

## RESEARCH

## Open Access

# Searching contexts in paraconsistent rough description logic

Henrique Viana<sup>\*†</sup>, João Alcântara<sup>†</sup> and Ana Teresa Martins**Abstract**

**Background:** Query refinement is an interactive process of query modification used to increase or decrease the scope of search results in databases or ontologies.

**Methods:** We present a method to obtain optimized query refinements of assertion axioms in the paraconsistent rough description logic  $\mathcal{PR}_{\mathcal{ALC}}$ , a four-valued paraconsistent version of the rough  $\mathcal{ALC}$ , which is grounded on Belnap's Logic. This method is based on the notion of the discernibility matrix commonly used in the process of attribute reduction in the rough set theory. It consists of finding sets of concepts which satisfy the rough set approximation operations in assertion axioms. Consequently, these sets of concepts can be used to restrict or relax queries in this logic.

**Results:** We propose two algorithms to settle this problem of query refinement in  $\mathcal{PR}_{\mathcal{ALC}}$  and show their complexity results.

**Conclusions:** The problem of query restrictions using contextual approximation is proved to have exponential time complexity, while the problem of query relaxations has polynomial space complexity.

**Keywords:** Description logics; Rough sets; Query refinement

**Background**

In large databases as hypertext document collections, it is often possible to find too many results available for specific queries, and it is not always that all these answers are important. In an opposite situation (but potentially undesirable), too few results can be available for queries with keywords rarely used. Some approaches based on query refinements can, then, be employed to settle these problems. As we know, query refinement is an interactive process of query modification used to increase or decrease the scope of search results.

One of these approaches is the rough set theory introduced by Z. Pawlak [1] to represent and reason about uncertainty through two operations of set approximation: the lower and the upper approximations. For a set  $S$ , its lower approximation gives the set of elements that certainly belong to  $S$ , while its upper approximation gives the set of elements that possibly belong to  $S$ .

Because of such approximations, the rough set theory may be applied to produce two forms of query refinement: query restriction and query relaxation. A query can be restricted in order to obtain only the necessary results, or it can be relaxed, aiming at increasing the number of its results. In this paper, we propose a new method settled on the rough set theory and description logics (DLs) for automatically generating refinements or related terms to queries.

Some works can be found in the literature with respect to query refinement in DLs as in [2, 3]. Most of them focus on syntactical manipulations of knowledge bases to increase or decrease the number of results of a query. Unfortunately, these syntactical manipulations may generate results without connections with the initial query. In [4], by resorting to rough sets as a tool of query refinement in DLs, it is ensured that the obtained results always have some kind of relationship with the initial query, because of the role played by the approximation operator [5]. In addition, the representation of rough sets in DLs brings no growth in the complexity of the corresponding satisfiability problem.

\*Correspondence: [henrique.viana.oliveira@gmail.com](mailto:henrique.viana.oliveira@gmail.com)

†Equal contributors

Departamento de Computação, Universidade Federal do Ceará, P.O. Box 12166, 60455-760 Fortaleza, CE, Brazil

Still, in [4], in order to generate query refinements of assertion axioms, the authors employed the notion of contexts in rough set approximations [6]. However, these contexts are given in the query. This problem is solved in [7], where a method is proposed to obtain optimized query refinements of assertion axioms in rough  $\mathcal{ALC}$  [6]. We will now extend this result to work with paraconsistent rough description logic ( $\mathcal{PR}_{\mathcal{ALC}}$ ) [4]. It is a four-valued extension of the rough  $\mathcal{ALC}$ , and its semantics follows the well-known Belnap's paraconsistent logic [8]. With that, we increase the expressivity of uncertainty representation, allowing the expression and approximation of unknown and contradictory knowledge bases. It will be achieved by resorting to two algorithms to search for contexts which will be applied in the approximation operations. These algorithms generate optimized solutions when they exist. Depending on the considered refinement, minimal or maximal cardinality sets which satisfy approximations are chosen (minimal sets for query restrictions and maximal sets for query relaxations). Furthermore, we present complexity results of both algorithms.

The paper is structured as follows. In the “Background” section, we present the basic notions of rough  $\mathcal{ALC}$ , an extension of  $\mathcal{ALC}$  with the approximations of the rough set theory, and we introduce the paraconsistent extension of the rough  $\mathcal{ALC}$ , called  $\mathcal{PR}_{\mathcal{ALC}}$ . In the “Methods” and “Results and discussion” sections, we will find the main contributions of this work, where we define a method to obtain optimized refined queries in this logic and we present the corresponding algorithms (as well as their complexity analysis), respectively. Finally, in the “Conclusions” section, we conclude the paper.

### Rough description logic $\mathcal{ALC}$

Rough description logics (RDLs) [6, 7, 9–12] introduced a mechanism to model uncertain reasoning by means of concept approximation. They extend DLs with two operations: the lower and upper approximations. Both approximations are conceived to capture uncertainty from an indiscernibility relation (an equivalence relation). We can define the upper approximation of a concept  $C$  in  $\mathcal{ALC}$  as the set of individuals in  $C$  that are indiscernible from at least an individual known to belong to  $C$  [6]. Similarly, we can define the lower approximation of a concept  $C$  as the set of all indiscernible individuals in  $C$ . In the sequel, we will introduce some basic characteristics of the rough  $\mathcal{ALC}$ , beginning with the syntax, semantics, and, lastly, alternative approaches to represent approximations, which will be used later in the query refinements. For a detailed explanation about DLs and rough sets, see [13] and [14], respectively.

### Syntax

As mentioned above, the basic idea behind RDLs is straightforward: we can approximate an uncertain concept  $C$  through lower and upper bounds.

**Definition 1. (Concepts)** Concepts in rough  $\mathcal{ALC}$  are defined by the following rules, where  $C$  and  $D$  are concepts,  $A$  is an atomic concept, and  $R$  is an atomic role:

$$C, D \longrightarrow A \mid \perp \mid \neg C \mid C \sqcap D \mid \exists R.C \mid \overline{C} \mid \underline{C}.$$

The rough  $\mathcal{ALC}$  is based on  $\mathcal{ALC}$  with the addition of the upper and lower approximation as unary constructors of concepts, i.e., if  $C$  is a concept then  $\overline{C}$  (possibly  $C$ ) and  $\underline{C}$  (necessarily  $C$ ) are also concepts. Others concepts can be defined by the following equivalences:  $\top \equiv \neg \perp$ ,  $(C \sqcup D) \equiv \neg(\neg C \sqcap \neg D)$ ,  $\forall R.C \equiv \neg(\exists R.\neg C)$ .

The notions of TBox and ABox, as well the knowledge base in rough  $\mathcal{ALC}$ , extend the original notion of  $\mathcal{ALC}$ .

**Definition 2. (TBox)** A TBox  $\mathcal{T}$  is a finite set of terminological axioms of the form  $C \sqsubseteq D$  or  $C \equiv D$ .

The first axiom  $C \sqsubseteq D$  (inclusion axiom) means that each individual of  $C$  is also an instance of  $D$ , while the axiom  $C \equiv D$  (equivalence axiom) means that each individual of  $C$  is also an instance of  $D$  and each individual of  $D$  is also an instance of  $C$ .

**Definition 3. (ABox)** An ABox  $\mathcal{A}$  consists of the finite set of assertion axioms of the form  $C(a)$  or  $R(a, b)$ .

The concept assertion  $C(a)$  denotes that the individual  $a$  belongs to the concept  $C$ , and the role assertion  $R(a, b)$  denotes that the individual  $a$  is related to individual  $b$  by the role  $R$ .

**Definition 4. (Knowledge base)** A knowledge base  $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$  in rough  $\mathcal{ALC}$  consists of a TBox  $\mathcal{T}$  and an ABox  $\mathcal{A}$ .

### Semantics

The semantics of rough  $\mathcal{ALC}$  is given by an interpretation  $I = (\Delta^I, \cdot^I, R^\sim)$ , where  $\Delta^I$  is a domain set,  $\cdot^I$  is an interpretation function, and  $R^\sim$  is an equivalence relation on  $\Delta^I$ , which will be used in concept approximations. The function  $\cdot^I$  maps atomic concepts to subsets of  $\Delta^I$  and role names to binary relations on the domain  $\Delta^I$ . The interpretation for complex concepts remains the same as in  $\mathcal{ALC}$ :

- For an individual  $a$ ,  $a^I \in \Delta^I$ ;
- For atomic concepts  $A$ ,  $A^I \subseteq \Delta^I$ ;
- For atomic roles  $R$ ,  $R^I \subseteq \Delta^I \times \Delta^I$ ;
- $\top^I = \Delta^I$ ;
- $\perp^I = \emptyset$ ;

- $(\neg A)^I = \Delta^I \setminus A^I$ ;
- $(C \sqcap D)^I = C^I \cap D^I$ ;
- $(C \sqcup D)^I = C^I \cup D^I$ ;
- $(\exists R.C)^I = \{a \in \Delta^I \mid \exists b \in \Delta^I, (a, b) \in R^I \wedge b \in C^I\}$ ;
- $(\forall R.C)^I = \{a \in \Delta^I \mid \forall b \in \Delta^I, (a, b) \in R^I \rightarrow b \in C^I\}$ .

For the approximation of concepts, we will have:

- $(\overline{C})^I = \{a \in \Delta^I \mid \exists b ((a, b) \in R^\sim \wedge b \in C^I)\}$ ,
- $(\underline{C})^I = \{a \in \Delta^I \mid \forall b ((a, b) \in R^\sim \rightarrow b \in C^I)\}$ .

