Ontology engineering with rough concepts and vague instances

C. Maria Keet

KRDB Research Centre, Free University of Bozen-Bolzano, Italy
keet@inf.unibz.it

KRDB Lunch Seminar, 15 June 2010

Motivations

- Extending the TOG [Kee08] with roughness
- Hypothesis testing with bio-ontologies linked to data [KRM07]:
  - *Which bacteria are promiscuous?*
  - *Which properties are necessary or sufficient to identify and retrieve promiscuous bacteria?*
- DL & vagueness: fuzzy or rough
  - Based on the chosen properties, some instances are *indistinguishable*
  - Do not deal with degree of membership, but with *definitely* or *possibly* being a promiscuous bacterium
  - Hence, roughness
- What about using notions of rough sets at the knowledge representation layer?

Motivations

- Extending the TOG [Kee08] with roughness
- Hypothesis testing with bio-ontologies linked to data [KRM07]:
  - *Which bacteria are promiscuous?*
  - *Which properties are necessary or sufficient to identify and retrieve promiscuous bacteria?*
- DL & vagueness: fuzzy or rough
  - Based on the chosen properties, some instances are *indistinguishable*
  - Do not deal with degree of membership, but with *definitely* or *possibly* being a promiscuous bacterium
  - Hence, roughness
- What about using notions of rough sets at the knowledge representation layer?

Motivations

- Extending the TOG [Kee08] with roughness
- Hypothesis testing with bio-ontologies linked to data [KRM07]:
  - *Which bacteria are promiscuous?*
  - *Which properties are necessary or sufficient to identify and retrieve promiscuous bacteria?*
- DL & vagueness: fuzzy or rough
  - Based on the chosen properties, some instances are *indistinguishable*
  - Do not deal with degree of membership, but with *definitely* or *possibly* being a **promiscuous** bacterium
  - Hence, roughness
- What about using notions of rough sets at the knowledge representation layer?

Motivations

- Extending the TOG [Kee08] with roughness
- Hypothesis testing with bio-ontologies linked to data [KRM07]:
  - *Which bacteria are promiscuous?*
  - *Which properties are necessary or sufficient to identify and retrieve promiscuous bacteria?*
- DL & vagueness: fuzzy or rough
  - Based on the chosen properties, some instances are *indistinguishable*
  - Do not deal with degree of membership, but with *definitely* or *possibly* being a **promiscuous** bacterium
  - Hence, roughness
- What about using notions of rough sets at the knowledge representation layer?

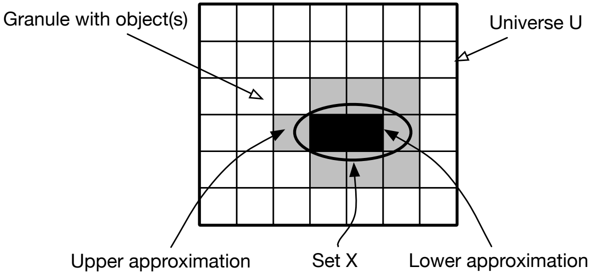
Motivations

- Related works [BS09, FDEL08, IGNI07, JWTX09, Lia96, SKP07]:
  - Diverge in commitment as to which aspects of rough sets are included in the ontology language
  - Theory instead of demonstrating successful use of the rough DL in applications and ontology engineering
- What are the requirements for rough DL knowledge bases to faithfully *represent* the core notions of rough sets and to *implement* it?

Motivations

- Related works [BS09, FDEL08, IGNI07, JWTX09, Lia96, SKP07]:
  - Diverge in commitment as to which aspects of rough sets are included in the ontology language
  - Theory instead of demonstrating successful use of the rough DL in applications and ontology engineering
- What are the requirements for rough DL knowledge bases to faithfully *represent* the core notions of rough sets and to *implement* it?

Pawlak rough set model



Pawlak rough set model

- $I = (U, A)$ is called an information system, U universe of objects
- A set of attributes s.t. for every $a \in A$, $a : U \mapsto V_a$, with V_a the set of values that attribute a can have
- For any subset of attributes $P \subseteq A$, one defines the indistinguishability relation $\text{IND}(P)$ as

$$\text{IND}(P) = \{(x, y) \in U \times U \mid \forall a \in P, a(x) = a(y)\} \quad (1)$$

- $\text{IND}(P)$ generates a partition of U, denoted with U/P
- If $(x, y) \in \text{IND}(P)$, then x and y are indistinguishable w.r.t. the attributes in P, i.e, they are *p-indistinguishable*

Pawlak rough set model

- Represent set X such that $X \subseteq U$ using P (where $P \subseteq A$)
- X can be approximated by using *lower* and *upper approximation*:

$$\underline{P}X = \{x \mid [x]_P \subseteq X\} \quad (2)$$
$$\overline{P}X = \{x \mid [x]_P \cap X \neq \emptyset\} \quad (3)$$

  where $[x]_P$ denotes the equivalence classes of the p-indistinguishability relation
- Lower approximation is the set of objects that are positively classified as being members of set X, i.e., union of all equivalence classes in $[x]_P$
- Upper approximation is the set of objects that are possibly in X
- Its complement, $U - \overline{P}X$, is the negative region with sets of objects that are definitely not in X (i.e., $\neg X$)

Pawlak rough set model

- With every rough set, we associate two *crisp* sets (the *lower* and *upper approximation*), denoted as a tuple $X = \langle \underline{X}, \overline{X} \rangle$
- Boundary region $B_P X = \overline{P}X - \underline{P}X$, where its objects neither can be classified as to be member of X nor that they are not in X
- If $B_P X = \emptyset$ then X is, in fact, a crisp set with respect to P and when $B_P X \neq \emptyset$ then X is rough w.r.t. P.
- Accuracy of approximation, reduct (sufficient conditions), core (necessary conditions), and

$$\underline{P}X \subseteq X \subseteq \overline{P}X \quad (4)$$

Example: second-hand vehicles

	Age	Wheels	Engine	Helmet
o_1	< 5	2	no	no
o_2	> 5	2	no	no
o_3	> 5	2	yes	yes
o_4	> 5	3	yes	yes
o_5	> 5	3	yes	yes
o_6	> 5	3	no	yes
o_7	> 5	3	no	no
o_8	< 5	2	no	no
o_9	< 5	4	yes	no

Induced equivalence classes:

$[\text{one}] = \{o_1, o_8\}$ $[\text{four}] = \{o_4, o_5\}$
$[\text{two}] = \{o_2\}$ $[\text{five}] = \{o_6, o_7\}$
$[\text{three}] = \{o_3\}$ $[\text{six}] = \{o_9\}$

