiarity.

For a given postcondition $\alpha \in \mathcal{F}(Y)$ and a state $x \in X$, the angelic update $\{\!\{P\}\!\}$ in state x establishes the postcondition α with the degree of $\{\!\{P\}\!\}(\alpha)(x) = \sup_{y \in Y} P(x, y) \otimes \alpha(y)$. By the definitions of \otimes in Łukasiewicz’s adjoint pair and Wang’s adjoint pair, this degree is equal to

$$\sup_{\{y \in Y | P(x, y) + \alpha(y) > 1\}} P(x, y) \wedge \alpha(y)$$

and

$$\sup_{\{y \in Y \mid P(x, y) + \alpha(y) > 1\}} (P(x, y) + \alpha(y) - 1),$$

respectively.

It is easy to verify that for any $y \in \{y \in Y \mid P(x, y) + \alpha(y) > 1\}$, $P(x, y) \wedge \alpha(y) \geq P(x, y) + \alpha(y) - 1$. Hence, $\langle \{P\} \rangle_w(\alpha)(x) \geq \langle \{P\} \rangle_L(\alpha)(x)$, where $\langle \{P\} \rangle_w$ and $\langle \{P\} \rangle_L$ refer to, in the definition of angelic update, the product \otimes taken is in Wang's adjoint pair and Łukasiewicz's adjoint pair, respectively. That is, for me, the possibility of winning (*i.e.* satisfying α) when using \otimes in Wang's adjoint pair is greater than or equal to that when using \otimes in Łukasiewicz's adjoint pair for initial state x .

On the other hand, the demonic update $[P]$ in state x establishes the postcondition α with degree of $\langle [P] \rangle(\alpha)(x) = \inf_{y \in Y} (P(x, y) \rightarrow \alpha(y))$. By the definitions of \rightarrow in Łukasiewicz's adjoint pair and Wang's adjoint pair, this degree is equal to

$$\inf_{\{y \in Y \mid P(x, y) > \alpha(y)\}} ((1 - P(x, y)) \vee \alpha(y))$$

and

$$\inf_{\{y \in Y \mid P(x, y) > \alpha(y)\}} (1 - P(x, y) + \alpha(y)),$$

respectively.

Clearly, for any $y \in \{y \in Y \mid P(x, y) > \alpha(y)\}$, $(1 - P(x, y)) \vee \alpha(y) \leq 1 - P(x, y) + \alpha(y)$. Hence, $\langle [P] \rangle_w(\alpha)(x) \leq \langle [P] \rangle_L(\alpha)(x)$, where $\langle [P] \rangle_w$ and $\langle [P] \rangle_L$ refer to, in the definition of demonic update, the implicator \rightarrow taken is in Wang's adjoint pair and Łukasiewicz's adjoint pair, respectively. That is, for opponent, the possibility of winning (*i.e.* satisfying α) when using \rightarrow in Wang's adjoint pair is smaller than or equal to that when using \rightarrow in Łukasiewicz's adjoint pair for initial state x .

Hence, considering the previous two factors, it seems more appropriate to use Wang's adjoint pair in contract language.

6. Fuzzy Semantics of Contract Language

Based on the previous discussion, we propose the fuzzy predicate transformer semantics of the contract language. The syntax of contract statements is given by

$$S ::= \{P\} \mid [P] \mid S_1; S_2 \mid (\prod_{i \in I} S_i) \mid (\sqcup_{i \in I} S_i).$$

Here P is a fuzzy relation term and I a set of higher-order logic which is simply thought of as an index set.

Definition 6.1 The fuzzy predicate transformer semantics $\llbracket S \rrbracket$ of a contract statement S from X to Y is defined to be a fuzzy predicate transformer which is inductively defined as follows:

$$\begin{aligned} \llbracket \{P\} \rrbracket &= \langle \{P\} \rangle \\ \llbracket [P] \rrbracket &= \langle [P] \rangle \\ \llbracket S_1; S_2 \rrbracket &= \llbracket S_1 \rrbracket; \llbracket S_2 \rrbracket \\ \llbracket \prod_{i \in I} S_i \rrbracket &= \prod_{i \in I} \llbracket S_i \rrbracket \\ \llbracket \sqcup_{i \in I} S_i \rrbracket &= \sqcup_{i \in I} \llbracket S_i \rrbracket. \end{aligned}$$

We are supposed to give a contract statement S which is described by a fuzzy relation P from state space X to state space Y . These semantics of the angelic update $\{P\}$ and the demonic update $[P]$ are respectively defined as the angelic update $\langle \{P\} \rangle$ and the demonic update $\langle [P] \rangle$. Both map postconditions to preconditions.