One of the advantages of this method of modeling uncertainties in concepts is that it does not increase the complexity of inference with respect to the original logic without upper and lower approximations. In fact, reasoning with RDLs can be reduced to reasoning with DLs, by translating rough concepts in usual DL concepts with a new relation which is reflexive, symmetric, and transitive. On the other hand, this method does not increase its expressivity with respect to DLs, i.e., by resorting only to DL, it is possible to simulate the approximations. A translation function of concepts  $\cdot^t : \text{RDL} \rightarrow \text{DL}$  can be defined to show this equivalence of expressivity (introducing a new role name  $R^\sim$  for the equivalence relation):

- $A^t = A$ , for all atomic concepts  $A$  in RDL,
- $(\overline{C})^t = \exists R^\sim.C^t$  e  $(\underline{C})^t = \forall R^\sim.C^t$ .

For all others complex concepts, the translation function is applied recursively in their subconcepts. The same definition is extended to inclusion and assertion axioms [11, 12].

A different way of representing equivalence between individuals was also proposed in [6], in which an alternative approach to approximation in DL was introduced. In this work, an approximation depends on a specific set of concepts to determine the indiscernibility of the individuals (and not anymore an explicit indiscernibility relation). We will detail this idea in the sequel.

### Contextual approximation

In [6], the notion of contextual indiscernibility relation in RDLs was introduced as a manner to define an equivalence relation based on indiscernibility criteria. In particular, the notion of context is introduced, allowing the definition of specific equivalence relationships to be applied on approximations. The great advantage of this approach is that the reasoning with equivalence classes is optimized, because the equivalence relation is discovered in the process of inference, differently from the traditional RDL, where the equivalence relation must be explicitly defined.

We will show the definition of contexts of a collection of concepts and successively the definitions of lower and upper approximations through a context. First, we will present the notion of a projection function in DL [6]:

**Definition 5. (Projection)** Let  $\mathcal{K}$  be a knowledge base and  $A$  an atomic concept. The projection function  $\pi_A^{\mathcal{K}} : N_I \rightarrow \{0, *, 1\}$  is defined as

$$\forall a \in N_I : \pi_A^{\mathcal{K}}(a) = \begin{cases} 1, & \mathcal{K} \models A(a); \\ 0, & \mathcal{K} \models \neg A(a); \\ *, & \text{otherwise.} \end{cases}$$

where  $N_I$  is the set of individuals in the knowledge base  $\mathcal{K}$ .

A context can be defined as a finite set of relevant features in the form of DL concepts, which may encode a kind of context information for the similarity to be measured [6].

**Definition 6. (Context)** A context is a nonempty finite set of atomic concepts  $\Sigma = \{A_1, \dots, A_n\}$ .

Two individuals  $a$  and  $b$  are indiscernible with respect to the context  $\Sigma = \{A_1, \dots, A_n\}$  and a knowledge base  $\mathcal{K}$  if and only if for all  $A_i$  in which  $i \in \{1, \dots, n\}$ ,  $\pi_{A_i}^{\mathcal{K}}(a) = \pi_{A_i}^{\mathcal{K}}(b)$ . This easily induces an equivalence relation:

**Definition 7. (Contextual indiscernibility relation)** Let  $\Sigma = \{A_1, \dots, A_n\}$  be a context and  $\mathcal{K}$  a knowledge base. The indiscernibility relation  $R_\Sigma$  induced by  $\Sigma$  is defined as:  $R_\Sigma = \{(a, b) \in N_I \times N_I \mid \text{for all } A_i \text{ in which } i \in \{1, \dots, n\}, \pi_{A_i}^{\mathcal{K}}(a) = \pi_{A_i}^{\mathcal{K}}(b)\}$ .

As DLs induce the representation of uncertain information when it is the case that  $\mathcal{K} \not\models A(a)$  and  $\mathcal{K} \not\models \neg A(a)$ , a similarity relation (instead of an equivalence relation) may be more adequate to model relationships between individuals, because it permits grouping of individuals that are close, but not necessarily indiscernible. Formally, a binary relation is a similarity relation if it is at least reflexive (an equivalence relation is reflexive, symmetric, and transitive). We introduce the following similarity relation based in [15], which loosens the original condition of indiscernibility:

**Definition 8. (Contextual similarity relation)** Let  $\Sigma = \{A_1, \dots, A_n\}$  be a context and  $\mathcal{K}$  a knowledge base. The similarity relation  $S_\Sigma$  induced by  $\Sigma$  is defined as

$$S_\Sigma = \{(a, b) \in N_I \times N_I \mid \text{for all } A_i \text{ in which } i \in \{1, \dots, n\}, \pi_{A_i}^{\mathcal{K}}(a) = \pi_{A_i}^{\mathcal{K}}(b) \text{ or } \pi_{A_i}^{\mathcal{K}}(a) = * \text{ or } \pi_{A_i}^{\mathcal{K}}(b) = *\}$$

The contextual approximations are defined below.

**Definition 9. (Contextual upper/lower approximation)** Let  $\Sigma$  be a context,  $C$  a concept,  $I$  an interpretation, and  $\text{Sim} \in \{R, S\}$ . The contextual upper and lower approximations of  $C$  w.r.t.  $\Sigma$ , are defined as

- $(\overline{C}^{\text{Sim}_\Sigma})^I = \{a \in \Delta^I \mid \exists b((a, b) \in \text{Sim}_\Sigma^I \wedge b \in C^I)\}$ ,
- $(\underline{C}_{\text{Sim}_\Sigma})^I = \{a \in \Delta^I \mid \forall b((a, b) \in \text{Sim}_\Sigma^I \rightarrow b \in C^I)\}$ .

We can generalize the definitions of contextual approximations by using the notion of *k-step relation* [16].

**Definition 10. (k-step relation)** Let  $\Delta^I$  be the nonempty universe set,  $S$  a binary relation in  $\Delta^I$ , and  $k$  a natural number. The *k-step relation* of  $S$  is defined as:

- $S_1 = S$ ;
- $S_{k+1} = S_k \cup \{(x, y) \in \Delta^I \times \Delta^I \mid \text{exists } y_1, y_2, \dots, y_k \in \Delta^I, \text{ such that } xSy_1, y_1Sy_2, \dots, y_kSy\}$ , for  $k \geq 1$ .

The idea behind the *k-step relation* is that when we are using a similarity relation, e.g.,  $S_\Sigma$ , it may happen that  $S_{\Sigma_1} \subseteq S_{\Sigma_2} \subseteq \dots \subseteq S_{\Sigma_n}$ . As a consequence, the successive application of approximation operations may imply in different results.

**Definition 11. (Loose and tight approximations)** For the loose upper approximation and the tight lower approximation of a concept  $C$  w.r.t., the similarity relation  $\text{Sim}$  in  $n$  steps is denoted by  $\overline{C}^{(\text{Sim}, n)}$  and  $\underline{C}_{(\text{Sim}, n)}$ , respectively, and defined as

- $(\overline{C}^{(\text{Sim}_\Sigma, n)})^I = \{a \mid \exists b((a, b) \in \text{Sim}_\Sigma^I \wedge b \in C^I)\}$ ,
- $(\underline{C}_{(\text{Sim}_\Sigma, n)})^I = \{a \mid \forall b((a, b) \in \text{Sim}_\Sigma^I \rightarrow b \in C^I)\}$ .

The contextual approximations will play a central role in the process of query refinement. In this paper, refining a concept means to apply a lower (restriction of a concept) or an upper (relaxation of a concept) approximation. Hence, it is needed to identify first a context that can be used in approximations. This process of finding contexts and some more relevant questions will be tackled and exemplified in the next sections.

### Paraconsistent rough description logic $\mathcal{PR}_{\mathcal{ALC}}$

We begin this section by presenting a brief explanation about Belnap's logic. Subsequently, we will show a paraconsistent extension of rough  $\mathcal{ALC}$ , based on Belnap's semantics, called  $\mathcal{PR}_{\mathcal{ALC}}$ .

#### Belnap's logic

Belnap's logic [8] has four truth values instead of the two classic truth values true and false. These values are **t** (true), **f** (false), **u** (unknown), and **i** (inconsistent). The truth value **i** represents a contradictory information, while **u** means neither true nor false, i.e., the absence of any information about veracity or falsity.

Syntactically, Belnap's logic is very similar to classical logic. However, it introduces different notions of implication. In fact, we will show three of these notions brought from literature. The connectives that will be used here are negation ( $\neg$ ), disjunction ( $\vee$ ), conjunction ( $\wedge$ ), material implication ( $\mapsto$ ), internal implication ( $\supset$ ), and strong implication ( $\rightarrow$ ) [17].

The interpretations of the formulas are mappings of the set formulas to one of the possible four truth values, respecting the truth table of the connectives as detailed in the Tables 1 and 2. The semantics of the truth tables of the implications are:

- $x \mapsto y$  is defined as  $\neg x \vee y$ ;
- $x \supset y$  is evaluated to  $y$  if  $x \in \{\mathbf{t}, \mathbf{i}\}$ ;  $\mathbf{t}$  if  $x \in \{\mathbf{f}, \mathbf{u}\}$ ;
- $x \rightarrow y$  is defined as  $(x \supset y) \wedge (\neg y \supset \neg x)$ .

Models are defined as follows, in which  $\{\mathbf{t}, \mathbf{i}\}$  are the designated values (i.e., the truth values considered satisfiable with respect to the consequence relation defined below).

