

A connection between Similarity Logic Programming and Gödel Modal Logic

Luciano Blandi¹, Lluís Godo², and Ricardo O. Rodríguez³

¹ DMI-Università di Salerno, Via Ponte Don Melillo, 84084 Fisciano (SA), Italy
lblandi@unisa.it

² IIIA - CSIC, Campus UAB, 08193 Bellaterra, Spain
godo@iia.csic.es

³ Dpto. de Computación, Fac. Ciencias Exactas y Naturales, Universidad de Buenos Aires,
Ciudad de Buenos Aires, Argentina
ricardo@dc.uba.ar

Abstract. In this paper we relate two logical similarity-based approaches to approximate reasoning. One approach extends the framework of (propositional) classical logic programming by introducing a similarity relation in the alphabet of the language that allows for an extended unification procedure. The second approach is a many-valued modal logic approach where $\diamond p$ is understood as *approximately p*. Here, the similarity relations are introduced at the level of the Kripke models where possible worlds can be similar to some extent. We show that the former approach can be expressed inside the latter.

Keywords: Similarity, Logic Programming, Gödel Logic.

1 Introduction and previous works

One of the goals of a variety of approximate reasoning models is to cope with inference patterns more flexible than those of classical reasoning. Among them, the similarity-based reasoning based on the notion of similarity relation [16] which provides a way to manage alternative instances of an entity that can be considered “equal” with a given degree expressed by a value in $[0,1]$.

Two main similarity-based approaches to approximate reasoning are put into relation in this paper. The first one is an approach based on introducing a similarity relation \mathcal{R} (in the sense of a reflexive, symmetric and min-transitive fuzzy relation) in the set of object names in a language of classical propositional Logic Programming. Following [15], [2] and [14], we consider inferences that may be approximated by allowing the antecedent clauses of a rule to match its premises only approximately. In particular, the classical SLD Resolution is modified in order to overcome failure situations in the unification process if the entities involved in the matching have a non-zero similarity degree. Such a procedure allows us to compute numeric values belonging to the interval $[0,1]$, named *approximation degrees*, which provide an approximation measure of the obtained solutions. This framework, which we shall call *Similarity Propositional Logic Programming (SPLP)*, is the propositional version of that one proposed by Sessa in [14] which is based upon a first order language. In [7] we find the first proposal to introduce similarity in the frame of the declarative paradigm of Logic Programming. Logic programs on function-free languages are considered and approximate and imprecise information are represented by introducing a similarity relation between constant and predicate symbols. Two transformation techniques of logic programs are defined. In the underlying logic, the inference rule (Resolution rule) as well as the usual crisp representation of the considered universe are not modified. It allows to avoid both the introduction of weights on the clauses, and the use of fuzzy sets as elements of the language. The semantic equivalence between the two inference processes associated to the two kind of transformed programs has been proved by using an abstract interpretation technique. Moreover, the notion of fuzzy least Herbrand model has been introduced. In [13] the generalization of this approach to the case of programs

with function symbols is provided by introducing the general notion of *structural translation* of languages. In [14] the operational counterpart of this extension is faced by introducing a modified SLD Resolution procedure which allows us to perform these kinds of extended computations exploiting the original logic program, without any preprocessing steps in order to transform the given program. [10] presents an extended PROLOG interpreter, named *SiLog*, which implements this inference procedure. Finally, for completeness sake, we also cite [6] where a first and different (it takes into account substitutions of variable with sets of symbols) generalized unification algorithm based on similarity has been proposed.

