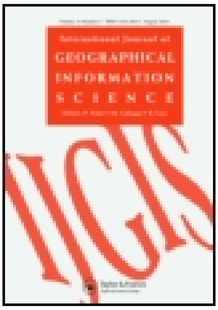

International Journal of Geographical Information Science
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tgis20

A formalization for semantic location granules

Jorge Bernad [a], Carlos Bobed [a], Eduardo Mena [a] & Sergio Ilarri [a]
[a] Department of Computer Science and Systems Engineering, University of Zaragoza, Zaragoza, Spain
Version of record first published: 16 Nov 2012.

International Journal of Geographical Information Science
i*First*, 2012, 1–19

A formalization for semantic location granules

Jorge Bernad*, Carlos Bobed, Eduardo Mena and Sergio Ilarri

Department of Computer Science and Systems Engineering, University of Zaragoza, Zaragoza, Spain

(*Received 6 June 2012; final version received 5 October 2012*)

Location-based services have become an increasingly interesting research area in the last two decades. However, in many scenarios, dealing with the most precise location coordinates is not the best solution since people structure the world in *geographic areas* instead of coordinates. Since humans work with abstractions, and names are the way we refer to those abstractions, introducing semantics in geographic definitions becomes natural. For example, users can be interested in states with vacation resorts and may want to retrieve the state names, instead of the exact geographic limits of such states. Moreover, semantics introduces new challenges, such as how to exploit the location semantics to infer new information from known definitions. For instance, we may want a system to automatically obtain the value added tax (VAT) that should be applied by a shop in Madrid, inferring the applicable tax by considering the economic area where Madrid is included (in this case, Spain); notice that the VAT should not be inferred from a bigger economic area, like Europe, although it also includes Madrid geographically. Thus, the expression of locations at different granularities extends the traditional location-based query processing to consider the most appropriate semantics for each user.

In this article, adopting description logics (DLs) as a base formalism, we provide a formalization of the notion of *semantic location granule* and *semantic granule map*. We benefit from the underlying semantics of the different granularities to extend the expressivity of location-based queries and automatically discover and infer new knowledge. The model we propose uses a DL reasoner to infer new granules relationships. In particular, a DL reasoner can infer containment and intersection relationships between location granules (and help to obtain several more relationships), which provides the way to introduce semantics in location-based queries. This is done within the logical frame of DLs, thus ensuring that our approach can be supported by existing regular DL reasoners (such as Pellet, Racer Pro, and HermiT) without the need to extend their reasoning capabilities.

Keywords: Description Logics; ontology; spatial reasoning

1. Introduction

Research in mobile computing has increased in the last two decades motivated by the ever-growing use of mobile devices and their computational power. In this context, *location-based services* (LBSs) (Schiller and Voisard (2004) have attracted the interest of both academy and industry. The added value of these services is that they can provide information relevant for the location of the mobile users.

*Corresponding author. Email: jbernad@unizar.es

ISSN 1365-8816 print/ISSN 1362-3087 online

http://dx.doi.org/10.1080/13658816.2012.739691
http://www.tandfonline.com

One of the main challenges for the development of LBSs is how to process continuous *location-dependent queries* (Ilarri *et al.* 2010), which are queries whose answers depend on the location of certain moving objects (the mobile user and/or others). Most works in this area assume that the locations of the moving objects are managed with Global Positioning System (GPS) precision. However, an end-user of an application does not usually want to retrieve information in terms of precise GPS locations. Instead, he/she probably needs to use more abstract notions of location, as the geographic coordinates are probably meaningless to him/her. Thus, users usually manage *area names*. For instance, if a user wants to go by subway to the Museum of Modern Art (MoMa) in New York, he/she will not be interested in the precise coordinates of the area where he/she has to get or leave the subway, but he/she may to know the name and number of the street where the nearest station is to get the subway, or he/she may need to know the name of a station near the MoMA to leave the subway. To take these situations into account, the concept of *location granule* was defined in Ilarri *et al.* (2011a) to enable the processing of location-dependent queries that reference *location granule maps* (sets of location granules). For example, Madrid could be a location granule according to the location granule map that defines the provinces of Spain.

Our previous approach presented in Ilarri *et al.* (2011a) to manage location-dependent queries with location granules enhances the expressivity of location-dependent queries and is an important step forward. However, as it basically considers a location granule as a simple set of GPS points, it neglects the semantics behind the locations and is unable to infer implicit knowledge automatically. For instance, if a user is in the city of Zaragoza, it could be inferred that he/she is also close to France, a country with famous red wines. In order to enable these kinds of deductions, the most popular approach is to represent the knowledge of the domain by using ontologies (Haarslev *et al.* 1998, Frank 2003), defined by Gruber as an 'explicit formalization of a conceptualization' (Gruber 1993). The interest of linking LBSs and semantics has been emphasized in Ilarri *et al.* (2011b).