**Definition 12. (Model)** Let  $I$  be a four-valued interpretation,  $\Gamma$  a set of formulas, and  $\varphi$  a formula in Belnap's logic. We say that  $I$  is a four-valued model of  $\varphi$  if and only if  $\varphi^I \in \{\mathbf{t}, \mathbf{i}\}$ .  $I$  is a four-valued model of  $\Gamma$  if and only if  $I$  is a four-valued model of each formula in  $\Gamma$ . We say  $\Gamma$  entails  $\varphi$ , written  $\Gamma \models \varphi$ , if and only if every four-valued model of  $\Gamma$  is also a four-valued model of  $\varphi$ .

#### Defining $\mathcal{PR}_{\mathcal{ALC}}$

Now, we will describe the syntax and the semantics of the paraconsistent rough description logic  $\mathcal{ALC}$  ( $\mathcal{PR}_{\mathcal{ALC}}$ ). Such a logic is an extension of the description logic  $\mathcal{ALC}_4$  [17], with the addition of the lower and upper approximation operators.

Syntactically,  $\mathcal{PR}_{\mathcal{ALC}}$  almost does not differ from rough  $\mathcal{ALC}$ . Complex concepts and the assertion axioms are defined in the same way:

$$C, D \longrightarrow A \mid \perp \mid \neg C \mid C \sqcap D \mid \exists R.C \mid \overline{C} \mid \underline{C}.$$

For concept inclusion axioms, we have three types of inclusions, relative to the three implications showed before:

**Table 1** Truth tables for  $\wedge, \vee$ , and  $\neg$

$\wedge$	f	u	i	t	$\vee$	f	u	i	t	$\neg$
f	f	f	f	f	f	f	u	i	t	f
u	f	u	f	u	u	u	u	t	t	u
i	f	f	i	i	i	i	t	i	t	i
t	f	u	i	t	t	t	t	t	t	t

**Table 2** Truth tables for  $\mapsto$ ,  $\supset$ , and  $\rightarrow$ 

$\mapsto$	f	u	i	t	$\supset$	f	u	i	t	$\rightarrow$	f	u	i	t
f	t	t	t	t	f	t	t	t	t	f	t	t	t	t
u	u	u	t	t	u	t	t	t	t	u	u	t	u	t
i	i	t	i	t	i	f	u	i	t	i	f	u	i	t
t	f	u	i	t	t	f	u	i	t	t	f	u	f	t

$C \mapsto D$ (material inclusion axiom),

$C \sqsubset D$ (internal inclusion axiom),

$C \rightarrow D$ (strong inclusion axiom).

As usual, semantically, interpretations map individuals to elements of the domain of interpretation. For concepts and atomic roles, however, some changes in the notion of interpretation need to be made in order to reason with inconsistencies.

Intuitively, in a four-valued logic, we need to consider four situations that can occur in terms of membership of an individual in a concept: (1) we know that it belongs to the set, (2) we know that it does not belong to the set, (3) we do not have any knowledge if it belongs or if it does not belong, and (4) we have a contradictory information, stating that the individual belongs and does not belong to the concept. There are many equivalent ways to formalize this notion; one of them will be described in the sequel.

For a given domain  $\Delta^I$  and a concept  $C$ , an interpretation on  $\Delta^I$  maps to  $C$  a pair  $\langle P, N \rangle$  of subsets (not necessarily disjoint) of  $\Delta^I$ , in which  $P$  is the set of the elements that we know belong to  $C$  (positive information), while  $N$  is the set of the elements that we know do not belong to  $C$  (negative information). Consider the functions  $\text{proj}^+$  and  $\text{proj}^-$ , the projections of the positive and negative part, respectively, which are defined as

$$\text{proj}^+ \langle P, N \rangle = P \text{ and } \text{proj}^- \langle P, N \rangle = N.$$

In more formal terms, we can define a four-valued interpretation as displayed below:

**Definition 13. (Four-valued interpretation)** A four-valued interpretation is a triple  $I = (\Delta^I, \cdot^I, R^\sim)$  with  $\Delta^I$  as domain,  $R^\sim$  is an equivalence relation on  $\Delta^I$ , and  $\cdot^I$  is a mapping function from elements of  $\Delta^I$  to individuals and subsets of  $\Delta^I \times \Delta^I$  to concepts such that the conditions showed below are satisfied:

- For atomic concepts  $A$ ,  $A^I = \langle P, N \rangle$ , such that  $P, N \subseteq \Delta^I$ ;
- For atomic roles  $R$ ,  $R^I = \langle P_1 \times P_2, N_1 \times N_2 \rangle$ , such that  $P_1 \times P_2, N_1 \times N_2 \subseteq \Delta^I \times \Delta^I$ ;
- $\top^I = \langle \Delta^I, \emptyset \rangle$ ;
- $\perp^I = \langle \emptyset, \Delta^I \rangle$ ;
- $(\neg C)^I = \langle N, P \rangle$  if  $C^I = \langle P, N \rangle$ ;

- $(C \sqcap D)^I = \langle P_1 \cap P_2, N_1 \cup N_2 \rangle$ , if  $C^I = \langle P_1, N_1 \rangle$  and  $D^I = \langle P_2, N_2 \rangle$ ;
- $(C \sqcup D)^I = \langle P_1 \cup P_2, N_1 \cap N_2 \rangle$ , if  $C^I = \langle P_1, N_1 \rangle$  and  $D^I = \langle P_2, N_2 \rangle$ ;
- $(\exists R.C)^I = \{ \{x \mid \exists y, (x, y) \in \text{proj}^+(R^I) \wedge y \in \text{proj}^+(C^I)\}, \{x \mid \forall y, (x, y) \in \text{proj}^+(R^I) \rightarrow y \in \text{proj}^-(C^I)\} \}$ ;
- $(\forall R.C)^I = \{ \{x \mid \forall y, (x, y) \in \text{proj}^+(R^I) \rightarrow y \in \text{proj}^+(C^I)\}, \{x \mid \exists y, (x, y) \in \text{proj}^+(R^I) \wedge y \in \text{proj}^-(C^I)\} \}$ ;
- $(\bar{C})^I = \{ \{x \mid \exists y, (x, y) \in R^\sim \wedge y \in \text{proj}^+(C^I)\}, \{x \mid \forall y, (x, y) \in R^\sim \rightarrow y \in \text{proj}^-(C^I)\} \}$ ;
- $(\underline{C})^I = \{ \{x \mid \forall y, (x, y) \in R^\sim \rightarrow y \in \text{proj}^+(C^I)\}, \{x \mid \exists y, (x, y) \in R^\sim \wedge y \in \text{proj}^-(C^I)\} \}$ .

Note that the conditions above for the role restrictions are described in a way that the logical equivalences  $\neg(\forall R.C) = \exists R.(\neg C)$  and  $\neg(\exists R.C) = \forall R.(\neg C)$  are preserved. This was the convenient manner found in [17] to deal with role restrictions and that allows a direct translation to  $\mathcal{ALC}$ . Note that in this language, only the positive part of the interpretation of the role is required, because it involves only atomic roles.

Obviously, under the restrictions that  $P \cap N = \emptyset$  and  $P \cup N = \Delta^I$ , the four-valued interpretations collapse into the usual two-valued case. The correspondence between the truth values and the membership of concepts and roles is described in the following way: let  $a, b \in \Delta^I$ ,  $C$  be a concept name and  $R$  a role name. We have that:

- $C^I(a) = \mathbf{t}$ , iff  $a^I \in \text{proj}^+(C^I)$  and  $a^I \notin \text{proj}^-(C^I)$ ;
- $C^I(a) = \mathbf{f}$ , iff  $a^I \notin \text{proj}^+(C^I)$  and  $a^I \in \text{proj}^-(C^I)$ ;
- $C^I(a) = \mathbf{i}$ , iff  $a^I \in \text{proj}^+(C^I)$  and  $a^I \in \text{proj}^-(C^I)$ ;
- $C^I(a) = \mathbf{u}$ , iff  $a^I \notin \text{proj}^+(C^I)$  and  $a^I \notin \text{proj}^-(C^I)$ ;
- $R^I(a, b) = \mathbf{t}$ , iff  $(a^I, b^I) \in \text{proj}^+(R^I)$  and  $(a^I, b^I) \notin \text{proj}^-(R^I)$ ;
- $R^I(a, b) = \mathbf{f}$ , iff  $(a^I, b^I) \notin \text{proj}^+(R^I)$  and  $(a^I, b^I) \in \text{proj}^-(R^I)$ ;
- $R^I(a, b) = \mathbf{i}$ , iff  $(a^I, b^I) \in \text{proj}^+(R^I)$  and  $(a^I, b^I) \in \text{proj}^-(R^I)$ ;
- $R^I(a, b) = \mathbf{u}$ , iff  $(a^I, b^I) \notin \text{proj}^+(R^I)$  and  $(a^I, b^I) \notin \text{proj}^-(R^I)$ .

Lastly, it follows the semantics of the different kinds of axioms.

- $I \models C \mapsto D$  iff  $\Delta^I \setminus \text{proj}^-(C^I) \subseteq \text{proj}^+(D^I)$ ;
- $I \models C \sqsubset D$  iff  $\text{proj}^+(C^I) \subseteq \text{proj}^+(D^I)$ ;
- $I \models C \rightarrow D$  iff  $\text{proj}^+(C^I) \subseteq \text{proj}^+(D^I)$  and  $\text{proj}^-(D^I) \subseteq \text{proj}^-(C^I)$ ;
- $I \models C(a)$  iff  $a^I \in \text{proj}^+(C^I)$ ;
- $I \models R(a, b)$  iff  $(a^I, b^I) \in \text{proj}^+(R^I)$ .