Rough sets for our vehicles

- Assume our target set X has members $\{o_3, o_4, o_5, o_6\}$
- Then [*four*] and [*three*] are *definitely* in out target set, i.e., is the *lower approximation* $\underline{X} = \{o_3, o_4, o_5\}$
- But what about o_6?
- Given [*five*] $= \{o_6, o_7\}$, we cannot distinguish between o_6 and o_7, so [*five*] as such cannot be part of our target set, i.e., with the given (incomplete?) information, there is no way to represent X such that it includes o_6 but excludes o_7
- *Upper approximation* of X is $\overline{X} = \{o_3, o_4, o_5, o_6, o_7\}$
- *Boundary region*: $\{o_6, o_7\}$
- 'heuristic': once there are equivalence classes with more than one object, there likely will be at least one rough set

Rough sets for our vehicles

- Assume our target set X has members $\{o_3, o_4, o_5, o_6\}$
- Then [*four*] and [*three*] are *definitely* in out target set, i.e., is the *lower approximation* $\underline{X} = \{o_3, o_4, o_5\}$
- But what about o_6?
- Given $[five] = \{o_6, o_7\}$, we cannot distinguish between o_6 and o_7, so [*five*] as such cannot be part of our target set, i.e., with the given (incomplete?) information, there is no way to represent X such that it includes o_6 but excludes o_7
- *Upper approximation* of X is $\overline{X} = \{o_3, o_4, o_5, o_6, o_7\}$
- *Boundary region*: $\{o_6, o_7\}$
- 'heuristic': once there are equivalence classes with more than one object, there likely will be at least one rough set

Rough sets for our vehicles

- Assume our target set X has members $\{o_3, o_4, o_5, o_6\}$
- Then [*four*] and [*three*] are *definitely* in out target set, i.e., is the *lower approximation* $\underline{X} = \{o_3, o_4, o_5\}$
- But what about o_6?
- Given [*five*] $= \{o_6, o_7\}$, we cannot distinguish between o_6 and o_7, so [*five*] as such cannot be part of our target set, i.e., with the given (incomplete?) information, there is no way to represent X such that it includes o_6 but excludes o_7
- *Upper approximation* of X is $\overline{X} = \{o_3, o_4, o_5, o_6, o_7\}$
- *Boundary region*: $\{o_6, o_7\}$
- 'heuristic': once there are equivalence classes with more than one object, there likely will be at least one rough set

Rough sets for our vehicles

- Assume our target set X has members $\{o_3, o_4, o_5, o_6\}$
- Then [*four*] and [*three*] are *definitely* in out target set, i.e., is the *lower approximation* $\underline{X} = \{o_3, o_4, o_5\}$
- But what about o_6?
- Given [*five*] $= \{o_6, o_7\}$, we cannot distinguish between o_6 and o_7, so [*five*] as such cannot be part of our target set, i.e., with the given (incomplete?) information, there is no way to represent X such that it includes o_6 but excludes o_7
- *Upper approximation* of X is $\overline{X} = \{o_3, o_4, o_5, o_6, o_7\}$
- *Boundary region*: $\{o_6, o_7\}$
- 'heuristic': once there are equivalence classes with more than one object, there likely will be at least one rough set

Rough sets for our vehicles

- Assume our target set X has members $\{o_3, o_4, o_5, o_6\}$
- Then [*four*] and [*three*] are *definitely* in out target set, i.e., is the *lower approximation* $\underline{X} = \{o_3, o_4, o_5\}$
- But what about o_6?
- Given [*five*] $= \{o_6, o_7\}$, we cannot distinguish between o_6 and o_7, so [*five*] as such cannot be part of our target set, i.e., with the given (incomplete?) information, there is no way to represent X such that it includes o_6 but excludes o_7
- *Upper approximation* of X is $\overline{X} = \{o_3, o_4, o_5, o_6, o_7\}$
- *Boundary region*: $\{o_6, o_7\}$
- 'heuristic': once there are equivalence classes with more than one object, there likely will be at least one rough set

Reduct and core

- Reduct can be considered to be the set of *sufficient* conditions (attributes) to maintain the equivalence class structure induced by P
- Core can be considered to be the set of *necessary* conditions to maintain the equivalence class structure induced by P
- $\textsc{Core} \subseteq \textsc{Red} \subseteq P$ such that $[x]_{\textsc{Red}} = [x]_P$ and $\textsc{Red}$ is minimal for any $a \in \textsc{Red}$ (i.e., $[x]_{\textsc{Red}-\{a\}} \neq [x]_P$)
  - That is, those attributes that are in P but not in $\textsc{Red}$ are superfluous with respect to the partitioning with P
- For any reduct of P, $\textsc{Red}_1, \ldots, \textsc{Red}_n$, the core is its intersection, i.e., $\textsc{Core} = \textsc{Red}_1 \cap \ldots \cap \textsc{Red}_n$
  - No attribute in $\textsc{Core}$ can be removed without destroying the equivalence structure
  - It is possible that $\textsc{Core}$ is an empty set

Reduct and core

- Reduct can be considered to be the set of *sufficient* conditions (attributes) to maintain the equivalence class structure induced by P
- Core can be considered to be the set of *necessary* conditions to maintain the equivalence class structure induced by P
- $\mathrm{CORE} \subseteq \mathrm{RED} \subseteq P$ such that $[x]_{\mathrm{RED}} = [x]_P$ and RED is minimal for any $a \in \mathrm{RED}$ (i.e., $[x]_{\mathrm{RED}-\{a\}} \neq [x]_P$)
  - That is, those attributes that are in P but not in RED are superfluous with respect to the partitioning with P
- For any reduct of P, $\mathrm{RED}_1, \ldots, \mathrm{RED}_n$, the core is its intersection, i.e., $\mathrm{CORE} = \mathrm{RED}_1 \cap \ldots \cap \mathrm{RED}_n$
  - No attribute in CORE can be removed without destroying the equivalence structure
  - It is possible that CORE is an empty set

Reduct and core

- Reduct can be considered to be the set of *sufficient* conditions (attributes) to maintain the equivalence class structure induced by P
- Core can be considered to be the set of *necessary* conditions to maintain the equivalence class structure induced by P
- $\textsc{Core} \subseteq \textsc{Red} \subseteq P$ such that $[x]_{\textsc{Red}} = [x]_P$ and $\textsc{Red}$ is minimal for any $a \in \textsc{Red}$ (i.e., $[x]_{\textsc{Red}-\{a\}} \neq [x]_P$)
  - That is, those attributes that are in P but not in $\textsc{Red}$ are superfluous with respect to the partitioning with P
- For any reduct of P, $\textsc{Red}_1, \ldots, \textsc{Red}_n$, the core is its intersection, i.e., $\textsc{Core} = \textsc{Red}_1 \cap \ldots \cap \textsc{Red}_n$
  - No attribute in $\textsc{Core}$ can be removed without destroying the equivalence structure
  - It is possible that $\textsc{Core}$ is an empty set