The second approach is based in introducing a similarity relation S in the set of interpretations or possible worlds. This kind of approach was started by Ruspini [12] by proposing a similarity-based semantics for fuzzy logic, trying to capture inference patterns like the so-called *generalized modus ponens*. The basic idea of this “similarity approach” is that the degree of truthlikeness of a sentence φ depends on the similarities between the states of affairs allowed by φ and the true state of the world. Intuitively speaking, a statement is truthlike if it is “like the truth” or “similar to the truth” but it does not have to be true or even probable. The idea is to attach to each proposition φ of a given basic language \mathcal{L} a new “fuzzy” proposition $\diamond\varphi$ read as “*approximately- φ* ”. This leads to deal with degrees of truth (how close is φ to truth = how true is *approximately- φ*) but, unlike to most systems of many-valued logic, this notion is not compositional (functional) and it is modelled by a logical modality. In our logic, we rely on a system of modal logic related to similarity-based reasoning on *fuzzy* propositions. Technically, we combine many-valued logic (to model fuzziness) and modal logic (to model similarity). Therefore we propose a modal fuzzy logic with semantics based on Kripke structures where the accessibility relations are fuzzy similarity relations measuring how similar are the possible worlds. This will result on a many-valued modal system, a many-valued counterpart of the classical S5 modal system, with many-valued similarity-based Kripke model semantics. In a previous work [9] a modal logic over the Rational Pavelka logic has been defined. Instead, we use a rational modal logic based upon the many-valued Gödel propositional Logic, named *Rational Gödel similarity-based S5 modal logic* (RGS5 $_{\diamond}$).

The main difference between these approaches is that in SPLP the similarity is defined between symbols in the alphabet of the language, i.e. it is exploited at a syntactic-level, whereas in RGS5 $_{\diamond}$ the similarity is defined in the set of the interpretations, i.e. it is exploited at a semantic-level. Nevertheless, both approaches can be put into relation.

The paper is organized as follows. After this introduction we survey in Section 2 SPLP and RGS5 $_{\diamond}$. The third section is devoted to study in detail the above mentioned correspondences between the two approaches. The last section contains some concluding remarks.

2 Two similarity-based approaches to approximate reasoning

This section briefly outlines those elements of SPLP and RGS5 $_{\diamond}$ which we shall take for granted in what follows, and at the same time explains some of the terminology which we shall use throughout the paper. Some common concepts follow.

Definition 1. A similarity on a domain \mathcal{U} is a fuzzy relation $\mathcal{R} : \mathcal{U} \times \mathcal{U} \rightarrow [0, 1]$ in \mathcal{U} such that the following properties hold

- i) $\mathcal{R}(x, x) = 1 \ \forall x \in \mathcal{U}$ (reflexivity)
- ii) $\mathcal{R}(x, y) = \mathcal{R}(y, x) \ \forall x, y \in \mathcal{U}$ (symmetry)
- iii) $\mathcal{R}(x, z) \geq \mathcal{R}(x, y) \wedge \mathcal{R}(y, z) \ \forall x, y, z \in \mathcal{U}$ (transitivity)

We say that \mathcal{R} is strict if the following implication is also verified

- iv) $\mathcal{R}(x, z) = 1 \ \Rightarrow \ x = z.$

The mathematical notion of Similarity relation is a many valued extension of the equality, indeed equal elements have similarity degree 1 and completely different elements have similarity degree 0, and it is widely exploited in any context where a weakening of the equality constraint is useful.

2.1 Similarity relations on symbols: SPLP

Let $Const$ be a set of propositional constants and let \mathcal{L} be the usual classical propositional language.

We briefly recall that a logic program P on \mathcal{L} is a conjunction of definite clauses of \mathcal{L} , denoted as $q \leftarrow p_1, \dots, p_n$, $n \geq 0$, and a goal is a negative clause, denoted with $\leftarrow q_1, \dots, q_n$, $n \geq 1$, where the symbol “,” that separates the propositional constants has to be interpreted as conjunction, and $p_1, \dots, p_n, q, q_1, \dots, q_n \in Const$. A *SPLP-program* is a pair (P, \mathcal{R}) , where P is a logic program defined on \mathcal{L} and \mathcal{R} is a similarity on $Const$. Given P , the least Herbrand model of P is given by $M_P = \{p \in Const \mid P \models p\}$, where \models denotes classical logical entailment. M_P is equivalent to the corresponding procedural semantics of P , defined by considering the SLD Resolution. In the classical case, a mismatch between two propositional constant names causes a failure of the unification process. Then, it is rather natural to admit a more flexible unification in which the syntactical identity is substituted by a Similarity \mathcal{R} defined on $Const$. The modified version of the SLD Resolution, which we shall call *Similarity-based SLD Resolution*, exploits this simple variation in the unification process. The basic idea of this procedure for first order languages has been outlined in [8]. The following definitions formalize these ideas in the case of propositional languages.