We say that a four-valued interpretation  $I$  satisfies a knowledge base  $\mathcal{K}$  (i.e.,  $I$  is a model of  $\mathcal{K}$ ) if and only

if it satisfies each inclusion and assertion axiom in  $\mathcal{K}$ . A knowledge base  $\mathcal{K}$  is satisfiable (respectively, unsatisfiable) if and only if there exists (respectively, there does not exist) a model for  $\mathcal{K}$ .

Considering the complexity of satisfiability of  $\mathcal{PR}_{\mathcal{ALCC}}$ , it was proved in [17] that the complexity of the satisfiability decision problem for the paraconsistent version of  $\mathcal{ALCC}$  is equivalent to the complexity of the same problem for  $\mathcal{ALCC}$ . This result shows that the paraconsistent reasoning is not more expressive than the classical two-valued reasoning and it can be simulated in a two-valued  $\mathcal{ALCC}$  without an increasing of complexity. To show such a result, a polynomial translation of a paraconsistent knowledge base to a knowledge base of  $\mathcal{ALCC}$  was described, which preserves all of its inference properties. According to this result, we can easily show that the satisfiability decision problem for  $\mathcal{PR}_{\mathcal{ALCC}}$  has the same complexity of this problem for  $\mathcal{ALCC}$  through the same translations presented in [11, 12, 17]

As in rough  $\mathcal{ALCC}$ , we can also define the contextual approximation related to  $\mathcal{PR}_{\mathcal{ALCC}}$ .

**Definition 14. (Four-valued contextual approximations)** Let  $\Sigma$  be a context,  $C$  be a concept,  $I$  be a four-valued interpretation, and  $\text{Sim}$  be a similarity relation. The contextual upper and lower approximations of  $C$  with respect to  $\Sigma$  are defined as:

- $(\overline{C}^{\text{Sim}_\Sigma})^I = \{\{x \mid \exists y, (x, y) \in \text{Sim}_\Sigma^I \wedge y \in \text{proj}^+(C^I)\}, \{x \mid \forall y, (x, y) \in \text{Sim}_\Sigma^I \rightarrow y \in \text{proj}^-(C^I)\}\}$ ,
- $(\underline{C}^{\text{Sim}_\Sigma})^I = \{\{x \mid \forall y, (x, y) \in \text{Sim}_\Sigma^I \rightarrow y \in \text{proj}^+(C^I)\}, \{x \mid \exists y, (x, y) \in \text{Sim}_\Sigma^I \wedge y \in \text{proj}^-(C^I)\}\}$ .

Furthermore, due to the possibility of representation of a different notion of uncertainty (contradictory information), we can also develop different similarity relations between individuals. In particular, we will work here with a specific similarity relation described in [15]. But first, we need to make a little change in the definition of the projection function described before to adapt it to the four-valued interpretation.

**Definition 15. (Four-valued projection)** Let  $\mathcal{K}$  be a knowledge base and  $A$  an atomic concept. The projection function  $\pi_A^{\mathcal{K}} : N_I \rightarrow \{\mathbf{t}, \mathbf{f}, \mathbf{u}, \mathbf{i}\}$  is defined as

$$\forall a \in N_I : \pi_A^{\mathcal{K}}(a) = \begin{cases} \mathbf{t}, & \mathcal{K} \models A(a) \text{ and } \mathcal{K} \not\models \neg A(a); \\ \mathbf{f}, & \mathcal{K} \models \neg A(a) \text{ and } \mathcal{K} \not\models A(a); \\ \mathbf{u}, & \mathcal{K} \not\models A(a) \text{ and } \mathcal{K} \not\models \neg A(a); \\ \mathbf{i}, & \mathcal{K} \models \neg A(a) \text{ and } \mathcal{K} \models \neg A(a). \end{cases}$$

where  $N_I$  is the set of individuals contained in the knowledge base  $\mathcal{K}$ .

Now the uncertainty of a concept can be modeled in two ways: as a contradiction or as an unknown information. We can define then the similarity relation  $P_\Sigma$ :

**Definition 16. (Similarity relation—unknown and inconsistent concepts)** Let  $\Sigma = \{A_1, \dots, A_n\}$  be a context. The similarity relation  $P_\Sigma$  induced by  $\Sigma$  is defined as follows:

$$P_\Sigma = \{(a, b) \in N_I \times N_I \mid \text{for all } A_i \text{ in } \Sigma \text{ in which } i \in \{1, \dots, n\}, \pi_{A_i}^{\mathcal{K}}(a) = \pi_{A_i}^{\mathcal{K}}(b) \text{ or}$$

$$\begin{aligned} & \pi_{A_i}^{\mathcal{K}}(a) = \mathbf{u} \text{ or } \pi_{A_i}^{\mathcal{K}}(b) = \mathbf{u} \\ & \text{or if } \pi_{A_i}^{\mathcal{K}}(a) = \mathbf{t} \text{ then } \pi_{A_i}^{\mathcal{K}}(b) = \mathbf{i} \\ & \text{or if } \pi_{A_i}^{\mathcal{K}}(a) = \mathbf{f} \text{ then } \pi_{A_i}^{\mathcal{K}}(b) = \mathbf{i}\}. \end{aligned}$$

In  $P_\Sigma$ , it is assumed that an information can be partially described because of our incomplete or contradictory knowledge [18]. From this point of view, an element  $a$  can be considered similar to the element  $b$  if the information contained in  $a$  is also contained in  $b$ . Thus, for a concept  $A$ , such that  $\pi_A^{\mathcal{K}}(a) = \mathbf{t}$  and  $\pi_A^{\mathcal{K}}(b) = \mathbf{i}$ , the individual  $a$  is similar to  $b$  because the truth value  $\mathbf{t}$  is contained in  $\mathbf{i}$ . Note that the reverse is not true:  $b$  is not similar to  $a$  according to  $P_\Sigma$ , because not every information in  $b$  is contained in  $a$ . We highlight that these similarity relations introduced in this work are two-valued, but nothing prevents one to create four-valued similarity relations.

*Example.* [4] (Query relaxation/restriction) Let  $\{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$  be a set of individuals representing houses;  $GL = \text{GoodLocation}$ ,  $B = \text{Basement}$ ,  $F = \text{Fireplace}$ ,  $E = \text{Expensive}$ ,  $C = \text{Cheap}$ , and  $M = \text{Medium}$  be concepts; and  $\Sigma = \{GL, B, F\}$  be a context and  $\mathcal{A}$  be a ABox such that

$$\begin{aligned} & GL(x_1); \neg GL(x_2); GL(x_3); \neg GL(x_4); GL(x_6); \neg GL(x_7); \\ & B(x_1); B(x_2); \neg B(x_2); \neg B(x_3); B(x_4); B(x_6); \neg B(x_6); B(x_7); \\ & F(x_1); \neg F(x_2); \neg F(x_4); F(x_5); F(x_6); F(x_7); \\ & \neg M(x_1); \neg M(x_2); M(x_3); M(x_4); M(x_5); \neg M(x_6); M(x_7); \\ & E(x_1); \neg E(x_2); \neg E(x_3); \neg E(x_4); \neg E(x_5); E(x_6); \neg E(x_7); \\ & \neg C(x_1); C(x_2); \neg C(x_3); \neg C(x_4); \neg C(x_5); \neg C(x_6); \neg C(x_7). \end{aligned}$$

First, we will consider an example using query relaxation. Suppose that we want to know which houses are expensive. We have that

$$\begin{aligned} & \mathcal{A} \models E(x_1), \mathcal{A} \not\models E(x_2), \mathcal{A} \not\models E(x_3), \mathcal{A} \not\models E(x_4), \\ & \mathcal{A} \not\models E(x_5) \text{ and } \mathcal{A} \models E(x_6). \end{aligned}$$

This means that  $x_1$  and  $x_6$  are the only expensive houses. But suppose that we want to know which houses are possibly expensive (houses that are not expensive but share some features of expensive houses) according to the

context  $\Sigma$ . Relaxing this query (we will use the similarity relation  $S_\Sigma$  in this example), we shall have

$$\mathcal{A} \models \overline{E}^{S_\Sigma}(x_1), \mathcal{A} \models \overline{E}^{S_\Sigma}(x_5) \text{ and } \mathcal{A} \models \overline{E}^{S_\Sigma}(x_6).$$

Thus,  $x_1$ ,  $x_5$ , and  $x_6$  are possibly expensive houses. Observe that  $x_5$  is possibly expensive because it is similar to  $x_1$ , which is evaluated as expensive. By employing again the query relaxation in this ABox, we shall have

$$\mathcal{A} \models \overline{E}^{S_\Sigma}(x_1), \mathcal{A} \models \overline{E}^{S_\Sigma}(x_3), \mathcal{A} \models \overline{E}^{S_\Sigma}(x_5) \text{ and } \mathcal{A} \models \overline{E}^{S_\Sigma}(x_6).$$

We have now that  $x_3$  is possibly a possible expensive house, because  $x_3$  is similar to  $x_5$  according to the relation  $S_\Sigma$ , i.e.,  $x_3$  has less features of expensive houses than  $x_5$ . We will show in the sequel another example related with query refinement but now using query restriction: suppose that we want to know which houses in the ABox have medium value. We shall have then

$$\mathcal{A} \not\models M(x_1), \mathcal{A} \not\models M(x_2), \mathcal{A} \models M(x_3), \mathcal{A} \models M(x_4) \text{ and } \mathcal{A} \models M(x_5).$$