Now, we continue the example in section three. Fuzzy relations from X to Y are given in [Table 2](#).

Table 2. The fuzzy relation between clients and servers.

<i>R1</i>						
<i>x</i> \ <i>y</i>	<i>y</i> ₁	<i>y</i> ₂	<i>y</i> ₃	<i>y</i> ₄	<i>y</i> ₅	
<i>x</i> ₁	1	0	0	0	0	
<i>x</i> ₂	$\frac{1}{2}$	1	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	
<i>x</i> ₃	$\frac{1}{4}$	$\frac{1}{2}$	1	$\frac{1}{2}$	$\frac{1}{4}$	
<i>x</i> ₄	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$	1	$\frac{1}{2}$	
<i>x</i> ₅	0	0	0	0	1	

<i>R2</i>						
<i>x</i> \ <i>y</i>	<i>y</i> ₁	<i>y</i> ₂	<i>y</i> ₃	<i>y</i> ₄	<i>y</i> ₅	
<i>x</i> ₁	1	0	0	0	0	
<i>x</i> ₂	$\frac{3}{4}$	1	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	
<i>x</i> ₃	$\frac{1}{3}$	$\frac{2}{3}$	1	$\frac{1}{3}$	$\frac{1}{3}$	
<i>x</i> ₄	$\frac{1}{4}$	$\frac{1}{3}$	$\frac{2}{3}$	1	$\frac{5}{6}$	
<i>x</i> ₅	0	0	0	0	1	

A postcondition α means the possibilities that the servers can be free to download the film “Gone with the wind”, which can be given as a fuzzy predicate as follows.

$$\alpha = \frac{0.2}{y_1} + \frac{0.4}{y_2} + \frac{0.6}{y_3} + \frac{0.8}{y_4} + \frac{1}{y_5}.$$

$\alpha(y_1) = 0.2$ means the possibility 0.2 that the server y_1 can be free to download the film ‘Gone with the wind’.

If we take \rightarrow and \otimes in the demonic and angelic update to Wang implicator and its adjoint, then through computing, we can get

$$\{|R1|\}(\alpha) = \frac{0.2}{x_1} + \frac{0.5}{x_2} + \frac{0.6}{x_3} + \frac{0.8}{x_4} + \frac{1}{x_5}$$

and

$$\langle\langle R1 \rangle\rangle(\alpha) = \frac{0.2}{x_1} + \frac{0.4}{x_2} + \frac{0.5}{x_3} + \frac{0.75}{x_4} + \frac{1}{x_5}$$

$$\{|R2|\}(\alpha) = \frac{0.2}{x_1} + \frac{0.4}{x_2} + \frac{0.6}{x_3} + \frac{0.83}{x_4} + \frac{1}{x_5}$$

and

$$\langle\langle R2 \rangle\rangle(\alpha) = \frac{0.2}{x_1} + \frac{0.25}{x_2} + \frac{0.33}{x_3} + \frac{0.6}{x_4} + \frac{1}{x_5}.$$

We can see $\langle\{R1\}\rangle(\alpha), \langle\{R1\}\rangle(\alpha), \langle\{R2\}\rangle(\alpha), \langle\{R2\}\rangle(\alpha)$ are all fuzzy predicates on the clients set X , which means the possibilities that the clients download the film ‘Gone with the wind’ by the angelic and demonic approach, respectively. For example, $\langle\{R1\}\rangle(\alpha)(x_4)=0.8$ means the possibility 0.8 that the client x_4 downloads the film “Gone with the wind” by the angelic approach.

Suppose the contract statement $S1 = [R1];(\{R2\} \sqcup [R1])$. This statement describes these rules: firstly taking the demon to $R1$ and then taking the angel to $R2$ or the demon to $R1$. Now we get that

$$\llbracket S1 \rrbracket(\alpha) = \frac{0.2}{1} + \frac{0.5}{2} + \frac{0.5}{3} + \frac{0.75}{4} + \frac{1}{5}.$$

If we have the contract statement $S2 = (\{R2\} \sqcup [R1]);\{R2\}$, then

$$\llbracket S2 \rrbracket(\alpha) = \frac{0.2}{1} + \frac{0.4}{2} + \frac{0.6}{3} + \frac{0.83}{4} + \frac{1}{5}.$$

From these two formulae above, we can get that the contract statement $S1$ will establish the postcondition α (i.e. downloads the film “Gone with the wind”) from the initial state x_4 with the possibility degree of 0.75. However, $S2$ has this degree of 0.83.