Ontologies are usually expressed by using *Description Logics* (DLs) (Baader 2007). Several works in the field of *Geographic Information Systems* (GIS) (Rigaux *et al.* 2002; Shekhar and Chawla 2002) have used the predominant relational data model to represent topological relations (e.g., the Region Connection Calculus (RCC), see Randell *et al.* (1992)). However, to support reasoning with geographic elements, several previous works have studied the introduction of ontologies in the area of GIS (Lutz and Klien 2006, Couclelis 2010), and the introduction of different types of topological relations in DLs (Lutz and Möller 1997, Haarslev *et al.* 1998). Despite these efforts, more research is needed in this area to effectively enhance the processing of location-dependent queries with inferencing capabilities over location granules. For example, *isContained* is a key topological relation (it allows representing the geographic hierarchy of areas), but the existing proposals to represent such a relation have important disadvantages (depending on the specific proposal, wrong conclusions may be obtained or a human asserting by hand many of the *isContained* relations may be required).

In this article, adopting DLs as a base formalism, we provide a formalization of the concepts of location granules and location granule maps and enhance their description with additional (not necessarily geographic) information by using DLs, which leads to what we call *semantic location granules* and *semantic location granule maps*. Besides, we also describe how a DL reasoner can automatically discover and infer new knowledge by exploiting these concepts. Thus, we benefit from the different granularities and their underlying semantics to extend the expressivity of location-based queries and add inferencing capabilities. The model we propose uses a DL reasoner to infer new relationships. In particular, a reasoner can infer *isContained* and *intersects* relationships between location granules (and other binary topological relationships), which enables the introduction of semantics in location-dependent queries. The model presented in this article contributes to improve our previous works in Bobed *et al.* (2010), Ilarri *et al.* (2011a) (focused mainly on the location-dependent constraint *inside*) by introducing new inferring capabilities. It also solves the problem of representing the *isContained* relation and working with it in a more powerful way. As an example, let us suppose that we are traveling with a smartphone and we want to retrieve the names of provinces with typical red wines and located within a radius of 100 miles from the province where we are. In this scenario, the issue is what kind of DL can be used and how to use a reasoner to infer: (i) the province where we are, (ii) which provinces are in a radius of 100 miles from that province, and (iii) which of these provinces are famous because of their production of Spanish red wine. The DL-based model that we propose would help in this scenario because it enables several automatic inferences, such as determining that a certain area is within another one, inferring that two areas intersect, and deducing other qualitative relations. Summing up, our main contributions are as follows.

- We define what conditions are necessary in a DL in order to use a reasoner to infer new containment and intersection relations avoiding wrong deductions due to topological inclusions.
- Using these conditions, we define the concepts of semantic granule and semantic granule map.
- We show how to use a reasoner to exploit the capabilities of semantic granules and semantic granule maps.

The structure of the rest of this article is as follows. First, we review concepts related to DLs in Section 2. Second, in Section 3, we give formal definitions of several concepts that are used in our model, such as location granules, location granule maps, and *inside* constraints. Third, in Section 4, we introduce the concepts of semantic location granules and semantic location granule maps, and what kind of DL is required to express these concepts. Fourth, we explain how a DL reasoner is used to deduce new facts in Section 5. Finally, we present some related works in Section 6, and our conclusions and future work in Section 7.

2. Overview of DL concepts

DLs are '*formal languages for representing knowledge and reasoning about it*' (Baader 2007). In this section, we briefly overview their definitions (we refer the interested reader to Baader (2007), for more details) and study the expressivity requirements for our purposes.

DLs are formed by an intensional layer $\mathcal{T}$ called TBox and an extensional layer $\mathcal{A}$ called ABox. The TBox is composed of a set of terminological axioms. Axioms are formulas of the form $C \equiv D$ or $C \sqsubseteq D$, where C and D are concepts. Concepts are formed by means of: (i) a set of concepts names N_C, conceptualizations of a set of individuals (or instances), for example, *Person*, *Car*, and *Dog*; (ii) a set of roles N_R, which are binary relations between individuals, for instance, *hasPet* and *hasChildren*; and (iii) constructors to define new concepts, such as $\sqcap$, $\sqcup$, $\exists$, and $\forall$. For example, given that we have *Person* and *Dog* as concepts and *hasPet* as role, we can define a new concept to represent people who have a pet dog as $Person \sqcap \exists hasPet.Dog$. An axiom of the form $C \equiv D$ says that concepts C and D are equivalent, that is, any individual that belongs to C also belongs to D, and vice versa. An axiom $C \equiv D$ is called a concept definition if the left hand of the axiom is a concept name. Axioms of type $C \sqsubseteq D$ represent that concept C is subsumed by D, that is., any individual in C is in D, but not necessarily vice versa. A *general TBox* is a finite set of axioms. An example of TBox expressing that men are human and fathers are men who have children is:

$$\mathcal{T} = \{Man \sqsubseteq Human; Father \equiv Man \cap \exists hasChild.Human\}$$