Definition 2. Let $\mathcal{R} : Const \times Const \rightarrow [0, 1]$ be a similarity and $p, q \in Const$ be two propositional constants in a propositional language \mathcal{L} . We define the unification-degree of p and q with respect to \mathcal{R} the value $\mathcal{R}(p, q)$. p and q are λ -unifiable if $\mathcal{R}(p, q) = \lambda$ with $\lambda > 0$, otherwise we say that they are not unifiable.

Definition 3. Given a similarity $\mathcal{R} : Const \times Const \rightarrow [0, 1]$, a program P and a goal G_0 , a similarity-based SLD derivation of $P \cup \{G_0\}$, denoted by

$$G_0 \Rightarrow_{C_1, \alpha_1} G_1 \Rightarrow \dots \Rightarrow_{C_k, \alpha_k} G_k$$

consists of a sequence G_0, G_1, \dots, G_k of negative clauses, together with a sequence C_1, C_2, \dots, C_k of clauses from P and a sequence $\alpha_1, \alpha_2, \dots, \alpha_k$ of values in $[0, 1]$, such that for all $i \in \{1, \dots, k\}$, G_i is a resolvent of G_{i-1} and C_i with unification degree α_i . The approximation degree of the derivation is $\alpha = \inf\{\alpha_1, \dots, \alpha_k\}$. If G_k is the empty clause \perp , for some finite k , the derivation is called a Similarity-based SLD refutation, otherwise it is called failed.

It is easy to see that when the similarity \mathcal{R} is the identity, the previous definition provides the classical notion of SLD refutation. The values α_i can be considered as constraints that allow the success of the unification processes. Then, it is natural to consider the best unification degree that allows us to satisfy all these constraints. In general, an answer can be obtained with different SLD refutations and different approximation degrees, then the maximum α of these values characterizes the best refutations of the goal. In particular, a refutation with approximation-degree 1 provides an exact solution. Let us stress that α belongs to the set $\lambda_1, \lambda_2, \dots$ of the possible similarity values in \mathcal{R} . In the sequel, we assume the Leftmost selection rule whenever Similarity-based SLD Resolution is considered. However, all the presented results can be analogously stated for any selection rule that does not depend on the propositional constant names and on the history of the derivation [1]. Similarity-based SLD Resolution provides a characterization of the fuzzy least Herbrand model $M_{P, \mathcal{R}}$ for (P, \mathcal{R}) defined in [7], as stated by the following result.

Proposition 1. Let a similarity \mathcal{R} and a logic program P on a propositional language \mathcal{L} be given. For any $q \in Const$, $M_{P, \mathcal{R}}(q) = \alpha > 0$ if and only if α is the maximum value in $(0, 1]$ for which there exists a Similarity-based SLD refutation for $P \cup \{\leftarrow q\}$ with approximation degree α .

Intuitively, the degree of membership $M_{P, \mathcal{R}}(q)$ of an atom q is given by the best “tolerance” level $\alpha \in (0, 1]$ which allows us to prove q exploiting the Similarity-based SLD Resolution on $P \cup \{\leftarrow q\}$.

Finally, let us remember the following relation between the classical least Herbrand Model of a program P and the fuzzy generalization of this notion.

Proposition 2. *Let P be a logic program on a propositional language \mathcal{L} with a strict Similarity \mathcal{R} . If we denote the least Herbrand model of P by M_P , then $q \in M_P$ if and only if $M_{P,\mathcal{R}}(q) = 1$.*

2.2 Similarity relations on possible worlds: $RGS5_\diamond$

The starting point in this approach is to assume that a possible world or state of a system may resemble more to some worlds than to another ones, and this basic fact may help us to evaluate to what extent a partial description (a proposition) may be close or similar to some other.

In [3] the authors define a many-valued modal logic over Gödel fuzzy logic by introducing only a possibility modal operator \diamond , where the intended meaning of $\diamond p$ is *approximately p*. Although p may be a classical proposition, $\diamond p$ is considered to be fuzzy since the current state of the world may be more or less close to p . Moreover, since we want to explicitly deal with similarity degrees in the language we will consider as base logic the expansion of Gödel logic with rational truth-constants, called RG in [5].