From ‘information system’ to DLs

- More constructors and possible constraints in a DL-ontology, notably a set of roles, $\mathcal{R}$, over objects quantification, role properties
- More flexibility on representing ‘attributes’ of a concept $C \in \mathcal{C}$: either with one or more roles $R \in \mathcal{R}$ or value attributions $D \in \mathcal{D}$, or both
- Need for a complete and appropriate model-theoretic semantics for $\underline{C}$ and $\overline{C}$, and the rough concept, denoted with “$\wr C$”
- Given that attributes are used to compute $\underline{C}$ and $\overline{C}$, then those attributes must be represented in the DL KB, and with $\wr C$ a tuple of the former two, then also it must have those attributes

On the semantics of $\wr C$

- Lower and upper approximation:

$$\underline{C} = \{x \mid \forall y : (x,y) \in Ind \rightarrow y \in C\} \qquad (5)$$
$$\overline{C} = \{x \mid \exists y : (x,y) \in Ind \land y \in C\} \qquad (6)$$

- $\wr C = \langle \underline{C}, \overline{C} \rangle$
- Interpretation maps every $\wr C = \langle \underline{C}, \overline{C} \rangle$ to a pair over $\Delta^{\mathcal{I}}$, i.e., extending $\cdot^{\mathcal{I}}$ as follows:

$$\wr C^{\mathcal{I}} = (\langle \underline{C}, \overline{C} \rangle)^{\mathcal{I}} = \langle (\underline{C})^{\mathcal{I}}, (\overline{C})^{\mathcal{I}} \rangle \qquad (7)$$

- Subsumption relation between the sets as in (4) and their corresponding concepts, but does not *define* it
- Make explicit the knowledge about the three sets and how they relate

On the semantics of $\wr C$

- Lower and upper approximation:

$$\underline{C} = \{x \mid \forall y : (x,y) \in \mathit{Ind} \rightarrow y \in C\} \quad (5)$$

$$\overline{C} = \{x \mid \exists y : (x,y) \in \mathit{Ind} \wedge y \in C\} \quad (6)$$

- $\wr C = \langle \underline{C}, \overline{C} \rangle$
- Interpretation maps every $\wr C = \langle \underline{C}, \overline{C} \rangle$ to a pair over $\Delta^{\mathcal{I}}$, i.e., extending $\cdot^{\mathcal{I}}$ as follows:

$$\wr C^{\mathcal{I}} = (\langle \underline{C}, \overline{C} \rangle)^{\mathcal{I}} = \langle (\underline{C})^{\mathcal{I}}, (\overline{C})^{\mathcal{I}} \rangle \quad (7)$$

- Subsumption relation between the sets as in (4) and their corresponding concepts, but does not *define* it
- Make explicit the knowledge about the three sets and how they relate

On the semantics of $\wr C$

- Introduce two binary relationships, *lapr* and *uapr*, to relate *any* rough concept and its associated approximations

$$\forall \phi, \psi . lapr(\phi, \psi) \rightarrow \wr C(\phi) \wedge \underline{C}(\psi) \quad (8)$$
$$\forall \phi, \psi . uapr(\phi, \psi) \rightarrow \wr C(\phi) \wedge \overline{C}(\psi) \quad (9)$$

- Make explicit that $\wr C$ is identified by the combination of its $\underline{C}$ and $\overline{C}$:

$$\begin{aligned}
&\forall \phi . \wr C(\phi) \rightarrow \exists \psi . lapr(\phi, \psi), \\
&\forall \phi . \wr C(\phi) \rightarrow \exists \psi . uapr(\phi, \psi), \\
&\forall \phi, \psi, \varphi . lapr(\phi, \psi) \wedge lapr(\phi, \varphi) \rightarrow \psi = \varphi, \quad (10) \\
&\forall \phi, \psi, \varphi . uapr(\phi, \psi) \wedge uapr(\phi, \varphi) \rightarrow \psi = \varphi, \\
&\forall \phi_1, \phi_2, \psi_1, \psi_2 . lapr(\phi_1, \psi_1) \wedge uapr(\phi_1, \psi_2) \wedge \\
&\quad lapr(\phi_2, \psi_1) \wedge uapr(\phi_2, \psi_2) \rightarrow \phi_1 = \phi_2 .
\end{aligned}$$

On the semantics of $\wr C$

- Introduce two binary relationships, *lapr* and *uapr*, to relate *any* rough concept and its associated approximations

$$\forall \phi, \psi . lapr(\phi, \psi) \rightarrow \wr C(\phi) \wedge \underline{C}(\psi) \quad (8)$$

$$\forall \phi, \psi . uapr(\phi, \psi) \rightarrow \wr C(\phi) \wedge \overline{C}(\psi) \quad (9)$$

- Make explicit that $\wr C$ is identified by the combination of its $\underline{C}$ and $\overline{C}$:

$$\begin{aligned}
&\forall \phi . \wr C(\phi) \rightarrow \exists \psi . lapr(\phi, \psi), \\
&\forall \phi . \wr C(\phi) \rightarrow \exists \psi . uapr(\phi, \psi), \\
&\forall \phi, \psi, \varphi . lapr(\phi, \psi) \wedge lapr(\phi, \varphi) \rightarrow \psi = \varphi, \quad (10) \\
&\forall \phi, \psi, \varphi . uapr(\phi, \psi) \wedge uapr(\phi, \varphi) \rightarrow \psi = \varphi, \\
&\forall \phi_1, \phi_2, \psi_1, \psi_2 . lapr(\phi_1, \psi_1) \wedge uapr(\phi_1, \psi_2) \wedge \\
&\quad lapr(\phi_2, \psi_1) \wedge uapr(\phi_2, \psi_2) \rightarrow \phi_1 = \phi_2 .
\end{aligned}$$

Further requirements

- In the DL KB, the set of 'attributes' amounts to $\mathcal{R} \cup \mathcal{D}$
- Impose that those attributes P taken from $\mathcal{R} \cup \mathcal{D}$ must be represented in the ontology with $\wr C$ as its domain
- The indistinguishability relation is reflexive, symmetric, and transitive

Related works

- DL or OWL ontologies:
  - More precise notion of $\wr C$ cf. the tuple notation [JWTX09]
  - Use both $\mathcal{R}$ and $\mathcal{D}$ for the 'attributes' (properties) of the concepts cf. $\mathcal{R}$ only in [IGNI07, SKP07]
  - Include the properties of the indistinguishability relation cf. their omission in [Lia96] or using the properties of the similarity relation [BS09]
  - Adhere to proper $\underline{C}$, $\overline{C}$, and $\wr C$ in that they all have the same collection of properties from $\mathcal{R} \cup \mathcal{D}$ cf. the 'approximations' with different sets of attributes in [SKP07]
- Other formalisations:
  - Propositional: [D97, KB08, Nak96]
  - Data/logical level: with DATALOG [DLSS06] or extended logic programs [VDM03]