7. The Duality

The angelic and demonic updates are basic strategies of contract language. The angel always wants to win, however, the demon always tries to prevent this winning. Of course, the angel also prevents the demon to win. Therefore, the angel and demon are a pair of opponents which present a duality between them.

In this section we introduce a duality for fuzzy predicate transformers through negation to confirm this duality. The specification language of Back and von Wright [21] was based on the fundamental dualities between demonic and angelic nondeterminism and between nontermination and miracles. Dijkstra and Scholten [5] introduced a notion of converse predicates that is closely related to duals.

Definition 7.1 The dual $t^\circ : X \mapsto_{\mathcal{F}} Y$ of a fuzzy predicate transformer $t : X \mapsto_{\mathcal{F}} Y$ is defined by $t^\circ(\alpha) = \neg t(\neg\alpha)$, where $\alpha \in \mathcal{F}(Y)$.

As $\neg\neg\alpha = \alpha$, it follows that $(t^\circ)^\circ = t$ and that $t_1 \sqsubseteq t_2$ implies $t_2^\circ \sqsubseteq t_1^\circ$. If t is monotone, then so is t° .

Definition 7.2 Let $t : X \mapsto_{\mathcal{F}} Y$. Then

- (1) t has the scaling property, if $t(\mathbf{r} \otimes \alpha) = \mathbf{r} \otimes t(\alpha)$ for all $r \in [0,1]$ and $\alpha \in \mathcal{F}(Y)$.
- (2) t has the implication property, if $t(\mathbf{r} \rightarrow \alpha) = \mathbf{r} \rightarrow t(\alpha)$ for all $r \in [0,1]$ and $\alpha \in \mathcal{F}(Y)$.

The following proposition shows that the scaling property and the implication property are dual in the sense that an FPT t has the scaling property if and only if t° has the implication property.

Proposition 7.3 Let t be an FPT from X to Y . Then, t has the scaling property if and only if its dual t° has the implication property, where (\otimes, \rightarrow) is an adjoint pair.

Proof. Assume that t has the scaling property, i.e., $t(\mathbf{r} \otimes \alpha) = \mathbf{r} \otimes t(\alpha)$ for any $r \in [0,1]$ and $\alpha \in \mathcal{F}(Y)$. Then,

$$\begin{aligned} t^\circ(\mathbf{r} \rightarrow \alpha) &= \neg t(\neg(\mathbf{r} \rightarrow \alpha)) \\ &= \neg t(\mathbf{r} \otimes \neg\alpha) && \text{(by Proposition 3.3(5))} \\ &= \neg(\mathbf{r} \otimes t(\neg\alpha)) && \text{(by assumption)} \\ &= \neg(\mathbf{r} \otimes \neg t^\circ(\alpha)) \\ &= \mathbf{r} \rightarrow t^\circ(\alpha) && \text{(by Proposition 3.3(5) again).} \end{aligned}$$

So, the dual t° has the implication property. Conversely, let t have the implication property. Then

$$t^\circ(\mathbf{r} \otimes \alpha) = \neg t(\neg(\mathbf{r} \otimes \alpha)) = \neg t(\mathbf{r} \rightarrow \neg \alpha) = \neg(\mathbf{r} \rightarrow t(\neg \alpha)) = \neg(\mathbf{r} \rightarrow \neg t^\circ(\alpha)) = \mathbf{r} \otimes t^\circ(\alpha)$$

Therefore, t° has the scaling property. \square

Theorem 7.4 (Duality Theorem) The following dualities hold for fuzzy predicate transformers:

$$\begin{aligned} \{\!|P|\!\}^\circ &= \langle\!|P|\!\rangle \\ \langle\!|P|\!\rangle^\circ &= \{\!|P|\!\} \\ (t_1; t_2)^\circ &= t_1^\circ; t_2^\circ \\ (\bigsqcup_{i \in I} t_i)^\circ &= \prod_{i \in I} t_i^\circ \\ (\prod_{i \in I} t_i)^\circ &= \bigsqcup_{i \in I} t_i^\circ. \end{aligned}$$