An ABox is a set of assertions that describe a specific state of the world represented by the associated TBox. We can express with assertions that John is Mary's father:

$$\mathcal{A} = \{Man(John); Human(Mary); hasChild(John, Mary)\}$$

John and *Mary* will be constants representing individuals. Let us note that we have not asserted that John is a father, as this is implicitly deduced from the TBox and the ABox. DLs are equipped with a *reasoner* to deduce new knowledge from TBoxes and ABoxes. The knowledge representation given by TBox $\mathcal{T}$ and ABox $\mathcal{A}$ is denoted by $\mathcal{K} = \{\mathcal{T}, \mathcal{A}\}$.

DLs can be classified according to its expressivity, that is, depending on how many different symbols can be used to express axioms and how these symbols can be combined. For example, a DL with constructors $\sqcap, \sqcup, \forall, \exists, \neg, \mathrm{T}$, and $\perp$ is an $\mathcal{ALC}$ DL, and if the set of constructors is enlarged with $\leq n$, $\geq n$ (unqualified number restrictions), we obtain an $\mathcal{ALCN}$ DL. An interpretation $\mathcal{I}$ is a set $\Delta^{\mathcal{I}}$ and a function that associates each concept name C with a subset $C^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, and each role R with a binary relation $R^{\mathcal{I}} \sqsubseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ (see Table 1). Qualitative spatial relations, such as tangential or containment relationships, can be expressed in DLs as roles, as they represent binary relations. However, several problems arise when expressing quantitative relations in a DL. For example, using a DL, computing if a location point is inside an area or if two areas intersect is not easy. These calculations need to do some arithmetic operations with real numbers that can be performed using *concrete domains* (see Lutz (2003), and Baader and Hanschke (1991), for further details). A concrete domain is a set $\Delta_{\mathcal{D}}$ along with a set of predicate names $\Phi_{\mathcal{D}}$ of different n-arities over $\Delta_{\mathcal{D}}$. The set of real numbers with predicates $=, \leq, \geq, +, *, \ldots$ is a usual example of concrete domain. With concrete domains, we express concepts like '*man married with a person of the same age*', $Man \sqcap \exists(wife\ age, age).=$, or '*people with less than 25 years old*', $Person \sqcap \exists age. \leq_{25}$. In order to extend the expressivity of DLs introducing concrete domains, the set of valid symbols to construct formulas is enlarged with: (i) a set of *abstract features* (or *functional roles*), N_{aF}, which is a subset of N_R representing functional relations; and (ii) a set of *concrete features*, N_{cF}, which are functions from the domain of an interpretation $\Delta^{\mathcal{I}}$ to a concrete domain $\Delta_{\mathcal{D}}$. In the examples above,

Table 1. Syntax and interpretation for *ALC* DLs.

$C, D \rightarrow A \in N_C$ (*concept name*) |	
T |	$\mathrm{T}^{\mathcal{I}} = \Delta^{\mathcal{I}}$
$\perp$|	$\perp^{\mathcal{I}} = \emptyset$
$\neg C$ |	$(\neg C)^{\mathcal{I}} = \Delta^{\mathcal{I}} \backslash C^{\mathcal{I}}$
$C \sqcap D$ |	$(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$
$C \sqcup D$ |	$(C \sqcup D)^{\mathcal{I}} = C^{\mathcal{I}} \cup D^{\mathcal{I}}$
$\forall R.C$ |	$(\forall R.C)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \vert \forall b, (a, b) \in R^{\mathcal{I}} \rightarrow b \in C^{\mathcal{I}}\}$
$\exists R.C$ |	$(\exists R.C)^{\mathcal{I}} = \{a \in \Delta^{\mathcal{I}} \vert \exists b \in C^{\mathcal{I}}, (a, b$

wife is an abstract feature, *age* is a concrete feature, and $=$, and $\leq_{25}$ are binary and unary predicates of the concrete domain of natural numbers, respectively. The set of constructors is also enlarged with several operators. Specifically, a path is a composition $u = f_1 \cdots f_n g$, $n \geq 0$, where $f_i \in N_{aF}$ and $g \in N_{cF}$, and new concepts are constructed using $\exists u_1, \ldots, u_n.P$, $P \in \Phi_{\mathcal{D}}$, and $g \uparrow$, with the following semantics:

$$(\exists u_1, \ldots, u_n.P)^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} \mid \exists x_1, \ldots, x_n \in \Delta_{\mathcal{D}} : u_i^{\mathcal{I}}(d) = x_i, (x_1, \ldots, x_n) \in P^{\mathcal{D}}\}$$

$$(g \uparrow)^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} | g^{\mathcal{I}}(d) \textit{ is undefined}\}$$

DLs with constructors of Table 1 extended with concrete domains are denoted by $\mathcal{ALC}(\mathcal{D})$.