We consider the language $\mathcal{L}_{G\diamond}$ of Gödel similarity modal logic, built over $Const$ with Gödel connectives $\wedge, \rightarrow, \bar{0}$ and \diamond and truth constants \bar{r} for each $r \in Q \cap [0, 1]$.

Definition 4. *The Rational Gödel similarity-based S5 modal logic $RGS5_\diamond$ is defined over the language $\mathcal{L}_{G\diamond}$ and is the smallest set of formulas containing every instance of the following axiom schemes and closed under the last two inference rules:*

Axioms of Rational Gödel logic:

$$\begin{aligned} &(\varphi \rightarrow \psi) \rightarrow ((\psi \rightarrow \chi) \rightarrow (\varphi \rightarrow \chi)). \\ &\varphi \rightarrow (\psi \rightarrow \varphi). \\ &(\varphi \wedge \psi) \rightarrow (\psi \wedge \varphi). \\ &(\varphi \wedge (\psi \wedge \chi)) \rightarrow ((\psi \wedge \varphi) \wedge \chi). \\ &(\varphi \rightarrow (\psi \rightarrow \chi)) \equiv ((\varphi \wedge \psi) \rightarrow \chi). \\ &(((\varphi \rightarrow \psi) \rightarrow \chi) \rightarrow (((\psi \rightarrow \varphi) \rightarrow \chi) \rightarrow \chi)). \\ &\bar{0} \rightarrow \varphi. \\ &\varphi \rightarrow (\varphi \wedge \varphi). \\ &\neg\varphi \equiv \varphi \rightarrow \bar{0}. \\ &\bar{r} \wedge \bar{s} \equiv \min\{r, s\}. \\ &\bar{r} \rightarrow \bar{s} \equiv r \Rightarrow s. \end{aligned}$$

D $_\diamond$: $\diamond(\varphi \vee \psi) \rightarrow (\diamond\varphi \vee \diamond\psi)$.

Z $_\diamond^+$: $\diamond\neg\neg\varphi \rightarrow \neg\neg\diamond\varphi$.

F $_\diamond$: $\neg\diamond\bar{0}$.

R1: $\bar{r} \equiv \diamond\bar{r}$.

R2: $\bar{r} \wedge \diamond\varphi \rightarrow \diamond(\bar{r} \wedge \varphi)$

T $_\diamond$: $\varphi \rightarrow \diamond\varphi$.

B $_\diamond$: $\varphi \rightarrow \neg\diamond\neg\diamond\varphi$.

4 $_\diamond$: $\diamond\diamond\varphi \rightarrow \diamond\varphi$.

RN $_\diamond^+$: *From $\varphi \rightarrow \psi$ infer $\diamond\varphi \rightarrow \diamond\psi$.*

MP : *From φ and $\varphi \rightarrow \psi$, infer ψ .*

We denote by $\vdash_{\bar{s}}$ the notion of derivability inside this logic.

Models are many-valued similarity-based Kripke model $M = \langle W, S, e \rangle$, in which $W \neq \emptyset$ is a set of possible worlds, S is a similarity relation on $W \times W$ and e represents an evaluation assigning to each atomic formula p_i and each interpretation $w \in W$ a truth value $e(p_i, w) \in [0, 1]$ of p_i in w . e is extended to formulas by means of Gödel logic truth functions by defining

$$e(\varphi \wedge \psi, w) = \min\{e(\varphi, w), e(\psi, w)\},$$

$$e(\varphi \rightarrow \psi, w) = e(\varphi, w) \Rightarrow_G e(\psi, w)$$

where \Rightarrow_G is the well-known Gödel implication function⁴, and

$$e(\bar{r}, w) = r, \text{ for all } r \in \mathbb{Q} \cap [0, 1],$$

$$e(\diamond\varphi, w) = \sup_{w' \in W} \min\{S(w, w'), e(\varphi, w')\}.$$

Based on the completeness results for RG [5] and for $GS5_\diamond$ [3] we state the following completeness result for $RGS5_\diamond$: a formula φ is provable in $RGS5_\diamond$, written $\vdash_S \varphi$, iff for every similarity Kripke model $M = \langle W, S, e \rangle$, $e(\varphi, w) = 1$ for every $w \in W$.