Core issues for a rough DL KB

- Identification of the rough concept
  - Second order logic
  - **id** constraint in the language ($\mathcal{DLR}_{ifd}$, *DL-Lite*$_{\mathcal{A},id}$)
  - Let each rough concept be subsumed by some "`RoughC`" to communicate with the modeller it is intended as a rough concept
- Properties of the indistinguishability relation ($\mathcal{SROIQ}$, $\mathcal{DLR}_{\mu}$)
- Reasoning: possibly and definitely satisfiable, rough subsumption, and instance classification
- Scalable non-empty ABox to figure out if some concept is actually a rough concept or not (*DL-Lite* family)

Core issues for a rough DL KB

- Identification of the rough concept
  - Second order logic
  - **id** constraint in the language ($\mathcal{DLR}_{ifd}$, $DL\text{-}Lite_{\mathcal{A},id}$)
  - Let each rough concept be subsumed by some "`RoughC`" to communicate with the modeller it is intended as a rough concept
- Properties of the indistinguishability relation ($\mathcal{SROIQ}$, $\mathcal{DLR}_{\mu}$)
- Reasoning: possibly and definitely satisfiable, rough subsumption, and instance classification
- Scalable non-empty ABox to figure out if some concept is actually a rough concept or not (*DL-Lite* family)

Core issues for a rough DL KB

- Identification of the rough concept
  - Second order logic
  - **id** constraint in the language ($\mathcal{DLR}_{ifd}$, *DL-Lite*$_{\mathcal{A},id}$)
  - Let each rough concept be subsumed by some "RoughC" to communicate with the modeller it is intended as a rough concept
- Properties of the indistinguishability relation ($\mathcal{SROIQ}$, $\mathcal{DLR}_{\mu}$)
- Reasoning: possibly and definitely satisfiable, rough subsumption, and instance classification
- Scalable non-empty ABox to figure out if some concept is actually a rough concept or not (*DL-Lite* family)

Three experiments

- Promiscuous bacteria with OBDA and a operational database (HGT-DB [GVGMR03] using set up described in [CKN+10] and OBDA plugin for Protégé for querying)—**to find vague instances**
- Promiscuous bacteria with the expressive ontology language OWL 2 DL and Protégé—**to model the rough concepts**
- Revisiting Septic patients example [JWTX09, SKP07] to give the Protégé with OWL 2 DL a fairer chance and use RacerPro 2 Preview too—**To model rough concepts, with complex definitions, and find vague instances in the few objects in the ABox**
- http://www.meteck.org/files/roughontosuppl/roughontotests.html

Three experiments

- Promiscuous bacteria with OBDA and a operational database (HGT-DB [GVGMR03] using set up described in [CKN+10] and OBDA plugin for Protégé for querying)—**to find vague instances**
- Promiscuous bacteria with the expressive ontology language OWL 2 DL and Protégé—**to model the rough concepts**
- Revisiting Septic patients example [JWTX09, SKP07] to give the Protégé with OWL 2 DL a fairer chance and use RacerPro 2 Preview too—**To model rough concepts, with complex definitions, and find vague instances in the few objects in the ABox**
- `http://www.meteck.org/files/roughontosuppl/roughontotests.html`

Procedure (1/2)

1. Develop a basic ontology in OWL 2 QL or *DL-Lite*$_{\mathcal{A}}$ stored as an OWL file;
2. Obtain the relational database in Oracle, DB2, MySQL, or PostgreSQL;
3. Set up the OBDA system with the QuOnto reasoner, Protégé, and OBDA plugin for Protégé;
4. Declare the mappings between the classes and properties in the ontology and SQL queries over the database in the OBDA plugin for Protégé;
5. Find all rough concepts with respect to the data through posing ontology-mediated queries (in SPARQL or EQL-Lite), evaluating the result set, and adding them to the ontology;

Procedure (2/2)

6. Migrate this ontology to an expressive OWL species, such as OWL 2 DL, by:
    i. Declaring the semantics from the WHERE clause in the SQL query of the mapping layer as object and data properties in the ontology;
    ii. Adding upper and lower approximations of each rough concept;
    iii. Adding the indistinguishability object property with its properties (reflexive, symmetric, transitive);
    iv. Adding the axioms relating the approximations to the rough concepts and vice versa;
7. When the rough reasoning services are implemented, run the reasoner with the enhanced ontology to check satisfiability and consistency

On step 6 of the procedure

- Recollecting the relevant part of OWL 2's direct semantics [MPSG09]: take a vocabulary V with, among others,
  i. V_C denotes the set of classes
  ii. Class interpretation function $\cdot^C$ assigns to each class $C \in V_C$ a subset $(C)^C \subseteq \Delta^I$
  iii. V_{OP} the set of object properties
  iv. $\cdot^{OP}$ is the object property interpretation function that assigns to each object property $OP \in V_{OP}$ a subset $(OP)^{OP} \subseteq \Delta^I \times \Delta^I$
- Add rough concept, upper, and lower approximation such that $\wr \mathtt{C}, \overline{\mathtt{C}}, \underline{\mathtt{C}} \in V_C$

On step 6 of the procedure (Cont'd)

- add the indistinguishability relation Ind over $\Delta^I \times \Delta^I$ as an object property such that $\texttt{Ind} \in V_{OP}$ and the ontology contains the assertions:
  `ReflexiveObjectProperty(a:Ind)`,
  `SymmetricObjectProperty(a:Ind)`, and
  `TransitiveObjectProperty(a:Ind)`
- assign the semantics to the classes:

$$
\begin{aligned}
(\overline{\mathtt{C}})^I &= \{x \in \Delta^I \mid \exists y \in \Delta^I, (x,y) \in \mathtt{Ind} \wedge y \in \mathtt{C}^I\} \\
(\underline{\mathtt{C}})^I &= \{x \in \Delta^I \mid \forall y \in \Delta^I, (x,y) \in \mathtt{Ind} \rightarrow y \in \mathtt{C}^I\} \\
(\wr \mathtt{C})^I &= (\langle \underline{\mathtt{C}}, \overline{\mathtt{C}} \rangle)^I = \langle (\underline{\mathtt{C}})^I, (\overline{\mathtt{C}})^I \rangle
\end{aligned} \tag{11}
$$

which amounts to the assertions for any $\overline{\mathtt{C}}$ and $\underline{\mathtt{C}}$ in OWL 2 DL functional syntax:
`EquivalentClasses(C̄ ObjectSomeValuesFrom(a:Ind a:C))` and
`EquivalentClasses(C̲ ObjectAllValuesFrom(a:Ind a:C))`