That is,  $x_3$ ,  $x_4$ , and  $x_5$  are houses of medium value. Using the query restriction with the context  $\Sigma$ , we will conclude that

$$\mathcal{A} \models \underline{M}_{S_\Sigma}(x_3), \text{ but } \mathcal{A} \not\models \underline{M}_{S_\Sigma}(x_4) \text{ and } \mathcal{A} \not\models \underline{M}_{S_\Sigma}(x_5).$$

The individuals  $x_4$  and  $x_5$  do not have necessarily a medium value. If we apply the query restriction again, we will observe that

$$\mathcal{A} \not\models \underline{M}_{S_\Sigma}(x_3).$$

Therefore,  $x_3$  does not have necessarily a necessary medium value, i.e.,  $x_3$  is similar to some house that necessarily does not have medium value. Focusing a little in the similarity relation for unknown and inconsistent information, we can show that

$$\mathcal{A} \not\models C(x_4) \text{ and } \mathcal{A} \not\models \overline{C}^{S_\Sigma}(x_4), \text{ but } \mathcal{A} \models \overline{C}^{P_\Sigma}(x_4).$$

This means that  $P_\Sigma$  can be used to discover which individuals share connections with contradictory information in the context  $\Sigma$ . Knowing that  $\mathcal{A} \not\models \overline{C}^{S_\Sigma}(x_4)$  and  $\mathcal{A} \models \overline{C}^{P_\Sigma}(x_4)$ , we can infer that by accepting the presence of similarities with contradictions in  $\Sigma$ ,  $x_4$  can be viewed as a possibly cheap object. A resembling intuition may be used for the lower approximation after searching those individuals that certainly have a specific property when pointed relations with contradictions. For example, with

$$\mathcal{A} \models \underline{M}_{S_\Sigma}(x_3) \text{ and } \mathcal{A} \not\models \underline{M}_{P_\Sigma}(x_3),$$

we can conclude if we permit similarity relations with contradictions that  $x_3$  will not be considered as a medium

value house. In regarding the individual  $x_7$ , the result will be

$$\mathcal{A} \models \underline{M}_{S_\Sigma}(x_7) \text{ and } \mathcal{A} \models \underline{M}_{P_\Sigma}(x_7).$$

This shows that independently of analyzing the similarities with contradictions or not, the result will be the same. In other words, the presence of similarities with contradictory information for  $x_7$  does not exist.  $\square$

## Methods

Contextual approximations were designed to optimize the automation of approximate reasoning, since the relations between individuals are discovered during the reasoning process. However, if we think about an automated query refinement process, the possibilities of generating all contexts are  $2^{|N_C|} - 1$ , where  $N_C$  is the set of atomic concepts. Moreover, most of these contexts can be redundant or cannot satisfy a query refinement. In order to avoid this problem, in this section, we present a method based on the notions of discernibility and indiscernibility matrices [19] to compute contexts for lower and upper approximations.

### Using approximations

The main problem found in query refinements with rough sets is to determine a set of concepts which satisfy a restriction (lower approximation) or a relaxation (upper approximation) of a concept. The following results will help us to discover these appropriated sets.

**Lemma 1. (Generalized monotonicity)** [20] *Given two contexts  $\Sigma_1$  and  $\Sigma_2$ , such that  $\Sigma_1 \subseteq \Sigma_2$ , the following equations hold for all concept expressions  $C$ , an interpretation  $I$ , and a similarity relation  $\text{Sim} \in \{R, S, P\}$ :*

$$\left(\underline{C}_{\text{Sim}_{\Sigma_1}}\right)^I \subseteq \left(\underline{C}_{\text{Sim}_{\Sigma_2}}\right)^I \text{ and } \left(\overline{C}^{\text{Sim}_{\Sigma_2}}\right)^I \subseteq \left(\overline{C}^{\text{Sim}_{\Sigma_1}}\right)^I.$$

Intuitively, Lemma 1 states that by increasing the size of the context, the size of the interpretation of the concepts increases for the lower approximation and decreases for the upper approximation. Therefore, in order to find a context to satisfy the lower approximation of a concept  $C$ , only those minimal satisfying  $C$  are needed, since all their supersets will also satisfy  $C$ . Analogously, in order to find contexts satisfying the upper approximation of  $C$ , only the maximal ones satisfying  $C$  will suffice. Finally, for loose and tight approximations, the following statements hold:

**Proposition 1.** [20] *Given a context  $\Sigma$ , a concept  $C$ , an interpretation  $I$ , and a natural number  $n$ , it holds that*

$$\begin{aligned} \left(\overline{C}^{(R_\Sigma, n)}\right)^I &= \left(\overline{C}^{(R_\Sigma, n+1)}\right)^I \text{ and } \left(\underline{C}_{(R_\Sigma, n)}\right)^I = \left(\underline{C}_{(R_\Sigma, n+1)}\right)^I; \\ \left(\overline{C}^{(S_\Sigma, n)}\right)^I &\subseteq \left(\overline{C}^{(S_\Sigma, n+1)}\right)^I \text{ and } \left(\underline{C}_{(S_\Sigma, n+1)}\right)^I \subseteq \left(\underline{C}_{(S_\Sigma, n)}\right)^I; \\ \left(\overline{C}^{(P_\Sigma, n)}\right)^I &\subseteq \left(\overline{C}^{(P_\Sigma, n+1)}\right)^I \text{ and } \left(\underline{C}_{(P_\Sigma, n+1)}\right)^I \subseteq \left(\underline{C}_{(P_\Sigma, n)}\right)^I. \end{aligned}$$

Loose upper approximations can be applied when there are no contexts which satisfy the upper approximation of a concept. In other words, a similarity relation of a higher step can be used to find a possible context. Similarly, a tight lower approximation can be applied to discover a set of concepts reinforcing the lower approximation, i.e., contexts which preserve the lower approximation in a similarity relation of a higher step. Note that the result for loose upper approximation does not change for the indiscernibility relation, since it is transitive and does not increase the size of the interpretation when it is applied successively (or does not decrease when the tight lower approximation is considered).

### Contexts for lower approximations

First, we consider the problem of searching for an adequate context for query restriction. This problem can be formulated as follows:

- *Input*: The set of concept names  $N_C$ , an ABox  $\mathcal{A}$ , a similarity relation  $\text{Sim}$ , a concept  $C$ , and an individual  $a$ .
- *Output*: A nonempty context  $\Sigma \subseteq N_C - \text{atom}(C)$  and a natural number  $n \geq 1$  such that  $\mathcal{A} \models \underline{C}_{(\text{Sim}_\Sigma, n)}(a)$ .

The function  $\text{atom}(C)$  returns the set of atomic concept names contained in the concept  $C$ . We emphasize that in this paper the problem of query refinement is restricted only to ABoxes and atomic concept assertions (i.e., we consider that empty TBox and complex concepts are not allowed in the ABox). One of the problems related to applications of rough set methods is whether the whole set of attributes (concepts) is necessary and, if not, how to determine the simplified and still sufficient subset of attributes that preserves the distinguishability information of the original one, called reduct. For a knowledge base in  $\mathcal{ALC}$ , the reducts are determined by the minimal sets of concepts that preserve discernibility of all individuals from one another. A resulting reduct is, therefore, a minimal set of concepts enabling one to introduce the same indiscernibility on the universe as the whole set of concepts does.

In the rough set theory, the computation of all types of reducts is based on discernibility matrices [19]. Such matrices are constructed from the discernibility relation. In this paper, we consider dissimilarity instead [21], because we are working with the notion of similarity. We highlight that a dissimilarity relation can be viewed as the complement of a similarity relation.

**Definition 17. (Dissimilarity matrix)** Let  $N_I$  be the set of individuals,  $x, y \in N_I$ ,  $\Sigma$  be a context, and  $\text{Sim}$  be a similarity relation. A dissimilarity matrix is defined as

$$\text{DIS}(\Sigma, x, y, \text{Sim}) = \{A_i \in \Sigma \mid y \notin \text{Sim}_{\{A_i\}}(x)\}, \text{ such that}$$

$$\text{Sim}_{\{A_i\}}(x) = \{y \in N_I \mid (x, y) \in \text{Sim}_{\{A_i\}}\}.$$

Intuitively,  $\text{DIS}(\Sigma, x, y, \text{Sim})$  describes the set of all concepts in  $\Sigma$  in which the individual  $x$  is not similar to  $y$  with respect to  $\text{Sim}$ . To evaluate this set of concepts, we will define a Boolean function  $f(\Sigma, x, y, \text{Sim})$ , called dissimilarity function, that will return the set of reducts.

**Definition 18. (Dissimilarity function)** The dissimilarity function of an individual  $x$  with respect to a context  $\Sigma$ , a concept  $C$ , and a similarity relation  $\text{Sim}$  is defined by

$$f(\Sigma, C, x, \text{Sim}) = \bigwedge_{y \in N_I, \mathcal{A} \models C(x) \Leftrightarrow \mathcal{A} \not\models C(y)} \bigvee \text{DIS}(\Sigma, x, y, \text{Sim}).$$

By Lemma 1, the intersection of all dissimilarities of the individual  $x$  is calculated, since we are interested only in minimal contexts. In order to find a context for an assertion axiom of the form  $A(x)$ , the function  $f(N_C - \{A\}, A, x, \text{Sim})$  can be used. From the point of view of rough sets,  $A$  is not considered in the context because it is treated as the decision attribute and  $N_C - \{A\}$  as the conditional attributes.