3 Relationship

Let $P = Facts \cup Rules \subset \mathcal{L}$ be a definite program, where $Facts \subseteq Const$ and

$$Rules = \{q_i \leftarrow p_{(i,1)}, p_{(i,2)}, \dots, p_{(i,n_i)} \mid i, n_i \in \mathbb{N}, p_{(i,j)}, q_i \in Const \forall i, j\}$$

are a set of facts and rules, respectively.

Let $\mathcal{R} : Const \times Const \rightarrow [0, 1]$ be a similarity relation on $Const$.

Define a mapping $*$: $\mathcal{L} \rightarrow \mathcal{L}_{G_\diamond}$ by

$$(q \leftarrow p_1, \dots, p_n)^* = \diamond p_1 \wedge \dots \wedge \diamond p_n \rightarrow \diamond q$$

$$(p_1, \dots, p_n)^* = \diamond p_1 \wedge \dots \wedge \diamond p_n$$

where $n \geq 1$ and the symbol “,” that separates the propositional constants on the left side has to be interpreted as conjunction. Then, we can define

$$Rules^* = \{\varphi^* \mid \varphi \in Rules\},$$

$$Crisp = \{p \vee \neg p \mid p \in Const\},$$

$$Sim = \{\mathcal{R}(p, q) \rightarrow ((p \rightarrow \diamond q) \wedge (q \rightarrow \diamond p)) \mid p, q \in Const\}$$

and the following theory in the language \mathcal{L}_{G_\diamond}

$$P_\diamond = Facts \cup Rules^* \cup Crisp \cup Sim.$$

Notice that the theory $Crisp$ ensures that the all propositional constants of P are treated as Boolean constants in P_\diamond . The aim is to show that one can derive in the similarity logic programming framework introduced in [14] a goal $\leftarrow q'$ from P with a unification degree α if and only if one can derive in $RGS5_\diamond$ the formula $\bar{\alpha} \rightarrow \diamond q'$ from P_\diamond . This result, besides to be relevant for itself because puts into relation different logical approaches to approximate reasoning, in particular will allows us to prove a formula in a theory P_\diamond in $RGS5_\diamond$ by using an Automatic Theorem Prover [10]. So far we have only been able to prove the “only if” direction. The rest of this section is devoted to this task.

Proposition 3. *Let $\mathcal{R} : Const \times Const \rightarrow [0, 1]$ be a similarity relation, $P = Rules \cup Facts$ a definite program on a propositional language \mathcal{L} and $\leftarrow q'$ a goal. If there exists a Similarity-based SLD derivation with approximation degree α for $P \cup \{\leftarrow q'\}$*

$$D = G_0 \Rightarrow_{C_1, \alpha_1} G_1 \Rightarrow \dots \Rightarrow_{C_k, \alpha_k} G_k$$

where $G_0 = \leftarrow q'$, $G_k \neq \perp$ and $\alpha = \min\{\alpha_1, \dots, \alpha_k\}$ then

$$P_\diamond \vdash_S \bar{\alpha} \rightarrow (G_k^* \rightarrow \diamond q')$$

Proof. We prove the thesis by induction on the length of D . If the length of D is zero, then the thesis is true. Indeed,

$$\vdash_S \diamond q' \rightarrow \diamond q' \tag{1}$$

$$\vdash_S (\diamond q' \rightarrow \diamond q') \rightarrow (\bar{1} \rightarrow (\diamond q' \rightarrow \diamond q')) \tag{2}$$

Then, from 1, 2 and modus ponens, follows that

$$\vdash_S \bar{1} \rightarrow (\diamond q' \rightarrow \diamond q')$$

⁴ \Rightarrow_G is defined as $x \Rightarrow_G y = 1$ if $x \leq y$ and $x \Rightarrow_G y = y$, otherwise