Proof. We only prove $\{\!|P|\!\}^\circ = \langle\!|P|\!\rangle$ and $(t_1; t_2)^\circ = t_1^\circ; t_2^\circ$. Let $P: X \leftrightarrow_{\mathcal{F}} Y$. For any $\alpha \in \mathcal{F}(Y)$ and $x \in X$,

$$\begin{aligned} \{\!|P|\!\}^\circ(\alpha)(x) &= \neg \{\!|P|\!\}(\neg \alpha)(x) \\ &= \neg \sup_{y \in Y} P(x, y) \otimes \neg \alpha(y) \\ &= \neg \sup_{y \in Y} \neg(P(x, y) \rightarrow \alpha(y)) \quad (\text{by Proposition 3.3(5)}) \\ &= \inf_{y \in Y} (P(x, y) \rightarrow \alpha(y)) \\ &= \langle\!|P|\!\rangle(\alpha)(x). \end{aligned}$$

Hence $\{\!|P|\!\}^\circ = \langle\!|P|\!\rangle$.

Now, let $t_1: X \mapsto_{\mathcal{F}} Y$, $t_2: Y \mapsto_{\mathcal{F}} Z$ and $\alpha \in \mathcal{F}(Z)$. Then,

$$\begin{aligned} (t_1; t_2)^\circ(\alpha) &= \neg(t_1; t_2)(\neg \alpha) = \neg(t_1(t_2(\neg \alpha))) = \neg(t_1(\neg \neg t_2(\neg \alpha))) \\ &= \neg(t_1(\neg t_2^\circ(\alpha))) = t_1^\circ(t_2^\circ(\alpha)) = (t_1^\circ; t_2^\circ)(\alpha). \end{aligned} \quad \square$$

Proposition 7.5 Let $P: X \leftrightarrow_{\mathcal{F}} Y$, then $\{\!|P|\!\}$ has the scaling property and $\langle\!|P|\!\rangle$ has the implication property.

Proof. Due to the duality, we only need to prove that $\{\!|P|\!\}$ has the scaling property. Let $\beta \in \mathcal{F}(Y)$, $r \in [0, 1]$ and $x \in X$. Then:

$$\begin{aligned} \{\!|P|\!\}(\mathbf{r} \otimes \beta)(x) &= \sup_{y \in Y} P(x, y) \otimes (\mathbf{r} \otimes \beta)(y) \\ &= \sup_{y \in Y} P(x, y) \otimes (\mathbf{r}(y) \otimes \beta(y)) \\ &= \sup_{y \in Y} r \otimes (P(x, y) \otimes \beta(y)) \quad (\text{by properties of } \otimes) \\ &= r \otimes \sup_{y \in Y} (P(x, y) \otimes \beta(y)) \quad (\text{by Proposition 3.3(8)}) \\ &= r \otimes \{\!|P|\!\}(\beta)(x) \\ &= (\mathbf{r} \otimes \{\!|P|\!\}(\beta))(x). \end{aligned} \quad \square$$

8. Monotonicity

In the previous section, we set up the fuzzy predicate transformer semantics of contract statements. It is well-known that not all predicate transformers are interpretations of programming statements. Dijkstra [22] stated that predicate transformers accepted as semantic functions of programming statements are required to satisfy certain healthiness conditions. The basic conditions, that he imposed originally, were strictness, monotonicity, conjunctivity and disjunctivity. With respect to possibility computations, we proposed healthiness conditions in [18]

(Definition 3.4, p. 2666).

R. J. Back and J. von Wright in [7] discovered a very interesting theorem, called the Normal Form Theorem, which says that all monotone predicate transformers can be represented in terms of angelic and demonic updates. In this section, we will introduce the notion of strong monotonicity and establish the Normal Form Theorem for fuzzy predicate transformers. which answers what fuzzy predicate transformers can be represented by angelic and demonic updates of fuzzy relations (see, Theorem 8.6 below).

Definition 8.1 [23] Let X be a state space. For two fuzzy predicates α and β in X , the degree of I -inclusion of α in β is given by

$$S(\alpha, \beta) = \inf_{x \in X} (\alpha(x) \rightarrow \beta(x)).$$

Note that this notion is also called strength of implication by Ying in [8].