The decidability of a DL is a critical question to consider. DLs $\mathcal{ALC}(\mathcal{D})$ for general TBoxes are not decidable (see Lutz 2003), so we cannot expect to implement a reasoner for those DLs. The first condition to make $\mathcal{ALC}(\mathcal{D})$ decidable is to consider *admissible* concrete domains, that is, concrete domains where the set of predicates Φ_D is closed under negation and contains a name $\mathrm{T}_{\mathcal{D}}$ for $\Delta_{\mathcal{D}}$, and the satisfiability of conjunctions of predicates is decidable (for a formal definition, see Baader and Hanschke 1991). The second condition refers to the length of paths u in formulas $\exists u_1, \ldots, u_n.P$: Only *path-free* compositions are considered, that is, each composition must have the form $u = g.P$, where g is a concrete feature in N_{cF} (in the examples above, $(\textit{wife age}, \textit{age}).=$ is not a path-free composition, and $\textit{age}.\leq_{25}$ is a path-free composition). More precisely, it is proved in Haarslev *et al.* (2001) that DLs $\mathcal{SHN}(\mathcal{D})$ (which extends $\mathcal{ALC}(\mathcal{D})$ with transitive roles, role hierarchies and unqualified number restriction) are decidable for admissible concrete domains and path-free compositions. As we will see later on, to formalize semantic granules and semantic granule maps we need only DL $\mathcal{ALC}(\mathcal{D})$ with transitive roles and path-free compositions. Therefore, we ensure the existence of a suitable reasoner for our needs.

Another interesting question is the performance of the reasoner with the type of DL that we will use, $\mathcal{ALC}(\mathcal{D})$. It is proved in Lutz (2003) that if the complexity of the satisfiability problem in the concrete domain $\mathcal{D}$ is PTime, then the satisfiability problem in $\mathcal{ALC}(\mathcal{D})$ is PSpace-complete. To ensure that the satisfiability problem in the concrete domain is in PTime, we restrict our concrete domain to real numbers with linear equations. This implies that the only regions that we can express in our DLs are regions composed of polygons. This restriction is not so important because : (i) many areas are described in real scenarios with polygons (as in SVG format files); and (ii) linear equations are supported in OWL 2 (Parsia and Sattler 2009) and are implemented by reasoners such as RacerPro and HermiT.

3. Location granules and granule maps

Informally, a *location granule* is composed of one or more geographic areas which identify a set of GPS locations under a common name and a *location granule map* is a set of granules. On the one hand, location granules could be retrieved as part of the query projections (i.e., in the SELECT clause of an SQL-like query), therefore affecting the presentation of the results. On the other hand, it is possible to specify the use of location granules in the location-dependent constraints themselves (i.e., in the WHERE clause of the query); in this case, the use of location granules affects the semantics of the constraint because the suitable granularity has to be considered to process the constraint. In this article, we focus on semantic aspects, and so on the use of location granules in location-dependent constraints.

In this section, we formally define the concepts of location granule and location granule maps, and we introduce the type of queries that we will extend using a reasoner.

Let $(\mathbf{R}^n, d)$ be the n-dimensional Euclidean space with a defined Euclidean distance $d : \mathbf{R}^n \times \mathbf{R}^n \to \mathbf{R}$. Typically, we consider the two-dimensional Euclidean space of GPS locations. Given a set S, we denote by $\mathcal{P}(S)$ the power set of S, that is, the set of all subsets of S. Using a distance between points, we can define a distance between points and sets, $d : \mathbf{R}^n \times \mathcal{P}(\mathbf{R}^n) \to \mathbf{R}$, where $d(g, G) = \mathit{inf}\{d(g, g') \mid g' \in G\}$, and a distance between sets $d : \mathcal{P}(\mathbf{R}^n) \times \mathcal{P}(\mathbf{R}^n) \to \mathbf{R}$, where $d(G, G') = \mathit{inf}\{d(g, G') \mid g \in G\}$.

Definition 3.1: A *location granule* (or simply a granule) is a tuple $(G, name_G)$ where G is a subset of $\mathbf{R}^n$ and $name_G$ is a string to identify the granule. An element $g \in G$ is called a location point.