Now, let us suppose that the thesis is true for length of D equal k . Let

$$G_0 \Rightarrow_{C_1, \alpha_1} G_1 \Rightarrow \cdots \Rightarrow_{C_k, \alpha_k} G_k \Rightarrow_{C_{k+1}, \alpha_{k+1}} G_{k+1}$$

where $G_{k+1} \neq \perp$, be an existing Similarity-based SLD derivation for $P \cup \{\leftarrow q'\}$ of length $k+1$ with approximation degree $\alpha = \min\{\alpha_1, \dots, \alpha_k, \alpha_{k+1}\}$. Since $\min\{\alpha_1, \dots, \alpha_k\}$ is the approximation degree of $D_1 = G_0 \Rightarrow_{C_1, \alpha_1} G_1 \Rightarrow \cdots \Rightarrow_{C_k, \alpha_k} G_k$ then, by the inductive hypothesis,

$$P_\diamond \vdash_{\mathcal{S}} \overline{\min\{\alpha_1, \dots, \alpha_k\}} \rightarrow (G_k^* \rightarrow \diamond q') \quad (3)$$

Let us denote with q_i the head of the input clause $C_{k+1} = q_i \leftarrow p_{(i,1)}, p_{(i,2)}, \dots, p_{(i,n_i)}$, and with p the leftmost atom of $G_k = (\leftarrow p, \varphi)$ which is the selected atom in D_1 . Re-typing 3 in an equivalent manner, results that

$$P_\diamond \vdash_{\mathcal{S}} \overline{\min\{\alpha_1, \dots, \alpha_k\}} \rightarrow ((\diamond p \wedge \varphi^*) \rightarrow \diamond q')$$

thus,

$$P_\diamond \vdash_{\mathcal{S}} \diamond p \rightarrow (\overline{\min\{\alpha_1, \dots, \alpha_k\}} \rightarrow (\varphi^* \rightarrow \diamond q')) \quad (4)$$

Since $\alpha_{k+1} = \mathcal{R}(q_i, p)$,

$$\begin{aligned} P_\diamond \vdash_{\mathcal{S}} \overline{\alpha_{k+1}} &\rightarrow ((q_i \rightarrow \diamond p) \wedge (p \rightarrow \diamond q_i)) \\ P_\diamond \vdash_{\mathcal{S}} \overline{\alpha_{k+1}} &\rightarrow (q_i \rightarrow \diamond p) \\ P_\diamond \vdash_{\mathcal{S}} (\overline{\alpha_{k+1}} \wedge q_i) &\rightarrow \diamond p \end{aligned} \quad (5)$$

$$\begin{aligned} P_\diamond \vdash_{\mathcal{S}} \diamond(\overline{\alpha_{k+1}} \wedge q_i) &\rightarrow \diamond \diamond p \\ P_\diamond \vdash_{\mathcal{S}} (\overline{\alpha_{k+1}} \wedge \diamond q_i) &\rightarrow \diamond p \\ P_\diamond \vdash_{\mathcal{S}} \diamond q_i &\rightarrow (\overline{\alpha_{k+1}} \rightarrow \diamond p) \end{aligned} \quad (6)$$

We have to distinguish two cases:

(i) If $C_{k+1} \in \text{Rules}$ then

$$C_{k+1}^* = \diamond p_{(i,1)} \wedge \diamond p_{(i,2)} \wedge \cdots \wedge \diamond p_{(i,n_i)} \rightarrow \diamond q_i \in P_\diamond \quad (7)$$

Thus, by 6, 7, transitivity and modus ponens,

$$\begin{aligned} P_\diamond \vdash_{\mathcal{S}} (\diamond p_{(i,1)} \wedge \diamond p_{(i,2)} \wedge \cdots \wedge \diamond p_{(i,n_i)}) &\rightarrow (\overline{\alpha_{k+1}} \rightarrow \diamond p) \\ P_\diamond \vdash_{\mathcal{S}} (\overline{\alpha_{k+1}} \wedge \diamond p_{(i,1)} \wedge \cdots \wedge \diamond p_{(i,n_i)}) &\rightarrow \diamond p \end{aligned} \quad (8)$$

Then, by 8, 4, transitivity and modus ponens,

$$P_\diamond \vdash_{\mathcal{S}} (\overline{\alpha_{k+1}} \wedge \diamond p_{(i,1)} \wedge \cdots \wedge \diamond p_{(i,n_i)}) \rightarrow (\overline{\min\{\alpha_1, \dots, \alpha_k\}} \rightarrow (\varphi^* \rightarrow \diamond q'))$$

and

$$P_\diamond \vdash_{\mathcal{S}} \overline{\min\{\alpha_1, \dots, \alpha_{k+1}\}} \rightarrow ((\diamond p_{(i,1)} \wedge \cdots \wedge \diamond p_{(i,n_i)} \wedge \varphi^*) \rightarrow \diamond q')$$