For instance, let X be the set of non-negative integers and let $\alpha(0) = 1 - 10^{-1000}$, $\alpha(x) = \frac{1}{2}$ for all $x > 0$, and $\beta(x) = 1 - 10^{-100}$ for all $x \in X$. Then, both α and β are fuzzy predicates on X . We first notice that $\alpha \not\sqsubseteq \beta$ for $\alpha(0) = 1 - 10^{-1000} \not\leq 1 - 10^{-100} = \beta(0)$ although $\alpha(x) \leq \beta(x)$ holds for all $x \neq 0$. But, it is reasonable to say that α implies β with a very high degree $S(\alpha, \beta) = 1 - 10^{-100}$ very close to 1.

Definition 8.2 Let X and Y be two state spaces, and $t: X \mapsto_{\mathcal{F}} Y$. Then t is said to be strongly monotone if and only if for all $\alpha, \beta \in \mathcal{F}(Y)$, we have $S(\alpha, \beta) \leq S(t(\alpha), t(\beta))$.

It is easy to see that strong monotonicity implies (weak) monotonicity, but the converse does not hold.

Proposition 8.3 If FPT t is strongly monotone then so is t° .

Proof. Assume that t is a strongly monotone FPT from X to Y . We need to show that the dual FPT t° is strongly monotone, too. Now, $\forall \alpha, \beta \in \mathcal{F}(Y)$, due to the strong monotonicity of t , we have

$$S(\alpha, \beta) \leq S(t(\alpha), t(\beta)) \text{ and}$$

$$\begin{aligned} S(t^\circ(\alpha), t^\circ(\beta)) &= \inf_{x \in X} (t^\circ(\alpha)(x) \rightarrow t^\circ(\beta)(x)) \\ &= \inf_{x \in X} (\neg t(\neg\alpha)(x) \rightarrow \neg t(\neg\beta)(x)) \\ &= \inf_{x \in X} (t(\neg\beta)(x) \rightarrow t(\neg\alpha)(x)) \quad (\text{by Proposition 3.3(2)}) \\ &= S(t(\neg\beta), t(\neg\alpha)) \\ &\geq S(\neg\beta, \neg\alpha) \quad (\text{since } t \text{ is strongly monotone}) \\ &= S(\alpha, \beta). \quad (\text{by Proposition 3.3(2)}) \end{aligned}$$

Therefore, t° is a strongly monotone FPT. □

We will show that strong monotonicity can be derived from (weak) monotonicity and the scaling property.

Proposition 8.4 Assume that an FPT $t: X \mapsto_{\mathcal{F}} Y$ is monotone. Then, if t has the scaling property or the implication property, then t is strongly monotone.

Proof. We have to show: $S(\alpha, \beta) \leq S(t(\alpha), t(\beta))$. For this, it suffices to show: whenever $r \leq S(\alpha, \beta)$, then $r \leq S(t(\alpha), t(\beta))$. We need to verify it for the case of t being the scaling property. For another case, We can also verify it using the adjoint between \otimes and \rightarrow in the following induction. But, we like to get this verification by using the duality between scaling and implication (Proposition 7.3 and 8.3).

$$\begin{aligned} r &\leq S(\alpha, \beta) \\ &\Leftrightarrow \mathbf{r} \sqsubseteq (\alpha \rightarrow \beta) \quad (\text{by the definition of } S) \\ &\Leftrightarrow \mathbf{r} \otimes \alpha \sqsubseteq \beta \quad (\text{by adjoint condition between } \otimes \text{ and } \rightarrow) \\ &\Rightarrow t(\mathbf{r} \otimes \alpha) \sqsubseteq t(\beta) \quad (\text{by the monotonicity of } t) \quad \square \\ &\Leftrightarrow \mathbf{r} \otimes t(\alpha) \sqsubseteq t(\beta) \quad (\text{as } t \text{ is scaling}) \\ &\Leftrightarrow \mathbf{r} \sqsubseteq (t(\alpha) \rightarrow t(\beta)) \quad (\text{by adjoint condition between } \otimes \text{ and } \rightarrow) \\ &\Leftrightarrow r \leq S(t(\alpha), t(\beta)) \end{aligned}$$

We now establish the normal form theorem for fuzzy predicate transformers. We first give a theorem which says that updates of all basic statements are strongly monotone and that operations for constructing compound statements also preserve strong monotonicity.