On step 6 of the procedure (Cont'd)

- add the indistinguishability relation Ind over $\Delta^I \times \Delta^I$ as an object property such that $\texttt{Ind} \in V_{OP}$ and the ontology contains the assertions:
  `ReflexiveObjectProperty(a:Ind)`,
  `SymmetricObjectProperty(a:Ind)`, and
  `TransitiveObjectProperty(a:Ind)`
- assign the semantics to the classes:

$$
\begin{aligned}
(\overline{\texttt{C}})^I &= \{x \in \Delta^I \mid \exists y \in \Delta^I, (x,y) \in \texttt{Ind} \land y \in \texttt{C}^I\} \\
(\underline{\texttt{C}})^I &= \{x \in \Delta^I \mid \forall y \in \Delta^I, (x,y) \in \texttt{Ind} \rightarrow y \in \texttt{C}^I\} \\
(\wr\texttt{C})^I &= (\langle \underline{\texttt{C}}, \overline{\texttt{C}} \rangle)^I = \langle (\underline{\texttt{C}})^I, (\overline{\texttt{C}})^I \rangle
\end{aligned}
\qquad (11)
$$

which amounts to the assertions for any $\overline{\texttt{C}}$ and $\underline{\texttt{C}}$ in OWL 2 DL functional syntax:

`EquivalentClasses(`$\overline{\texttt{C}}$` ObjectSomeValuesFrom(a:Ind a:C))` and

`EquivalentClasses(`$\underline{\texttt{C}}$` ObjectAllValuesFrom(a:Ind a:C))`

On step 6 of the procedure (Cont'd)

- (11) is approximated in an arbitrary ontology by adding two auxiliary roles, `uapr`, `lapr` $\in V_{OP}$ that each have ≀C as domain, and cardinality exactly 1:
  `ObjectPropertyDomain(a:upar a:≀C)`,
  `ObjectPropertyDomain(a:lapr a:≀C)`,
  `ObjectExactCardinality(1 a:uapr a:`$\overline{\texttt{C}}$`)`, and
  `ObjectExactCardinality(1 a:lapr a:`$\underline{\texttt{C}}$`)`.
- All this is added to the expressive ontology for each rough concept

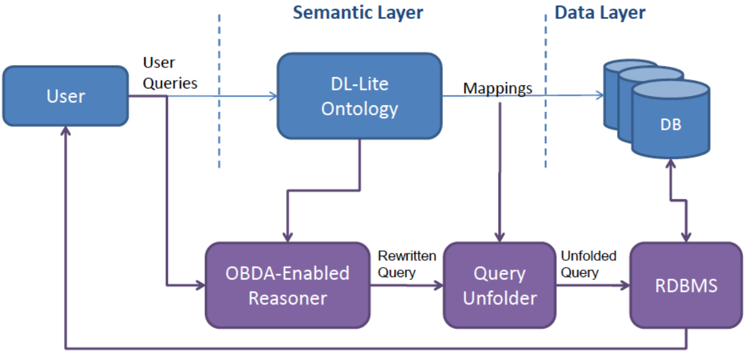
On step 6 of the procedure (Cont'd)

- (11) is approximated in an arbitrary ontology by adding two auxiliary roles, `uapr`, `lapr` $\in V_{OP}$ that each have $\wr$C as domain, and cardinality exactly 1:
  ObjectPropertyDomain(a:upar a:$\wr$C),
  ObjectPropertyDomain(a:lapr a:$\wr$C),
  ObjectExactCardinality(1 a:uapr a:$\overline{\text{C}}$), and
  ObjectExactCardinality(1 a:lapr a:$\underline{\text{C}}$).
- All this is added to the expressive ontology for each rough concept

Materials

- OBDA: Oracle 10g RDMBS, QuOnto, Protégé 3.3.1, OBDA Plugin for Protégé
- Expressive ontologies: Protégé 4.0 with Pellet 2.0, Fact++, RacerPro 2.0 Preview
- Ontologies: HGT conceptual data model in $DL\text{-}Lite_{\mathcal{A}}$ and an expressive version in OWL 2 DL, sepsis ontology based on description in [SKP07]
- The experiments were carried out on a Macbook Pro with Mac OS X v 10.5.8 with 2.93 GHz Intel core 2 Duo and 4 GB memory

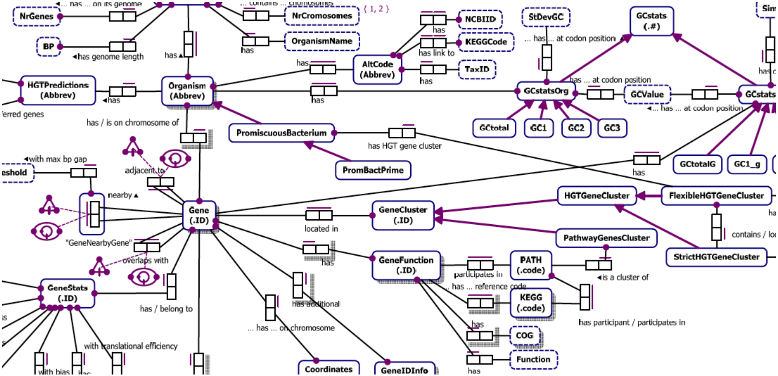
Section of the ORM version of the HGT model



Materials

- OBDA: Oracle 10g RDMBS, QuOnto, Protégé 3.3.1, OBDA Plugin for Protégé
- Expressive ontologies: Protégé 4.0 with Pellet 2.0, Fact++, RacerPro 2.0 Preview
- Ontologies: HGT conceptual data model in $DL\text{-}Lite_{\mathcal{A}}$ and an expressive version in OWL 2 DL, sepsis ontology based on description in [SKP07]
- The experiments were carried out on a Macbook Pro with Mac OS X v 10.5.8 with 2.93 GHz Intel core 2 Duo and 4 GB memory

Section of the ORM version of the HGT model



Materials

- OBDA: Oracle 10g RDMBS, QuOnto, Protégé 3.3.1, OBDA Plugin for Protégé
- Expressive ontologies: Protégé 4.0 with Pellet 2.0, Fact++, RacerPro 2.0 Preview
- Ontologies: HGT conceptual data model in $DL\text{-}Lite_{\mathcal{A}}$ and an expressive version in OWL 2 DL, sepsis ontology based on description in [SKP07]
- The experiments were carried out on a Macbook Pro with Mac OS X v 10.5.8 with 2.93 GHz Intel core 2 Duo and 4 GB memory