(ii) If $C_{k+1} \in \text{Facts}$ then

$$C_{k+1} = q_i \in P_\diamond \quad (9)$$

Then, by 5, 4, transitivity and modus ponens,

$$P_\diamond \vdash_{\mathcal{S}} (\overline{\alpha_{k+1}} \wedge q_i) \rightarrow (\overline{\min\{\alpha_1, \dots, \alpha_k\}} \rightarrow (\varphi^* \rightarrow \diamond q'))$$

Thus,

$$P_\diamond \vdash_{\mathcal{S}} q_i \rightarrow (\overline{\alpha_{k+1}} \rightarrow (\overline{\min\{\alpha_1, \dots, \alpha_k\}} \rightarrow (\varphi^* \rightarrow \diamond q')))$$

$$P_{\diamond} \vdash_{\mathcal{S}} q_i \rightarrow ((\overline{\alpha_{k+1}} \wedge \overline{\min\{\alpha_1, \dots, \alpha_k\}}) \rightarrow (\varphi^* \rightarrow \diamond q'))$$

$$P_{\diamond} \vdash_{\mathcal{S}} q_i \rightarrow (\overline{\min\{\alpha_1, \dots, \alpha_{k+1}\}} \rightarrow (\varphi^* \rightarrow \diamond q'))$$

and, by 9 and modus ponens,

$$P_{\diamond} \vdash_{\mathcal{S}} \overline{\min\{\alpha_1, \dots, \alpha_{k+1}\}} \rightarrow (\varphi^* \rightarrow \diamond q').$$

□

Corollary 1. *Let $\mathcal{R} : Const \times Const \rightarrow [0, 1]$ be a similarity relation, $P = Rules \cup Facts$ a definite program on a propositional language \mathcal{L} and $\leftarrow q'$ a goal. If there exists a Similarity-based SLD refutation with approximation degree α for $P \cup \{\leftarrow q'\}$*

$$D = G_0 \Rightarrow_{C_1, \alpha_1} G_1 \Rightarrow \dots \Rightarrow_{C_{k-1}, \alpha_{k-1}} G_{k-1} \Rightarrow_{C_k, \alpha_k} \perp$$

where $G_0 = \leftarrow q'$ and $\alpha = \min\{\alpha_1, \dots, \alpha_k\}$, then

$$P_{\diamond} \vdash_{\mathcal{S}} \overline{\alpha} \rightarrow \diamond q'$$

Proof. In D necessarily $G_{k-1} = \leftarrow p$ for some $p \in Const$, $C_k = q_i \in Facts$ and $\alpha_k = \mathcal{R}(p, q_i)$. Thus, since $G_{k-1} \neq \perp$, by Proposition 3 follows that

$$P_{\diamond} \vdash_{\mathcal{S}} \overline{\min\{\alpha_1, \dots, \alpha_{k-1}\}} \rightarrow (\diamond p \rightarrow \diamond q')$$

Moreover, from $\alpha_k = \mathcal{R}(p, q_i)$ results that

$$P_{\diamond} \vdash_{\mathcal{S}} (\overline{\alpha_k} \wedge q_i) \rightarrow \diamond p$$

Then, by transitivity and modus ponens,

$$P_{\diamond} \vdash_{\mathcal{S}} (\overline{\alpha_k} \wedge q_i) \rightarrow (\overline{\min\{\alpha_1, \dots, \alpha_{k-1}\}} \rightarrow \diamond q')$$

$$P_{\diamond} \vdash_{\mathcal{S}} q_i \rightarrow (\overline{\alpha_k} \rightarrow (\overline{\min\{\alpha_1, \dots, \alpha_{k-1}\}} \rightarrow \diamond q'))$$

$$P_{\diamond} \vdash_{\mathcal{S}} q_i \rightarrow ((\overline{\alpha_k} \wedge \overline{\min\{\alpha_1, \dots, \alpha_{k-1}\}}) \rightarrow \diamond q')$$

$$P_{\diamond} \vdash_{\mathcal{S}} q_i \rightarrow (\overline{\min\{\alpha_1, \dots, \alpha_k\}} \rightarrow \diamond q')$$