*Example.* [4] Consider the ABox  $\mathcal{A}$  below representing houses in which  $\{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$  are individuals,  $GL = \text{GoodLocation}$ ,  $B = \text{Basement}$ ,  $F = \text{FirePlace}$ , and  $M = \text{Medium}$ .

$$\begin{aligned} &GL(x_1); \neg GL(x_2); GL(x_3); \neg GL(x_4); GL(x_6); \neg GL(x_7); \\ &B(x_1); B(x_2); \neg B(x_2); \neg B(x_3); B(x_4); B(x_6); \neg B(x_6); B(x_7); \\ &F(x_1); \neg F(x_2); \neg F(x_4); F(x_5); F(x_6); F(x_7); \\ &\neg M(x_1); \neg M(x_2); M(x_3); M(x_4); M(x_5); \neg M(x_6); M(x_7). \end{aligned}$$

A result obtained here is that  $\mathcal{A} \models M(x_3)$ . In order to know if  $x_3$  necessarily has the property of medium value, we can apply  $f(\Sigma, M, x_3, S)$ , in which  $\Sigma = \{GL, B, F\}$ :

$$\begin{aligned} f(\Sigma, M, x_3, S) &= \bigwedge_{y \in N_I, \mathcal{A} \models M(x_3) \Leftrightarrow \mathcal{A} \not\models M(y)} \left( \bigvee \text{DIS}(\Sigma, x_3, y, S) \right). \\ &= \left( \bigvee \text{DIS}(\Sigma, x_3, x_1, S) \right) \wedge \left( \bigvee \text{DIS}(\Sigma, x_3, x_2, S) \right) \wedge \\ &\quad \left( \bigvee \text{DIS}(\Sigma, x_3, x_6, S) \right). \\ &= B \wedge (GL \vee B) \wedge B. \\ &= B. \end{aligned}$$

It follows that  $\mathcal{A} \models \underline{M}_{(\Sigma_1, 1)}(x_3)$ , where  $\Sigma_1 = \{B\}$ . This result shows that the context  $\{B\}$  is already enough to be used in the lower approximation of *Medium*. In fact, we have that  $\mathcal{A} \models \underline{M}_{(\Sigma, 1)}(x_3)$  is also satisfied, since, as it was mentioned in Lemma 1,  $\Sigma_1 \subseteq \Sigma$  implies  $\underline{M}_{\Sigma_1}(x_3) \subseteq$



$\underline{M}_{\Sigma}(x_3)$ . If we consult the dissimilarity function with relation to R, we will obtain

$$\begin{aligned} f(\Sigma, M, x_3, R) &= \bigwedge_{y \in N_I, \mathcal{A} \models M(x_3) \Leftrightarrow \mathcal{A} \not\models M(y)} \left( \bigvee \text{DIS}(\Sigma, x_3, y, R) \right). \\ &= \left( \bigvee \text{DIS}(\Sigma, x_3, x_1, R) \right) \wedge \left( \bigvee \text{DIS}(\Sigma, x_3, x_2, R) \right) \wedge \\ &\quad \left( \bigvee \text{DIS}(\Sigma, x_3, x_6, R) \right). \\ &= (B \vee F) \wedge (GL \vee B \vee F) \wedge (B \vee F). \\ &= (B \vee F). \end{aligned}$$

The context  $\Sigma_2 = \{B, F\}$  satisfies  $\mathcal{A} \models \underline{M}_{(R\Sigma_2, 1)}(x_3)$ . Comparing with the similarity relation S, we can note that the context  $\Sigma_2$  exhibits similarities with unknown information, since it does not appear in  $f(\Sigma, M, x_3, S)$ , but it is found in  $f(\Sigma, M, x_3, R)$ . By applying the dissimilarity function with the relation P we will obtain

$$\begin{aligned} f(\Sigma, M, x_3, P) &= \bigwedge_{y \in N_I, \mathcal{A} \models M(x_3) \Leftrightarrow \mathcal{A} \not\models M(y)} \left( \bigvee \text{DIS}(\Sigma, x_3, y, P) \right). \\ &= \left( \bigvee \text{DIS}(\Sigma, x_3, x_1, P) \right) \wedge \left( \bigvee \text{DIS}(\Sigma, x_3, x_2, P) \right) \wedge \\ &\quad \text{quad} \left( \bigvee \text{DIS}(\Sigma, x_3, x_6, P) \right). \\ &= B \wedge GL \wedge \emptyset \\ &= \emptyset. \end{aligned}$$

This empty result shows that context satisfying the lower approximation of *Medium* to  $x_3$  and the similarity P does not exist. Intuitively, we can say that when we permit similarities of the values of  $x_3$ , with contradictory information, there will be an individual similar to  $x_3$  that will not satisfy the concept *Medium*—in this case, the individual  $x_6$ .  $\square$

### Contexts for upper approximations

The second problem can be formulated as follows:

- *Input*: The set of concept names  $N_C$ , an ABox  $\mathcal{A}$ , a similarity relation Sim, a concept  $C$ , and an individual  $a$ .
- *Output*: A nonempty context  $\Sigma \subseteq N_C - \text{atom}(C)$  and a natural number  $n \geq 1$  such that  $\mathcal{A} \models \overline{C}^{(\text{Sim}_{\Sigma}, n)}(a)$ .

Unlike the lower approximation, we will consider now the notion of similarity matrix instead of dissimilarity matrix. The motivation behind the search for contexts to an upper approximation comes from the following idea: when an assertion axiom is not satisfied by a knowledge base, we need to find individuals satisfying the assertion of the consulted concept that share some similarities with the individual of the original query. These similarities will characterize the context. Thus, we can calculate the set of maximal concepts preserving similarity.

**Definition 19. (Similarity matrix)** Let  $N_I$  be the set of individuals,  $x, y \in N_I$ ,  $\Sigma$  a context, and  $\text{Sim}$  a similarity relation. A similarity matrix is defined as

$$\begin{aligned} \text{SIM}(\Sigma, x, y, \text{Sim}) &= \{A_i \in \Sigma \mid y \in \text{Sim}_{\{A_i\}}(x)\}, \text{ such that} \\ \text{Sim}_{\{A_i\}}(x) &= \{y \in N_I \mid (x, y) \in \text{Sim}_{\{A_i\}}\}. \end{aligned}$$

The matrix  $\text{SIM}(\Sigma, x, y, \text{Sim})$  describes the set of all concepts in  $\Sigma$  in which an individual  $x$  is similar to  $y$  with respect to Sim. To evaluate this set of concepts, we define the Boolean function  $g(\Sigma, x, y, \text{Sim})$ , called similarity function.

**Definition 20. (Similarity function)** The similarity function of an individual  $x$  with respect to a context  $\Sigma$ , a concept  $C$ , and a similarity relation Sim is defined by

$$g(\Sigma, C, x, \text{Sim}) = \bigvee_{y \in N_I, \mathcal{A} \models C(x) \Leftrightarrow \mathcal{A} \models C(y)} \bigwedge \text{SIM}(\Sigma, x, y, \text{Sim}).$$

Now, we are not interested in finding the minimal set of concepts (reduct). By Lemma 1, we are searching for the maximal set of concepts, so the disjunction of all similarities of the individual  $x$  is performed.

*Example.* [4] Consider now the ABox  $\mathcal{A}'$  below, where  $\{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}$  are individuals,  $GL = \text{GoodLocation}$ ,  $B = \text{Basement}$ ,  $F = \text{FirePlace}$ , and  $E = \text{Expensive}$ .

$$\begin{aligned} &GL(x_1); \neg GL(x_2); GL(x_3); \neg GL(x_4); GL(x_6); \neg GL(x_7); \\ &B(x_1); B(x_2); \neg B(x_2); \neg B(x_3); B(x_4); B(x_6); \neg B(x_6); B(x_7); \\ &F(x_1); \neg F(x_2); \neg F(x_4); F(x_5); F(x_6); F(x_7); \\ &\neg E(x_1); \neg E(x_2); \neg E(x_3); \neg E(x_4); \neg E(x_5); E(x_6); \neg E(x_7). \end{aligned}$$

Consequently,  $\mathcal{A}' \not\models E(x_7)$ . We can apply the function  $g(\Sigma, E, x_7, S)$ , in which  $\Sigma = N_C - \{E\}$ , to discover if there exists a context satisfying the upper approximation of *Expensive* according to  $x_7$ :

$$\begin{aligned} g(\Sigma, E, x_7, S) &= \bigvee_{y \in N_I, \mathcal{A}' \models E(x_7) \Leftrightarrow \mathcal{A}' \models E(y)} \left( \bigwedge \text{SIM}(\Sigma, x_7, y, S) \right). \\ &= \left( \bigwedge \text{SIM}(\Sigma, x_7, x_1, S) \right) \vee \left( \bigwedge \text{SIM}(\Sigma, x_7, x_6, S) \right). \\ &= (B \wedge F) \vee F. \\ &= (B \wedge F). \end{aligned}$$

Therefore, we have  $\mathcal{A}' \models \overline{E}^{(S\Sigma_1, 1)}(x_7)$ , where  $\Sigma_1 = \{B, F\}$ . We have chosen  $(B \wedge F)$  as the simplification of  $((B \wedge F) \vee F)$ , because, as stated in Lemma 1, all nonempty subsets of  $\{B, F\}$  also satisfy the upper approximation of  $E(x_7)$ . We consider  $\Sigma_1$  an optimized context as it covers a greater number of concepts for the upper approximation.