OBDA and QuOnto

- Promiscuous bacterium is a Organism and, **putatively**, it must have more than 5 flexible hgt-gene clusters and the percentage of genes on the chromosome that are predicted to be horizontally acquired as > 10
- Head of the mapping:
  PromiscuousBacterium(getPromBact($abbrev,$ccount, $percentage))
- Body:
  ```
  SELECT organisme.abbrev, ccount, organisme.percentage
  FROM ( SELECT idorganisme, COUNT(distinct cstart)
  as ccount FROM COMCLUSTG2 GROUP BY idorganisme
  ) flexcount, organisme
  WHERE organisme.abbrev = flexcount.idorganisme AND
  organisme.percentage > 10 AND flexcount.ccount > 5
  ```

OBDA and QuOnto

- Promiscuous bacterium is a Organism and, **putatively**, it must have more than 5 flexible hgt-gene clusters and the percentage of genes on the chromosome that are predicted to be horizontally acquired as > 10
- Head of the mapping:

```
PromiscuousBacterium(getPromBact($abbrev,$ccount,
$percentage))
```

- Body:

```
SELECT organisme.abbrev, ccount, organisme.percentage
FROM ( SELECT idorganisme, COUNT(distinct cstart)
as ccount FROM COMCLUSTG2 GROUP BY idorganisme
) flexcount, organisme
WHERE organisme.abbrev = flexcount.idorganisme AND
organisme.percentage > 10 AND flexcount.ccount > 5
```

OBDA and QuOnto

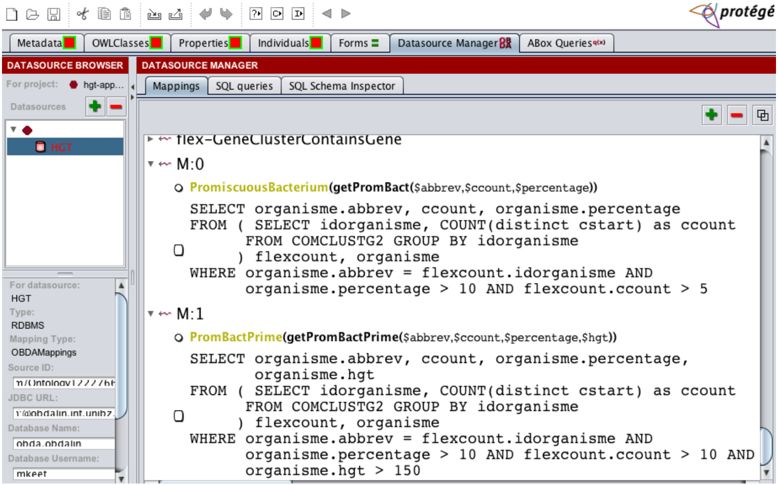
- Promiscuous bacterium is a Organism and, **putatively**, it must have more than 5 flexible hgt-gene clusters and the percentage of genes on the chromosome that are predicted to be horizontally acquired as > 10
- Head of the mapping:

```
PromiscuousBacterium(getPromBact($abbrev,$ccount,
$percentage))
```

- Body:

```
SELECT organisme.abbrev, ccount, organisme.percentage
FROM ( SELECT idorganisme, COUNT(distinct cstart)
as ccount FROM COMCLUSTG2 GROUP BY idorganisme
) flexcount, organisme
WHERE organisme.abbrev = flexcount.idorganisme AND
organisme.percentage > 10 AND flexcount.ccount > 5
```

Mappings


protégé
Metadata
OWLClasses
Properties
Individuals
Forms
Datasource Manager
ABox Queries
DATASOURCE BROWSER
For project:
hgt-app...
Datasources
HGT
For datasource:
HGT
Type:
RDBMS
Mapping Type:
OBDAMappings
Source ID:
JDBC URL:
Database Name:
obda.obdalin
Database Username:
mkeet
DATASOURCE MANAGER
Mappings
SQL queries
SQL Schema Inspector
flex-GeneClusterContainsGene
M:0
PromiscuousBacterium(getPromBact($abbrev,$ccount,$percentage))
SELECT organisme.abbrev, ccount, organisme.percentage
FROM (SELECT idorganisme, COUNT(distinct cstart) as ccount
FROM COMCLUSTG2 GROUP BY idorganisme
) flexcount, organisme
WHERE organisme.abbrev = flexcount.idorganisme AND
organisme.percentage > 10 AND flexcount.ccount > 5
M:1
PromBactPrime(getPromBactPrime($abbrev,$ccount,$percentage,$hgt))
SELECT organisme.abbrev, ccount, organisme.percentage,
organisme.hgt
FROM (SELECT idorganisme, COUNT(distinct cstart) as ccount
FROM COMCLUSTG2 GROUP BY idorganisme
) flexcount, organisme
WHERE organisme.abbrev = flexcount.idorganisme AND
organisme.percentage > 10 AND flexcount.ccount > 10 AND
organisme.hgt > 150

Promiscuous bacteria with OBDA and QuOnto

- Query the HGT-DB through the ontology with a SPARQL query:
  - 98 objects are retrieved where *Dehalococcoides CBDB1* and *Thermotoga maritima* are truly indistinguishable bacteria, i.e. they have the same values for all the selected and constrained attributes
  - A few others are very close to being so, e.g., *Pelodictyon luteolum DSM273* and *Synechocystis PCC6803* who have both 6 clusters and 10.1% and 10.2%, respectively
- Refine to Prombact′ to check if we can obtain a crisp concept

```
SELECT organisme.abbrev,ccount,organisme.percentage,
organisme.hgt FROM ...
WHERE organisme.abbrev = flexcount.idorganisme AND
organisme.percentage > 10 AND flexcount.ccount > 10 AND
organisme.hgt > 150
```

Promiscuous bacteria with OBDA and QuOnto

- Query the HGT-DB through the ontology with a SPARQL query:
  - 98 objects are retrieved where *Dehalococcoides CBDB1* and *Thermotoga maritima* are truly indistinguishable bacteria, i.e. they have the same values for all the selected and constrained attributes
  - A few others are very close to being so, e.g., *Pelodictyon luteolum DSM273* and *Synechocystis PCC6803* who have both 6 clusters and 10.1% and 10.2%, respectively
- Refine to `Prombact'` to check if we can obtain a crisp concept

```
SELECT organisme.abbrev,ccount,organisme.percentage,
organisme.hgt FROM ...
WHERE organisme.abbrev = flexcount.idorganisme AND
organisme.percentage > 10 AND flexcount.ccount > 10 AND
organisme.hgt > 150
```