Definition 3.2: A *granule map* is a tuple $(M, name)$ where M is a finite set of location granules and *name* is a string to identify the map. We will use the following notation: $M_G = \{G \mid (G, name_G) \in M\}$.

For our purpose, we will not consider that a granule G can be any subset of $\mathbf{R}^n$. Instead, we restrict to granules G that can be expressed with linear formulas using the arithmetic operators $+$ and $*$ and the relational predicates $<, =, >$, etc. For example, in R^2, the formula $(2 * x - y \leq 5) \wedge (3 * x + y \leq 8) \wedge (x >= 0)$ represents a triangle. With this restriction, we can define any kind of polygon, and at the same time, as we discussed in Section 2, we will obtain a good performance of the reasoners.

In Ilarri *et al.* (2011a), an inside constraint using granules and granule maps is defined and it is used to ask for objects o of a certain type (e.g., cars and ambulances) that have an attribute $o.loc$ which defines its location. There are three types of *inside* constraints defined in Ilarri *et al.* (2011a); in the following $r \in \mathbf{R}$, l is a location point, G is a granule, and M is a granule map:

$$\mathit{inside}(r, G, \mathit{type}) = \{o \in \mathit{type} \mid d(o.loc, G) \leq r\}$$

$$\mathit{inside}(r, l, M, \mathit{type}) = \{o \in \mathit{type} \mid o.loc \in G_i, d(l, G_i) \leq r, G_i \in M\}$$

$$\mathit{inside}(r, G, M, \mathit{type}) = \{o \in \mathit{type} \mid o.loc \in G_i, d(G_i, G) \leq r, G_i \in M\}$$

Examples of these queries are retrieve museums within 10 miles of a given province, museums in provinces which are within 10 miles from a given point, and museums in provinces which are within 10 miles from a given province. We illustrate the different types of inside constraints in Figure 1. Observe that we will extend these queries to an appropriate DL.

4. Semantic location granules and semantic granule maps

To formalize the concepts of semantic granules and semantic granule maps with DLs, we will consider that a transitive role named *isContained* and concrete features $loc_{x_1}, \ldots, loc_{x_n}$ are defined in our model. Intuitively, the role *isContained* will be used to express that a granule is geographically contained in another one by subsumption and participation in the relationship (e.g., $NewYork \sqsubseteq \exists isContained.EEUU$), and $locx_1, \ldots, locx_n$ will be the coordinates of a point. These concrete features allow us to define areas, for example,

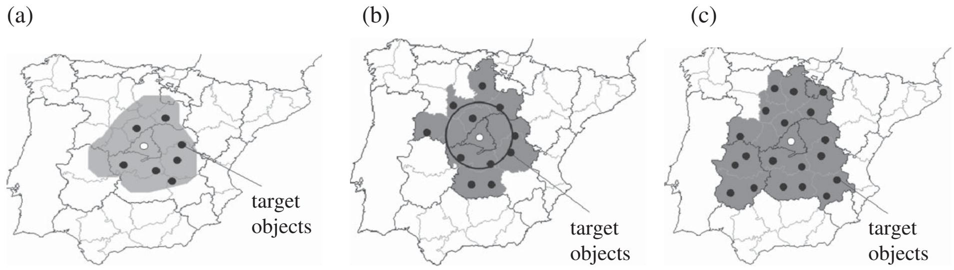

Figure 1. Examples of different types of inside constraints. (a) *inside* (r, G, *type*), (b) *inside* (r, g, M, *type*), and (c) *inside* (r, G, M, *type*).

$loc_x \geq 10 \wedge loc_x \leq 20 \wedge loc_y \geq 20 \wedge loc_y \leq 30$ (if we work in $\mathbf{R}^2$, we write loc_x, loc_y instead of loc_{x_1}, loc_{x_2}).

Definition 4.1: An *area concept* $f(loc_{x_1}, \ldots, loc_{x_n})$ is a concept constructed with operators $\cap$ and $\cup$ and a combination of path-free compositions using concrete features $loc_{x_1}, \ldots, loc_{x_n}$. An *area concept name* A is a concept name such that $A \equiv f(loc_{x_1}, \ldots, loc_{x_n})$, where f is an area concept. The set of names of area concepts is denoted by N_A.

For example, $f(loc_x, loc_y) = \exists loc_x. \geq_{10} \sqcap \exists loc_x. \leq_{20} \sqcap \exists loc_y. \geq_{10} \sqcap \exists loc_y. \leq_{15}$ is an area concept, whereas $\exists loc_x. \geq_{10} \sqcap \exists loc_x. \leq_{20} \sqcap \exists loc_y. \geq_{10} \sqcap \exists loc_y. \leq_{15} \sqcap City$ is not an area concept.