Regarding the similarity relation R, we have the following outcome:

$$\begin{aligned} g(\Sigma, E, x_7, R) &= \bigvee_{y \in N_I, \mathcal{A}' \neq E(x_7) \Leftrightarrow \mathcal{A}' = E(y)} \left( \bigwedge \text{SIM}(\Sigma, x_7, y, R) \right). \\ &= \left( \bigwedge \text{SIM}(\Sigma, x_7, x_1, R) \right) \vee \left( \bigwedge \text{SIM}(\Sigma, x_7, x_6, R) \right). \\ &= (B \wedge F) \vee F. \\ &= (B \wedge F). \end{aligned}$$

In this example, we obtained the same results to the relations R and S. In this case, we can conclude that unknown information related to the individual  $x_7$  does not exist. For the similarity function with the relation P, we will have

$$\begin{aligned} g(\Sigma, E, x_7, P) &= \bigvee_{y \in N_I, \mathcal{A}' \neq E(x_7) \Leftrightarrow \mathcal{A}' = \text{Caro}(y)} \left( \bigwedge \text{SIM}(\Sigma, x_7, y, P) \right). \\ &= \left( \bigwedge \text{SIM}(\Sigma, x_7, x_1, P) \right) \vee \left( \bigwedge \text{SIM}(\Sigma, x_7, x_6, P) \right). \\ &= (B \wedge F) \vee (B \wedge F). \\ &= (B \wedge F). \end{aligned}$$

The outcome is the same obtained with the relations R and S. However, differently from them,  $(\bigwedge \text{SIM}(\Sigma, x_7, x_6, P)) = (B \wedge F)$ . This implies that there are some evidence of similarities with contradictions in P, but in this case, this evidence is redundant, because they do not change the final result.  $\square$

## Results and discussion

Now, we present the algorithms to find contexts for query refinements. They consist of searching for minimal sets of concepts (for query restrictions) or maximal sets of concepts (for query relaxations). If no result is found, then the process is repeated taking into account a similarity relation of a higher step. The algorithms finish when any context is found (which will be the answer of the problem) or when they search in all possible  $k$ -step relations and no result is returned (in this case, an empty set will be the answer of the problem). In the sequel, we explain the algorithms and show their complexities.

We assume in the problem that a formula is represented in conjunctive normal form (CNF) (if it is the input of Algorithms 1–2) or in disjunctive normal form (DNF) (if it is the input of Algorithm 3). For instance,  $S = \{\{A_1, A_2\}, \{A_2, A_3\}\}$  can be treated as  $S = (A_1 \wedge A_2) \vee (A_2 \wedge A_3)$  (if it is the input of Algorithms 1–2) or  $S = (A_1 \vee A_2) \wedge (A_2 \vee A_3)$  (if it is the input of Algorithm 3). Algorithm 1 simplifies a formula in DNF by removing redundant clauses. This procedure is done by employing absorption law, i.e.,  $(a \wedge b) \vee a \equiv a$ , which is performed in the function Extract (lines 6 and 10). Only this law is needed to simplify, since the input of the algorithm contains only atomic concepts. The function Extract consists simply in removing a clause from a set of clauses.

---

### Algorithm 1 SimplifyDNF( $S, n$ )

---

```

1:  $i \leftarrow 1$ 
2: while  $i \leq n$  do
3:    $j \leftarrow i + 1$ 
4:   while  $j \leq n$  do
5:     if  $S_i \subseteq S_j$  then
6:        $S \leftarrow \text{Extract}(S, S_j)$ 
7:        $j \leftarrow j + 1$ 
8:     else
9:       if  $S_j \subseteq S_i$  then
10:         $S \leftarrow \text{Extract}(S, S_i)$ 
11:         $i \leftarrow i + 1$ 
12:         $j \leftarrow i + 1$ 
13:      else
14:         $j \leftarrow j + 1$ 
15:      end if
16:    end if
17:  end while
18:   $i \leftarrow i + 1$ 
19: end while
20: return  $S$ 

```

---

The function SimplifyDNF2( $S, n$ ) (Algorithm 2) follows the idea of Algorithm 1, but instead of the absorption law, the rule  $(a \wedge b) \vee a \equiv (a \wedge b)$  to represent maximal contexts is applied to it. Then, SimplifyDNF2( $S, n$ ) is obtained by exchanging the lines 5 and 9 of Algorithm 1.

---

### Algorithm 2 SimplifyDNF2( $S, n$ )

---

```

1:  $i \leftarrow 1$ 
2: while  $i \leq n$  do
3:    $j \leftarrow i + 1$ 
4:   while  $j \leq n$  do
5:     if  $S_j \subseteq S_i$  then
6:        $S \leftarrow \text{Extract}(S, S_j)$ 
7:        $j \leftarrow j + 1$ 
8:     else
9:       if  $S_i \subseteq S_j$  then
10:         $S \leftarrow \text{Extract}(S, S_i)$ 
11:         $i \leftarrow i + 1$ 
12:         $j \leftarrow i + 1$ 
13:      else
14:         $j \leftarrow j + 1$ 
15:      end if
16:    end if
17:  end while
18:   $i \leftarrow i + 1$ 
19: end while
20: return  $S$ 

```

---

**Theorem 1.** *The time complexities of SimplifyDNF( $S, n$ ) and SimplifyDNF2( $S, n$ ) are  $O(|N_C| \cdot n^2)$ , where  $n$  is the number of clauses of the disjunctive normal form formula received as input and  $N_C$  is the set of atomic concepts.*

*Proof.* The Algorithms have two nested loops of the type **while**, which are limited by  $n$  (thus,  $O(n^2)$ ), where  $n$  is the number of clauses in the DNF formula of the input. The complexity of Extract has a linear upper bound in the size of the formula, i.e.,  $O(|N_C|)$ . Therefore, the complexity of the algorithms SimplifyDNF( $S, n$ ) and SimplifyDNF2( $S, n$ ) are  $O(|N_C| \cdot n^2)$ .  $\square$

**Theorem 2.** *The time complexity of SimplifyCNF( $S, n$ ) is  $O(2^{|N_C|})$ .*

*Proof.* Algorithm 3 simplifies a formula  $S$  in CNF by removing redundant clauses, similarly to Algorithm 1. This is achieved by translating a CNF formula into a DNF formula via distribution rules (Distribute) and then applying the function SimplifyDNF. The complexity of the algorithm which translates a CNF to a DNF formula is  $O(2^n)$  [22], where  $n$  is the number of different variables of the CNF formula. The complexity of line 1 is  $O(2^{|N_C|})$ , in which  $|N_C|$  is the number of atomic concepts and the limit of variables that can occur in the CNF formula. The complexity of line 2 is  $O(|N_C| \cdot n^2)$ , as showed above. Therefore, the complexity of Algorithm 3 is  $O(2^{|N_C|})$ .  $\square$

---

### Algorithm 3 SimplifyCNF( $S, n$ )

---

- 1:  $A \leftarrow \text{Distribute}(S)$
  - 2: **return** SimplifyDNF( $S, n$ )
- 

Algorithm 4 searches sets of contexts for lower approximations. In other words, it implements the dissimilarity function  $f(\Sigma, C, x, \text{Sim})$ . First, the dissimilarity matrix of a specific individual with respect to the similarity relation Sim (lines 8–18) is computed. After, the dissimilarity function through SimplifyCNF (line 19) is calculated.

If the result is nonempty, then it will be the solution of the problem. Otherwise, the procedure is restarted with a similarity relation of a higher step. The result of the problem is found by finding a  $k$ -step relation returning a nonempty set. If all  $k$ -step relations return empty sets, then the solution will be empty (i.e., there are no contexts to solve the problem).

For instance, taking an example of the previous subsection,  $f(\{GL, B, F\}, M, x_3, S)$  will find the contexts  $B, (GL \vee B)$ , and  $B$  in the first iteration through the dissimilarity matrix. Then, the context  $B$  is concluded after a

simplification. As the result is nonempty, this will be the result of the problem.

**Theorem 3.** *Determining if there exist contexts for lower approximations of an ABox assertion in paraconsistent rough  $\mathcal{ALC}$  is EXPTIME.*

*Proof.* By Algorithm 4, the complexity of the loop **for all** (lines 8–18) takes into account the complexity of logical consequence of an assertion axiom in  $\mathcal{PR}_{\mathcal{ALC}}$  and the computation of the dissimilarity matrix DIS. By definition, DIS also depends on the problem of logical consequence in  $\mathcal{PR}_{\mathcal{ALC}}$ , i.e., its complexity is PSPACE. Thus, the complexity of the lines 8–18 takes polynomial space. As pointed in Proposition 1, *tight* and *loose* approximations are monotonic, and since we are leading with finite sets of individuals, there is a finite number of  $k$ -step relations, which are bounded by  $|N_I|$ . Therefore, the maximum number of tests done through the **while** (line 5) is  $O(|N_I|)$ . Lastly, we have that SimplifyCNF( $S, n$ ) has a time complexity of  $O(2^{|N_C|})$ .  $\square$

Algorithm 5 follows the rationale of Algorithm 4, but it constructs the similarity matrix SIM. After that, the algorithm computes the similarity function through SimplifyDNF2. Considering the example  $g(\{GL, B, F\}, E, x_7, R)$  from the previous subsection, the algorithm will discover the contexts  $(B \wedge F)$  and  $F$  in the first iteration. Then, the simplification will result in  $(B \wedge F)$  that will be the result of the query relaxation.