Promiscuous bacteria with OWL 2 DL

Add definitions

$$PromBact \equiv Organism \sqcap \exists\, Percentage.real_{>10} \sqcap \geq 6\ hasHGTCl.FlexibleHGTGeneCl \quad (12)$$

$$PromBact' \equiv PromBact \sqcap \exists\, Percentage.real_{>10} \sqcap \geq 11\ hasHGTCl.FlexibleHGTGeneCl \sqcap \exists\, NrPredHGTgenes.integer_{>150} \quad (13)$$

and rough concept notions (14-17) with their relational properties **only for the rough concept**:

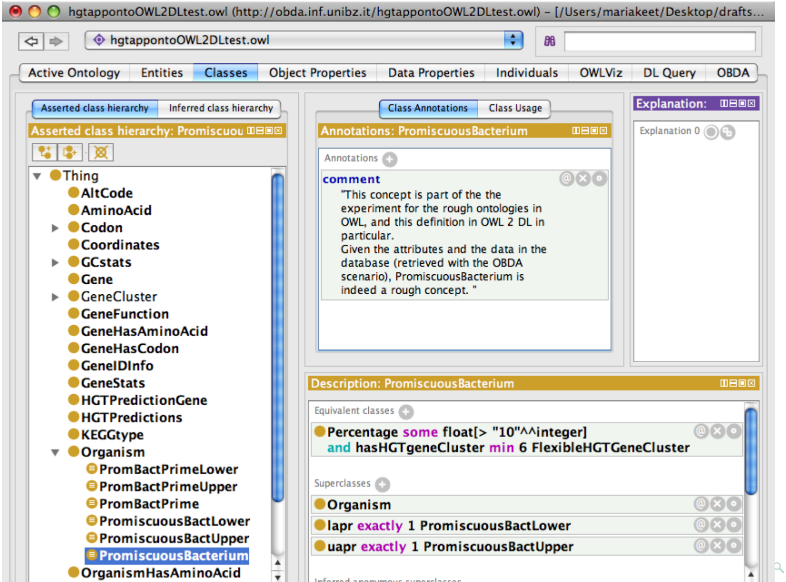
$$PromBact \sqsubseteq\ = 1\ lapr.PromBactLapr \quad (14)$$

$$PromBact \sqsubseteq\ = 1\ uapr.PromBactUapr \quad (15)$$

$$PromBactLapr \equiv \forall\, ind.PromBact \quad (16)$$

$$PromBactUapr \equiv \exists\, ind.PromBact \quad (17)$$

In Protégé 4.0



Bone and strict septic

Patients may be septic or are certainly septic, according to the so-called *Bone criteria* and Bone criteria together with three out of another five criteria, respectively; e.g., the Bone criteria (from [SKP07]):

- *Has infection;*
- *At least two out of four criteria of the Systemic Inflammatory Response Syndrome:*
  - *temperature* $> 38^\circ C$ *OR temperature* $< 36^\circ C$*;*
  - *respiratory rate* > 20 *breaths/minute OR* $PaCO_2 < 32$ *mmHg;*
  - *heart rate* > 90 *beats/minute;*
  - *leukocyte count* $< 4000\ mm^3$ *OR leukocyte count* $> 12000\ mm^3$*;*
- *Organ dysfunction, hypoperfusion, or hypotension.*

Bone septic in OWL

can be encoded in OWL in Protégé as being an `EquivalentClass` to `BoneSeptic`, as follows:

```
(hasDiagnosis some Infection and
hasSymptom some (Hypoperfusion or Hypotension or OrganDysfunction)
and (((temperature some int[> 38] or temperature some int[<36])
and (respiratoryRate some int[>20] or paco2count some int[<32]))
or ((temperature some int[>38] or temperature some int[<36]) and
heartRate some int[>90])
or ((temperature some int[>38] or temperature some int[<36]) and
(leukocyteCount some int[<4000] or leukocyteCount some int[>12000]))
or ((respiratoryRate some int[>20] or paco2count some int[<32]) and
heartRate some int[>90])
or ((respiratoryRate some int[>20] or paco2count some int[<32]) and
(leukocyteCount some int[<4000] or leukocyteCount some int[>12000]))
or (heartRate some int[>90] and (leukocyteCount some int[<4000] or
leukocyteCount some int[>12000]))))
```

Results

- Definitions and encodings in Protégé 4.0 and RacerPro 2.0 preview and data of 17 'patients' s.t. the boundary region of each concept is not empty are available through `http://www.meteck.org/files/roughontosuppl/roughontotests.html`
- Protégé 4.0 with Pellet 2.0 did not work at all
- Protégé 4.0 with FaCT++ works well with a few dummy concepts and a few instances, but not properly with sepsis (due to unstable ODE, therefore not tested with Hermit)
- RacerPro 2.0 preview never crashed during exerimentation and did return the correct classifications within about 2 hours (probably due to heavy use of concrete domains)

Results

- Definitions and encodings in Protégé 4.0 and RacerPro 2.0 preview and data of 17 'patients' s.t. the boundary region of each concept is not empty are available through
  http://www.meteck.org/files/roughontosuppl/roughontotests.html
- Protégé 4.0 with Pellet 2.0 did not work at all
- Protégé 4.0 with FaCT++ works well with a few dummy concepts and a few instances, but not properly with sepsis (due to unstable ODE, therefore not tested with Hermit)
- RacerPro 2.0 preview never crashed during exerimentation and did return the correct classifications within about 2 hours (probably due to heavy use of concrete domains)

Results

- Definitions and encodings in Protégé 4.0 and RacerPro 2.0 preview and data of 17 'patients' s.t. the boundary region of each concept is not empty are available through `http://www.meteck.org/files/roughontosuppl/roughontotests.html`
- Protégé 4.0 with Pellet 2.0 did not work at all
- Protégé 4.0 with FaCT++ works well with a few dummy concepts and a few instances, but not properly with sepsis (due to unstable ODE, therefore not tested with Hermit)
- RacerPro 2.0 preview never crashed during exerimentation and did return the correct classifications within about 2 hours (probably due to heavy use of concrete domains)

Results

- Definitions and encodings in Protégé 4.0 and RacerPro 2.0 preview and data of 17 'patients' s.t. the boundary region of each concept is not empty are available through `http://www.meteck.org/files/roughontosuppl/roughontotests.html`
- Protégé 4.0 with Pellet 2.0 did not work at all
- Protégé 4.0 with FaCT++ works well with a few dummy concepts and a few instances, but not properly with sepsis (due to unstable ODE, therefore not tested with Hermit)
- RacerPro 2.0 preview never crashed during exerimentation and did return the correct classifications within about 2 hours (probably due to heavy use of concrete domains)