Definition 4.2: Given $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ a knowledge representation and M a granule map, a *semantic granule map* is a tuple $(M, \mathcal{K}, area, semGranule)$, where *area* and *semGranule* are functions from the set of granules of M to the concept names of $\mathcal{T}$; that is, *area*, $semGranule : M_G \rightarrow N_C$, such that the following must be satisfied for all $G \in M_G$:

(1) $semGranule(G) \sqsubseteq area(G)$
(2) $area(G) \sqsubseteq \exists isContained.semGranule(G)$

The concept names *semGranules*(G) are called *semantic granules*.

Let us show an example to explain the definition. Let M be a granule map with location granules $M = \{ZaragozaGr, AragonGr, MadridGr, SpainGr, LibourneGr, FranceGr, BordeauxWineRegionGr, PortugalGr, EuropeGr\}$ and let $\mathcal{T}$ be the following TBox[1] of a knowledge representation $\mathcal{K}$ (see Figure 2):

$$Zaragoza\_Area \equiv \exists loc_x. \geq_{25} \sqcap \exists loc_x. \leq_{30} \sqcap \exists loc_y. \geq_{23} \sqcap \exists loc_y. \leq_{30} \tag{4.1}$$

$$Aragon\_Area \equiv \exists loc_x. \geq_{25} \sqcap \exists loc_x. \leq_{30} \sqcap \exists loc_y. \geq_{20} \sqcap \exists loc_y. \leq_{32} \tag{4.2}$$

$$Madrid\_Area \equiv \exists loc_x. \geq_{15} \sqcap \exists loc_x. \leq_{20} \sqcap \exists loc_y. \geq_{17} \sqcap \exists loc_y. \leq_{23} \tag{4.3}$$

$$Spain\_Area \equiv \exists loc_x. \geq_{5} \sqcap \exists loc_x. \leq_{35} \sqcap \exists loc_y. \geq_{0} \sqcap \exists loc_y. \leq_{35} \tag{4.4}$$

Figure 2. Sample granule map.

$$Libourne\_Area \equiv \exists loc_x.\geq_{25} \sqcap \exists loc_x.\leq_{27} \sqcap \exists loc_y.\geq_{63} \sqcap \exists loc_y.\leq_{68} \tag{4.5}$$

$$BordeauxWR\_Area \equiv \exists loc_x.\geq_{20} \sqcap \exists loc_x.\leq_{30} \sqcap \exists loc_y.\geq_{60} \sqcap \exists loc_y.\leq_{70} \tag{4.6}$$

$$France\_Area \equiv \exists loc_x.\geq_{20} \sqcap \exists loc_x.\leq_{60} \sqcap \exists loc_y.\geq_{35} \sqcap \exists loc_y.\leq_{85} \tag{4.7}$$

$$Portugal\_Area \equiv \exists loc_x.\geq_{0} \sqcap \exists loc_x.\leq_{5} \sqcap \exists loc_y.\geq_{5} \sqcap \exists loc_y.\leq_{30} \tag{4.8}$$

$$Europe\_Area \equiv \exists loc_x.\geq_{0} \sqcap \exists loc_x.\leq_{250} \sqcap \exists loc_y.\geq_{0} \sqcap \exists loc_y.\leq_{250} \tag{4.9}$$

$$Zaragoza \equiv Zaragoza\_Area \sqcap City \tag{4.10}$$

$$Aragon \equiv Aragon\_Area \sqcap Region \tag{4.11}$$

$$Madrid \equiv Madrid\_Area \sqcap City \tag{4.12}$$

$$Spain \equiv Spain\_Area \sqcap Country \sqcap \exists vat.=_{21} \tag{4.13}$$

$$Libourne \equiv Libourne\_Area \sqcap City \tag{4.14}$$

$$BordeauxWR \equiv BordeauxWR\_Area \sqcap WineRegion \tag{4.15}$$

$$France \equiv France\_Area \sqcap Country \sqcap \exists vat. =_{19} \tag{4.16}$$

$$Portugal \equiv Portugal\_Area \sqcap Country \sqcap \exists vat. =_{23} \tag{4.17}$$