---

### Algorithm 4 $f(N_C, C, x, \text{Sim})$

---

- 1:  $j \leftarrow 1$
  - 2:  $\Sigma_1 \leftarrow \emptyset$
  - 3:  $\Sigma \leftarrow N_C - \{C\}$
  - 4:  $\text{Sim}_{0\Sigma} \leftarrow \emptyset$
  - 5: **while**  $\Sigma_1 = \emptyset$  e  $\text{Sim}_{j-1\Sigma} \neq \text{Sim}_{j\Sigma}$  **do**
  - 6:      $S \leftarrow \emptyset$
  - 7:      $i \leftarrow 1$
  - 8:     **for all**  $y \in N_I$  **do**
  - 9:         **if**  $\mathcal{A} \models C(x) \Leftrightarrow \mathcal{A} \not\models C(y)$  **then**
  - 10:              $S_i \leftarrow \text{DIS}(N_C - \{C\}, x, y, \text{Sim}_j)$
  - 11:             **if**  $S_i = \emptyset$  **then**
  - 12:                 **return**  $\emptyset$
  - 13:             **else**
  - 14:                  $S \leftarrow S \cup \{S_i\}$
  - 15:                  $i \leftarrow i + 1$
  - 16:             **end if**
  - 17:     **end if**
  - 18:     **end for**
  - 19:      $\Sigma_1 \leftarrow \text{SimplifyCNF}(S, i - 1)$
  - 20:      $j \leftarrow j + 1$
  - 21: **end while**
  - 22: **return**  $\Sigma_1, j - 1$
-

**Algorithm 5**  $g(N_C, C, x, \text{Sim})$ 


---

```

1:  $j \leftarrow 1$ 
2:  $\Sigma_1 \leftarrow \emptyset$ 
3:  $\Sigma \leftarrow N_C - \{C\}$ 
4:  $\text{Sim}_{0\Sigma} \leftarrow \emptyset$ 
5: while  $\Sigma_1 = \emptyset \wedge \text{Sim}_{j-1\Sigma} \neq \text{Sim}_{j\Sigma}$  do
6:    $S \leftarrow \emptyset$ 
7:    $i \leftarrow 1$ 
8:   for all  $y \in N_I$  do
9:     if  $\mathcal{A} \not\models C(x) \Leftrightarrow \mathcal{A} \models C(y)$  then
10:       $S_i \leftarrow \text{SIM}(N_C - \{C\}, x, y, \text{Sim}_j)$ 
11:      if  $S_i \neq \emptyset$  then
12:         $S \leftarrow S \cup \{S_i\}$ 
13:         $i \leftarrow i + 1$ 
14:      end if
15:    end if
16:  end for
17:   $\Sigma_1 \leftarrow \text{SimplifyDNF2}(S, i - 1)$ 
18:   $j \leftarrow j + 1$ 
19: end while
20: return  $\Sigma_1, j - 1$ 

```

---

**Theorem 4.** *Determining if there exist contexts for upper approximations of an ABox assertion in paraconsistent rough  $\mathcal{ALC}$  is PSPACE.*

*Proof.* As in Algorithm 4, the construction of the similarity matrix SIM takes into account the logical consequence problem in  $\mathcal{PR}_{\mathcal{ALC}}$  (lines 8–14) and has polynomial space complexity. In the same way, the number of  $k$ -step relations is bounded by  $O(|N_I|)$  (line 5). The time complexity of  $\text{SimplifyDNF2}(S, n)$  is polynomial, but the overall complexity of the algorithm is PSPACE for  $\mathcal{PR}_{\mathcal{ALC}}$ .  $\square$

## Conclusions

In this paper, we defined techniques for handling query refinements of assertion axioms in  $\mathcal{PR}_{\mathcal{ALC}}$ , by employing the notion of contextual approximation and similarity relations, which exploited the presence of unknown and inconsistent information. We also showed a method to compute optimized contexts for these queries based on the notion of reducts presented in the rough set theory. The problem of query restrictions using contextual approximation was proved to have exponential time complexity, while the problem of query relaxations has polynomial space complexity.

As future work, we will investigate some ways of choosing the most representative contexts, as approaches involving measures of inconsistency or information, since the method presented in this paper concerns only about minimal or maximal contexts. Some approaches to deal

with this problem can be found in [21, 23]. Another possible line of research is to extend the application of query refinement in complex assertion axioms (any concept  $C(a)$ ) as well to terminological axioms, i.e., axioms of the form  $C \sqsubseteq D$ .

## Competing interests

The authors declare that they have no competing interests.

## Authors' contributions

This work is part of HV Master thesis, JA was his advisor and ATM was his co-advisor. HV carried out most of the investigation, implementation and writing. JA gave important ideas to the project and also contributed in revision of this manuscript. ATM gave important background about related areas and also contributed in revision of this manuscript. All authors read and approved the final manuscript.

## Acknowledgements

This research is partially supported by CNPq (305980/2013-0,301607/2010-9, 474821/2012-9, 482481/2010-2), CAPES (PROCAD/NF789/2010), CNPq/CAPES(552578/2011-8).

Received: 1 May 2014 Accepted: 18 June 2015

Published online: 07 July 2015

## References

- Pawlak Z (1982) Rough sets. *Int J Inf Comput Sci* 11:341–356
- Schaerf M, Cadoli M (1995) Tractable reasoning via approximation. *Artif Intell* 74:249–310
- Stuckenschmidt H (2007) Partial matchmaking using approximate subsumption. In: *AAAI'07*. AAAI Press, Vancouver, British Columbia, Canada. pp 1459–1464
- Viana H, Alcántara J, Martins AT (2011) Paraconsistent rough description logic. *CEUR-WS.org*, Barcelona, Spain
- Cornelis C, De Cock M, Radzikowska AM (2008) Fuzzy rough sets: from theory into practice. In: *Pedrycz W, Skowron A, Kreinovich V (eds)*. Handbook of Granular Computing. John Wiley & Sons, Inc., New York, NY, USA. pp 533–552
- Fanizzi N, d'Amato C, Esposito F, Lukasiewicz T (2008) Representing uncertain concepts in rough description logics via contextual indiscernibility relations. In: *URSW*, Karlsruhe, Germany. Springer, Berlin Heidelberg
- Viana H, Alcántara J, Martins AT (2013) Searching contexts in rough description logics. In: *Intelligent systems (BRACIS), 2013 Brazilian Conference On Intelligent Systems*. pp 163–168
- Belnap ND (1977) A useful four-valued logic. In: *Dunn JM, Epstein G (eds)*. Modern uses of multiple-valued logic. D. Reidel, Springer Netherlands. pp 8–37
- Peñaloza R, Zou T (2013) Roughening the  $\mathcal{EL}$  envelope. In: *Frontiers of combining systems (FroCoS 2013)*. Springer Berlin Heidelberg. pp 71–86
- Jiang Y, Wang J, Tang S, Xiao B (2009) Reasoning with rough description logics: an approximate concepts approach. *Inf Sci* 179(5):600–612
- Keet CM (2010) On the feasibility of description logic knowledge bases with rough concepts and vague instances. In: *Description logics*, Waterloo, Canada. Citeseer
- Schlobach S, Klein M, Amsterdam VU (2007) Description logics with approximate definitions: precise modeling of vague concepts. In: *IJCAI 07*, Hyderabad, India. pp 557–562
- Baader F (2003) The description logic handbook: theory, implementation, and applications. Cambridge University Press, New York, NY, USA
- Komorowski J, Pawlak Z, Polkowski L, Skowron A (1998) Rough sets: a tutorial. <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.37.2477>
- Grzymala-Busse JW (2006) A rough set approach to data with missing attribute values. In: *RSKT*, Chongqing, China. Springer, Berlin Heidelberg. pp 58–67
- Wu W, Zhang W (2002) Neighborhood operator systems and approximations. *Inf Sci* 144(1-4):201–217

17. Ma Y, Hitzler P, Lin Z (2006) Paraconsistent reasoning with OWL — algorithms and the ParOWL reasoner. Technical report, AIFB, University of Karlsruhe, Germany
18. Grzymala-Busse JW (2006) Rough set strategies to data with missing attribute values. In: Foundations and novel approaches in data mining. Springer, Berlin Heidelberg Vol. 9. pp 197–212
19. Skowron A, Rauszer C (1992) The discernibility matrices and functions in information systems. In: Slowiński R (ed). Handbook of applications and advances of the rough sets theory, Kluwer, Dordrecht
20. Viana H (2012) Refinamento de Consultas em Lógicas de Descrição Utilizando a Teoria dos Rough Sets. [http://mdcc.ufc.br/teses/doc\\_download/183-161-henrique-viana-oliveira](http://mdcc.ufc.br/teses/doc_download/183-161-henrique-viana-oliveira)
21. Stepaniuk J (1998) Approximation spaces, reducts and representatives. In: Polkowski L, Skowron A (eds). Rough sets in knowledge discovery, Part I and II. Physica-Verlag, Heidelberg. pp 109–126
22. Miltersen PB, Radhakrishnan J, Wegener I (2005) On converting CNF to DNF. *Theor Comput Sci* 347(1-2):325–335
23. Nguyen HS (2006) Approximate Boolean reasoning: foundations and applications in data mining. *Trans Rough Sets V* 4100:334–506

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

---

Submit your next manuscript at ► [springeropen.com](http://springeropen.com)

---