Discussion

- Ontologies, OBDA, and ontology development tools provide a means to deal with successive de-vagueing during experimentation
- Makes the iterations with the selected properties explicit
- The two-step process with OBDA and OWL 2 DL is an advance w.r.t. traceability
- Meaningful and usable language extensions for a proper rough DL are limited, reasoning services can be augmented
- Other options:
  - At least: partition the data source;
  - Sophisticated modularization of both the ontology and the data(base) (e.g., [LTW09, BBLW10])
  - Turn the 'methodology' into a structured scientific workflow to be able to seamlessly go back and forth between OBDA and OWL 2 DL

Discussion

- Ontologies, OBDA, and ontology development tools provide a means to deal with successive de-vagueing during experimentation
- Makes the iterations with the selected properties explicit
- The two-step process with OBDA and OWL 2 DL is an advance w.r.t. traceability
- Meaningful and usable language extensions for a proper rough DL are limited, reasoning services can be augmented
- Other options:
  - At least: partition the data source;
  - Sophisticated modularization of both the ontology and the data(base) (e.g., [LTW09, BBLW10]
  - Turn the 'methodology' into a structured scientific workflow to be able to seamlessly go back and forth between OBDA and OWL 2 DL

Conclusions and future directions

- Given rough sets' semantics, there is no, nor will there be, a DL that represents all essential aspects precisely, though $\mathcal{SROIQ}(\mathcal{D})$ is fairly close
- Interaction with large amounts of data that makes any extension with roughness interesting and useful, e.g., the *DL-Lite* family with its OBDA infrastructure
- The experimentation showed it is possible, but—thus far—impractical, to have rough knowledge bases

- Streamline the rather elaborate procedure into a scientific workflow
- Develop implementations of sophisticated ontology and data modularization
- The 'attributes'

Conclusions and future directions

- Given rough sets' semantics, there is no, nor will there be, a DL that represents all essential aspects precisely, though $\mathcal{SROIQ}(D)$ is fairly close
- Interaction with large amounts of data that makes any extension with roughness interesting and useful, e.g., the *DL-Lite* family with its OBDA infrastructure
- The experimentation showed it is possible, but—thus far—impractical, to have rough knowledge bases

- Streamline the rather elaborate procedure into a scientific workflow
- Develop implementations of sophisticated ontology and data modularization
- The 'attributes'

References

Franz Baader, Meghyn Bienvenu, Carsten Lutz, and Frank Wolter.
Query and predicate emptiness in description logics.
In *Proc. of KR'10*, 2010.

Fernando Bobillo and Umberto Straccia.
Supporting fuzzy rough sets in fuzzy description logics.
In Claudio Sossai and Gaetano Chemello, editors, *Proceedings of the European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU'09)*, volume 5590 of *LNCS*, pages 676–687. Springer, 2009.

D. Calvanese, C.M. Keet, W. Nutt, M. Rodríguez-Muro, and G. Stefanoni.
Web-based graphical querying of databases through an ontology: the WONDER system.
In *Proceedings of ACM Symposium on Applied Computing (ACM SAC'10)*, pages 1389–1396. ACM, 2010.
March 22-26 2010, Sierre, Switzerland.

Ivo Düntsch.
A logic for rough sets.
Theoretical Computer Science (B), 179:427–436, 1997.

Patrick Doherty, Witold Lukaszewicz, Andrzej Skowron, and Andrzej Szalas.
Knowledge Representation Techniques: A Rough Set Approach.
Studies in fuzziness and soft computing. Springer, 2006.

N. Fanizzi, C. D'Amato, F. Esposito, and T. Lukasiewicz.
Representing uncertain concepts in rough description logics via contextual indiscernibility relations.
In *Proceedings of the 4th International Workshop on Uncertainty Reasoning for the Semantic Web (URSW'08)*, volume 423 of *CEUR-WS*, 2008.

References

S. Garcia-Vallvé, E. Guzman, M.A. Montero, and A. Romeu.
HGT-DB: a database of putative horizontally transferred genes in prokaryotic complete genomes.
Nucleic Acids Research, 31(1):187–189, 2003.

Soyohei Ishizu, Andreas Gehrmann, Yoshimitsu Nagai, and Yusei Inukai.
Rough ontology: extension of ontologies by rough sets.
In M. J. Smith and G. Salvendy, editors, *Proceedings of Human Interface and the Management of Information. Methods, Techniques and Tools in Informaiton Design*, volume 4557 of *LNCS*, pages 456–462. Springer, 2007.

Yuncheng Jiang, Ju Wang, Suqin Tang, and Bao Xiao.
Reasoning with rough description logics: An approximate concepts approach.
Information Sciences, 179:600–612, 2009.

Md. Aquil Khan and Mohua Banerjee.
Formal reasoning with rough sets in multiple-source approximation systems.
International Journal of Approximate Reasoning, 49(2):466 – 477, 2008.
Special Section on Probabilistic Rough Sets and Special Section on PGM'06.

C. Maria Keet.
A Formal Theory of Granularity.
Phd thesis, KRDB Research Centre, Faculty of Computer Science, Free University of Bozen-Bolzano, Italy, April 2008.

C. Maria Keet, Marco Roos, and M. Scott Marshall.
A survey of requirements for automated reasoning services for bio-ontologies in OWL.
In *Proceedings of the 3rd Workshop on OWL: Experiences and Directions (OWLED 2007)*, volume 258 of *CEUR-WS*, 2007.
6-7 June 2007, Innsbruck, Austria.

References

C.-J. Liau.
On rought terminological logics.
In *Proceedings of the 4th International Workshop on Rough Sets, Fuzzy Sets and machine Discovery (RSFD'96)*, pages 47–54, 1996.

C. Lutz, D. Toman, and F. Wolter.
Conjunctive query answering in the description logic el using a relational database system.
In *Proceedings of the 21st International Joint Conference on Artificial Intelligence IJCAI09*. AAAI Press, 2009.

Boris Motik, Peter F. Patel-Schneider, and Bernardo Cuenca Grau.
OWL 2 web ontology language: Direct semantics.
W3c recommendation, W3C, 27 October 2009 2009.

Akira Nakamura.
A rough logic based on incomplete information and its application.
International Journal of Approximate Reasoning, 15(4):367 – 378, 1996.
Rough Sets.

S. Schlobach, M. Klein, and L. Peelen.
Description logics with approximate definitions—precise modeling of vague concepts.
In *Proceedings of the 20th International Joint Conference on Articial Intelligence (IJCAI'07)*, pages 557–562. AAAI Press, 2007.

Aida Vitória, Carlos Viegas Damásio, and Jan Maluszynski.
From rough sets to rough knowledge bases.
Fundamenta Informaticae, 57(2-4):215–246, 2003.

Thank you for your attention