Thus, since $q_i \in Facts$ implies that $q_i \in P_{\diamond}$, by modus ponens follows that

$$P_{\diamond} \vdash_{\mathcal{S}} \overline{\min\{\alpha_1, \dots, \alpha_k\}} \rightarrow \diamond q'$$

□

4 Related works and conclusions

In the paper we have established some syntactic relationships between two approaches to similarity reasoning, namely SPLP and RGS5 $_{\diamond}$. It remains as a future task to check whether the original aim of this paper of having a full relationship can be devised, also regarding predicate languages. On the other hand, in the literature there are other approaches to similarity-based reasoning, both in the fuzzy logic programming framework, like [11], as well as within other logical formalisms [4]. It will also be an interesting future work to study possible links among all these formalisms.

References

1. R.K. Apt, "Logic Programming", in: J. van Leeuwen (Ed.), Handbook of Theoretical Computer Science, vol. B, Elsevier, Amsterdam, 1990, pp. 492-574.
2. L. Biacino, G. Gerla, M. Ying, "Approximate reasoning based on similarity", in: Math. Logic Quart., 46, 2000.
3. X. Caicedo, R. Rodriguez, "A Gödel modal logic", in: Proc. of Logic, Computability and Randomness 2004. Cordoba, Argentina.
4. F. Esteva, P. Garcia, L. Godo, "On syntactical and semantical approaches to similarity-based approximate reasoning", in: Proc. Joint 9th IFSA and 20th NAFIPS 2001, July 25-28, Vancouver (Canada) pp. 1598-1603.
5. F. Esteva, L. Godo, C. Noguera, "On Rational Gödel and Nilpotent Minimum logics", in: Proc. of the Xth International Conference IPMU 2004, Perugia (Italia), July 4-9, pp. 561-568.
6. F. Formato, G. Gerla and M.I. Sessa, "Similarity-based unification", in: Fundamenta Informaticae 40, 2000, pp. 1-22.
7. G. Gerla and M.I. Sessa, "Similarity in Logic Programming", in: G. Chen, M. Ying, K.-Y. Cai (Ed.s), Fuzzy Logic and Soft Computing, Kluwer Acc. Pub., Norwell, 1999, pp. 19-31.
8. G. Gerla, M.I. Sessa, "Similarity Logic and Similarity PROLOG", in: Proc. Joint EUROFUSE-SIC99 International Conference, 25-28 May 1999, Budapest, Hungary, pp.367-372.
9. L. Godo, R. Rodriguez, "A Fuzzy Modal Logic for Similarity Reasoning", in: Fuzzy Logic and Soft Computing, Guoqing Chen, Mingsheng Ying and Kai-Yuan Cai eds., Kluwer, 1999, pp. 33-48.
10. V. Loia, S. Senatore, M.I. Sessa, "Similarity-based SLD Resolution and its implementation in an Extended Prolog System", in: Proceedings of 10th IEEE International Conference on Fuzzy Systems, Melbourne, Australia, December 2-5 2001. IEEE PRESS.
11. J. Medina, M. Ojeda-Aciego, P. Vojtáš, "Similarity-based unification: a multi-adjoint approach", in: Fuzzy Sets and Systems, 146, 2004, pp. 43-62.
12. E.H. Ruspini, "On the semantics of fuzzy logic", in: Int. Journal of Approximate Reasoning, 5, 1991, pp. 45-88.
13. M.I. Sessa, "Translations and Similarity-based Logic Programming", in: Soft Computing 5(2), 2001.
14. M.I. Sessa, "Approximate Reasoning by Similarity-based SLD Resolution", in: Theoretical Computer Science, 275, 2002, pp. 389-426.
15. M.S. Ying, "A Logic for Approximated Reasoning", in: The Journal of Symbolic Logic, 59, 1994, pp. 830-837.
16. L.A. Zadeh, "Similarity Relations and Fuzzy Orderings", in: Information Sciences 3, 1971, pp. 177-200.