Theorem 8.5 *Strong monotonicity is closed under composition, meet, and join of fuzzy predicate transformers.*

Proof. (1) We prove that meet of fuzzy predicate transformers preserves strong monotonicity. Let $t_i (i \in I) : X \mapsto_{\mathcal{F}} Y$ and $\alpha, \beta \in \mathcal{F}(Y)$. Then

$$\begin{aligned} S(\sqcap_{i \in I} t_i(\alpha), \sqcap_{i \in I} t_i(\beta)) &= \inf_{x \in X} (\inf_{i \in I} t_i(\alpha)(x) \rightarrow \inf_{i \in I} t_i(\beta)(x)) \\ &= \inf_{x \in X} \inf_{i \in I} (\inf_{i \in I} t_i(\alpha)(x) \rightarrow t_i(\beta)(x)) \quad (\text{by Proposition 3.3(4)}) \\ &\geq \inf_{x \in X} \inf_{i \in I} (t_i(\alpha)(x) \rightarrow t_i(\beta)(x)) \quad (\text{by hybrid monotonicity of } \rightarrow) \\ &= \inf_{i \in I} \inf_{x \in X} (t_i(\alpha)(x) \rightarrow t_i(\beta)(x)) \\ &= \inf_{i \in I} S(t_i(\alpha), t_i(\beta)) \\ &\geq S(\alpha, \beta). \quad (\text{by the strong monotonicity of } t_i) \end{aligned}$$

Hence, meets preserve strong monotonicity. Similarly, we can prove that joins of fuzzy predicate transformers preserve strong monotonicity.

(2) We prove that composition preserves strong monotonicity. Let $t_1 : X \mapsto_{\mathcal{F}} Y$, $t_2 : Y \mapsto_{\mathcal{F}} Z$ and $\alpha, \beta \in \mathcal{F}(Y)$. Then

$$\begin{aligned} S((t_1; t_2)(\alpha), (t_1; t_2)(\beta)) &= S(t_1(t_2(\alpha)), t_1(t_2(\beta))) \\ &\geq S(t_2(\alpha), t_2(\beta)) \quad (\text{since } t_1 \text{ is strongly monotone}) \\ &\geq S(\alpha, \beta). \quad (\text{since } t_2 \text{ is strongly monotone}) \end{aligned}$$

Hence, composition preserves strong monotonicity. \square

Now we present the Normal Form Theorem for strongly monotone fuzzy predicate transformers.

Theorem 8.6 (Normal Form Theorem) Let X and Y be two state spaces and $t : X \mapsto_{\mathcal{F}} Y$. Then t is strongly monotone if and only if $t = \{P\};[Q]$, for some fuzzy relations P and Q .

Proof. The “if” part. By Proposition 7.5, $\{P\}$ has the scaling property and $\langle P \rangle$ the implication property. As $\{P\}$ and $\langle P \rangle$ are monotone, Proposition 8.4 implies that $\{P\}$ and $\langle P \rangle$ are strongly monotone.

The “only if” part. We define fuzzy relations $P : X \times \mathcal{F}(Y) \rightarrow [0, 1]$ and $Q : \mathcal{F}(Y) \times Y \rightarrow [0, 1]$ as follows:

$$\begin{aligned} P(x, \alpha) &= t(\alpha)(x) \\ Q(\alpha, y) &= \alpha(y) \end{aligned}$$

for all $x \in X, y \in Y$ and $\alpha \in \mathcal{F}(Y)$. Then we have

$$\begin{aligned} (\{P\}; \langle Q \rangle)(\alpha)(x) &= \{P\}(\langle Q \rangle(\alpha))(x) \\ &= \sup_{\beta \in \mathcal{F}(Y)} P(x, \beta) \otimes Q(\alpha)(\beta) \\ &= \sup_{\beta \in \mathcal{F}(Y)} [t(\beta)(x) \otimes \inf_{y \in Y} (Q(\beta, y) \rightarrow \alpha(y))] \\ &= \sup_{\beta \in \mathcal{F}(Y)} [t \cdot \beta \cdot x \otimes \inf_{y \in Y} (\beta(y) \rightarrow \alpha(y))] \\ &\geq t(\alpha)(x) \otimes \inf_{y \in Y} (\alpha(y) \rightarrow \alpha(y)) \quad (\text{since } \alpha \in \mathcal{F}(Y)) \\ &= t(\alpha)(x) \otimes 1 \\ &= t(\alpha)(x). \end{aligned}$$