$$Europe \equiv Europe\_Area \sqcap Continent \tag{4.18}$$

$$Zaragoza\_Area \sqsubseteq \exists isContained.Zaragoza \tag{4.19}$$

$$Aragon\_Area \sqsubseteq \exists isContained.Aragon \tag{4.20}$$

$$Madrid\_Area \sqsubseteq \exists isContained.Madrid \tag{4.21}$$

$$Spain\_Area \sqsubseteq \exists isContained.Spain \tag{4.22}$$

$$Libourne\_Area \sqsubseteq \exists isContained.Libourne \tag{4.23}$$

$$BordeauxWR\_Area \sqsubseteq \exists isContained.BordeauxWR \tag{4.24}$$

$$France\_Area \sqsubseteq \exists isContained.France \tag{4.25}$$

$$Portugal\_Area \sqsubseteq \exists isContained.Portugal \tag{4.26}$$

$$Europe\_Area \sqsubseteq \exists isContained.Europe \tag{4.27}$$

$$Region, Country, Continent, City, WineRegion \text{ are mutually disjoint} \tag{4.28}$$

$$RedWine \sqsubseteq Wine \tag{4.29}$$

$$RedWine \sqsubseteq \exists isTypical.\exists isContained.Aragon \tag{4.30}$$

$$RedWine \sqsubseteq \exists isTypical.\exists isContained.France \tag{4.31}$$

$$MadridShop \equiv Shop \sqcap \exists isContained.Madrid \tag{4.32}$$

$$AragonWine \equiv Wine \sqcap \exists isContained.Aragon \tag{4.33}$$

$$BourdeauxWine \equiv Wine \sqcap \exists isContained.BourdeauxWR \tag{4.34}$$

$$LibourneWine \equiv Wine \sqcap \exists isContained.Libourne \tag{4.35}$$

$$AragonWine \sqsubseteq \exists hasSimilarWine.BourdeauxWine \tag{4.36}$$

$$BourdeauxWine \sqsubseteq \exists hasSimilarWine.LibourneWine \tag{4.37}$$

$$WineShop \equiv Shop \sqcap \forall sell.Wine \tag{4.38}$$

$$AragonWineShop \equiv WineShop \sqcap \exists isContained.Aragon \tag{4.39}$$

$$ZaragozaWineShop \equiv WineShop \sqsubseteq \exists isContained.Zaragoza \tag{4.40}$$

$$AragonWineShop \sqcap \exists sell.AragonWine \tag{4.41}$$

We define a semantic granule map, $(M, \mathcal{K}, area, semGranule)$, where *area* and *semGranule* are the obvious functions *area*(*SpainGr*) $=$ *Spain_Area*, etc., and *semGranule*(*SpainGr*) $=$ *Spain*, etc. We can ensure that this is a semantic granule map since it holds the conditions (1) and (2) of Definition 4.2 from axioms (4.10)–(4.18), and (4.19)–(4.27), respectively.

Intuitively, the condition (1) of Definition 4.2 says that a semantic granule is not only its geographic area but it could also have more attributes ($semGranule(G) = area(G) \sqcap C$). For example, *Zaragoza* is an area and a *City*, and *Spain* is an area and a *Country*. The condition (2) allows to establish qualitative relations between granules such as '*Zaragoza* is a city in *Spain*', that is, $Zaragoza \sqsubseteq \exists isContained.Spain$, or to express the concept 'Aragon's wines', $Aragon\_Wine \equiv Wine \sqcap \exists isContained.Aragon$. Let us note that we do not express the concept Aragon's wine as $Wine \sqcap Aragon$, since *Aragon* is a *Region* and wines are not regions; and similarly, Aragon's wines are not defined as $Wine \sqcap Aragon\_Area$ because of wines could not have location information. Therefore, we have divided the containment relationship in two parts: one to make calculations about areas (quantitative reasoning) using the subsumption relationship, and another one to establish relationships with other concepts (qualitative reasoning) using the *isContained* relationship.

We emphasize that two different functions are needed in the definition of semantic granule map to distinguish the area of a granule from the semantic granule. Usually, granules have attributes which are incompatible. For example, *Zaragoza* is a city and *Spain* is a country, and a city cannot be a country. So, let us suppose that we do not differentiate between the area of a granule and the granule itself. In this case, for example, it would be held that $Zaragoza\_Area \equiv Zaragoza$ and $Spain\_Area \equiv Spain$. Thus, the TBox $\mathcal{T}$ would be inconsistent since $Zaragoza \sqsubseteq City$ and $Zaragoza \sqsubseteq Spain \sqsubseteq Country$, and Zaragoza is not a country.

Let us note that a reasoner can deduce a number of facts, as we explain in the following.

Proposition 4.3: *A reasoner under conditions of Definition 4.2 can infer that:*

(1) *a granule G is contained in a granule G', that is, it can be deduced that* $semGranule(G) \sqsubseteq \exists isContained.semGranule(G')$
(2) *a granule G intersects granule G'*

Proof: From the definition of the area concept, using concrete domains, a reasoner can infer that $area(G) \sqsubseteq area(G')$, and therefore that:

$$semGranule(G) \sqsubseteq area(G) \sqsubseteq area(G') \sqsubseteq \exists isContained.semGranule(G')$$

In the example above, it can be inferred that *Zaragoza* is contained in *Spain*, $Zaragoza \sqsubseteq Zaragoza\_Area \sqsubseteq Spain\_Area \sqsubseteq \exists isContained.Spain$.