On the other hand, for any $\beta \in \mathcal{F}(Y)$, since t is strongly monotone,

$$\begin{aligned} \inf_{y \in Y} (\beta(y) \rightarrow \alpha(y)) &= S(\beta, \alpha) \\ &\leq S(t(\beta), t(\alpha)) \\ &= \inf_{x' \in X} (t(\beta)(x') \rightarrow t(\alpha)(x')) \\ &\leq (t(\beta)(x) \rightarrow t(\alpha)(x)) \quad (\text{since } x \in X). \end{aligned}$$

Therefore,

$$\begin{aligned} (\langle P \rangle; \langle Q \rangle)(\alpha)(x) &= \sup_{\beta \in \mathcal{F}Y} [t(\beta)(x) \otimes \inf_{y \in Y} (\beta(y) \rightarrow \alpha(y))] \\ &\leq \sup_{\beta \in \mathcal{F}Y} t(\beta)(x) \otimes (t(\beta)(x) \rightarrow t(\alpha)(x)) \\ &\leq t(\alpha)(x) \quad (\text{by Proposition 3.3(7)}). \end{aligned}$$

We have gotten that $(\langle P \rangle; \langle Q \rangle)(\alpha) = t(\alpha)$. Hence $\langle P \rangle; \langle Q \rangle = t$. □

9. Conclusion and Further Work

We have given the fuzzy predicate transformer semantics of contract language introduced by Back and von Wright in [7]. Adjoint pair of a residuated lattice is used to define the fuzzy logic connectives: implication and production. In terms of these fuzzy operations, we have defined the fuzzy updates of the angelic and demonic statements in contract language. These two updates are the basis of fuzzy predicate transformers in the sense that strongly monotone fuzzy predicate transformers are just the sequential composition of these angelic and demonic updates of two fuzzy relations, which is called as Normal Form Theorem. Together with the standard strong negation, we have set up the duality of fuzzy predicate transformers. The duality describes the duality between the angel and the demon, which says that the angel and demon in games can change their strategies each other. In the future, we need to consider the denotational [1] and operational semantics [2] in the framework of fuzziness. Moreover, those semantics of contract language should be considered by taking into count the conditional structure **if b then S_1 else S_2** and loop structure **while b do S** .

Recently, Wang and Zhou [24] establish the foundation of quantitative logic. Pei [25] investigates the full implication of fuzzy reasoning. Both will promote the research on the quantitative semantics with respect to fuzzy programming language.

Acknowledgements

The first author was supported by the Zhejiang Provincial Natural Science Foundation of China (LY13F020046) and Zhejiang Provincial Education Department Fund of China (Y201223001). The second author acknowledges support from the Natural Science Foundation of China (No. 61370100) and the Shanghai Leading Academic Discipline Project (B412).

References

- [1] Stoy, J.E. (1977) Denotational Semantics: The Scott-Strachey Approach to Programming Language Theory. MIT Press, Cambridge, MA.
- [2] Plotkin, G.D. (1981) An Structural Approach to Operational Semantics. Technical Report DAIMI FN-19, Computer Science Department, Aarhus University.
- [3] Hoare, C.A.R. (1969) An Axiomatic Basis for Computer Programming. *Communications of the ACM*, **12**, 576-580. <http://dx.doi.org/10.1145/363235.363259>
- [4] Van Horebeek, I. and Lewi, J. (1989) Algebraic Specifications in Software Engineering an Introduction. Springer-Verlag, Berlin Heidelberg, New York.
- [5] Dijkstra, E.W. and Scholten, C.S. (1990) Predicate Calculus and Programs Semantics. Springer-Verlag, New York.
- [6] Hoare, C.A.R. (1978) Some Properties of Predicate Transformers. *Journal of the Association for Computing Machinery*, **25**, 461-480. <http://dx.doi.org/10.1145/322077.322088>

- [7] Back, R.-J. and von Wright, J. (1998) Refinement Calculus: A Systematic Introduction. Springer-Verlag, New York. <http://dx.doi.org/10.1007/978-1-4612-1674-2>
- [8] Ying, M.S. (2003) Reasoning about Probabilistic Sequential Programs in a Probabilistic Logic. *Acta Informatica*, **39**, 315-389. <http://dx.doi.org/10.1007/s00236-003-0113-z>
- [9] Morgan, C., McIver, A.K. and Seidel, K. (1996) Probabilistic Predicate Transformers. *ACM Transactions on Programming Languages and Systems*, **18**, 325-353. <http://dx.doi.org/10.1145/229542.229547>
- [10] He, J., Seidel, K. and McIver, A.K. (1997) Probabilistic Models for the Guarded Command Language. *Science of Computer Programming*, **28**, 171-192. [http://dx.doi.org/10.1016/S0167-6423\(96\)00019-6](http://dx.doi.org/10.1016/S0167-6423(96)00019-6)
- [11] Chen, Y.X. and Jung, A. (2004) An Introduction to Fuzzy Predicate Transformers. *The Third International Symposium on Domain Theory*, Xi'an, 10-24 May 2004.
- [12] Zadeh, L.A. (1965) Fuzzy Sets. *Information and Control*, **8**, 338-353. [http://dx.doi.org/10.1016/S0019-9958\(65\)90241-X](http://dx.doi.org/10.1016/S0019-9958(65)90241-X)
- [13] Dilworth, R.P. and Ward, N. (1939) Residuated Lattices. *Transactions of the American Mathematical Society*, **45**, 335-354. <http://dx.doi.org/10.1090/S0002-9947-1939-1501995-3>
- [14] Daňková, M. (2010) Approximation of Extensional Fuzzy Relations over a Residuated Lattice. *Fuzzy Sets and Systems*, **161**, 1973-1991. <http://dx.doi.org/10.1016/j.fss.2010.03.011>
- [15] She, Y.H. and Wang, G.J. (2009) An Axiomatic Approach of Fuzzy Rough Sets Based on Residuated Lattices. *Computers and Mathematics with Applications*, **58**, 189-201. <http://dx.doi.org/10.1016/j.camwa.2009.03.100>
- [16] Wang, G.J. (1999) On the Logic Foundation of Fuzzy Reasoning. *Information Sciences*, **117**, 47-88. [http://dx.doi.org/10.1016/S0020-0255\(98\)10103-2](http://dx.doi.org/10.1016/S0020-0255(98)10103-2)
- [17] Wang, G.J. and Wang, H. (2001) Non-Fuzzy Versions of Fuzzy Reasoning in Classical Logics. *Information Sciences*, **138**, 211-236. [http://dx.doi.org/10.1016/S0020-0255\(01\)00131-1](http://dx.doi.org/10.1016/S0020-0255(01)00131-1)
- [18] Chen, Y.X. and Wu, H.Y. (2008) Domain Semantics of Possibility Computations. *Information Sciences*, **178**, 2661-2679. <http://dx.doi.org/10.1016/j.ins.2008.01.016>
- [19] Nachttegael, M., De Cock, M., Van der Weken, D. and Kerre, E.E. (2002) Fuzzy Relational Images in Computer Science. In: de Swart, H., Ed., *Lecture Notes in Computer Science*, Vol. 2561, Springer, Berlin, 134-151. http://dx.doi.org/10.1007/3-540-36280-0_10
- [20] Bandler, W. and Kohout, L.J. (1980) Fuzzy Relational Products as a tool for Analysis and Synthesis of the Behaviour of Complex Natural and Artificial Systems. In: Wang, S.K. and Chang, P.P., Eds., *Fuzzy Sets: Theory and Application to Policy Analysis and Information Systems*, Plenum Press, New York and London, 341-367. http://dx.doi.org/10.1007/978-1-4684-3848-2_26
- [21] Back, R.-J. and von Wright, J. (1990) Duality in Specification Languages: A Lattice-Theoretical Approach. *Acta Informatica*, **27**, 583-625. <http://dx.doi.org/10.1007/BF00259469>
- [22] Dijkstra, E.W. (1976) A Discipline of Programming. Prentice Hall, Inc., Englewood Cliffs, 1976.
- [23] Bandler, W. and Kohout, L.J. (1980) Fuzzy Power Sets and Fuzzy Implication Operators. *Fuzzy Sets and Systems*, **4**, 13-30. [http://dx.doi.org/10.1016/0165-0114\(80\)90060-3](http://dx.doi.org/10.1016/0165-0114(80)90060-3)
- [24] Wang, G.J. and Zhou, H.J. (2009) Quantitative Logic. *Information Sciences*, **179**, 226-247. <http://dx.doi.org/10.1016/j.ins.2008.09.008>
- [25] Pei, D.W. (2008) Unified Full Implication Algorithms of Fuzzy Reasoning. *Information Sciences*, **178**, 520-530. <http://dx.doi.org/10.1016/j.ins.2007.09.003>