The second statement is obvious since it is equivalent to asking if the area concept $area(G) \sqcap area(G')$ is satisfiable. □

Another interesting remark is that the content of a granule only depends on its area. What does this mean? For example, let us suppose that we define two different semantic granules with the same area, $VaticanCity \equiv Vatican\_Area \sqcap City$ and $VaticanCountry \equiv Vatican\_Area \sqcap Country$. We would like that the content of Vatican as a city, $\exists isContained.VaticanCity$, be equal to the content of the Vatican as a country, $\exists isContained.VaticanCountry$, even when a country is not a city. Due to the conditions (1) and (2) of the Definition 4.2 and the transitivity of the role *isContained*, we can conclude that in our model $\exists isContained.VaticanCity \equiv \exists isContained.VaticanCountry$, as it is proved in the following proposition.

Proposition 4.4: *Let G be a granule under conditions of Definition 4.2. Then, it holds that*

$$\exists isContained.semGranule(G) \equiv \exists isContained.area(G).$$

And therefore, if G_1 *and* G_2 *are granules such that* $area(G_1) \equiv area(G_2)$, *then*

$$\exists isContained.semGranule(G_1) \equiv \exists isContained.semGranule(G_2).$$

Proof: By condition (1) of Definition 4.2, $semGranule(G) \sqsubseteq area(G)$, and so $\exists isContained.semGranule(G) \sqsubseteq \exists isContained.area(G)$. Now, let us prove that $\exists isContained.area(G) \sqsubseteq \exists isContained.semGranule(G)$, obtaining the desired result. Using condition (2) of Definition 4.2 and the transitivity of role *isContained*, we can deduce

$$\exists isContained.area(G) \sqsubseteq \exists isContained.\exists isContained.semGranule(G) \sqsubseteq$$

$$\sqsubseteq \exists isContained.semGranule(G).$$

The second part of the proposition is trivial. □

Let us also note that a reasoner can infer that the VAT applied in Madrid's shops is 18 (the same one than in Spain and different from other countries in Europe), that is, $MadridShop \sqsubseteq \exists isContained.\exists vat. =_{18}$ using the last proposition:

$$MadridShop \sqsubseteq \exists isContained.Madrid \equiv \exists isContained.Madrid\_Area \sqsubseteq$$

$$\sqsubseteq \exists isContained.Spain\_Area \equiv \exists isContained.Spain \sqsubseteq \exists isContained.\exists vat. =_{18}$$

To finalize this section, we would like to introduce another example in a different scenario, this time in R^3. We will use three concrete features, instead of two, *locx*, *locy*, and *locz* to express the three coordinates of the space. Let us suppose that we are looking for a person, *John*, in a university campus. A GPS could give us the exact localization of *John*, the coordinates x, y, and z of his position. This information is not very useful for a final user, since he/she would like to know in what department or building of the campus John is. To solve this problem, we have to mix some geographical information (GPS coordinates) with some semantic information (department of mathematics, building of medicine, . . .). If we consider that we have the following TBox:

$$ScienceBuilding \equiv \exists locx. \geq_3 \sqcap \exists locx. \leq_{10} \sqcap \exists locy. \geq_2 \sqcap \exists locy. \leq_{15} \sqcap \exists locz. \geq_8 \exists locz. \leq_{12}$$

$$Floor2ScienceBuilding \equiv \exists locx. \geq_5 \sqcap \exists locx. \leq_8 \sqcap \exists locy. \geq_8 \sqcap \exists locy. \leq_{12} \sqcap \exists locy. =_{10}$$

$$PhysiologyDepartment \sqsubseteq Floor2ScienceBuilding \sqcap Department$$

$$Floor2ScienceBuilding \sqsubseteq \exists isContained.PhysiologyDepartment$$

we can now deduce that *John* is in the *PhysiologyDepartment* introducing in the ABox the location of *Jonhn*, $locx(John) = 6$, $locy(John) = 9$, $locz(John) = 10$, since the reasoner would classify *John* as instance of the concept $\exists isContained.PhysiologyDepartment$. Let us note that we can also deduce that *John* is an instance of $\exists isContained.ScienceBuilding$ using the transitivity of the *isContained* rol. So, we could show to the user, depending on the level of granularity that we would want to use, that John is in the Department of Physiology or in the Science building of the campus.

5. Using *inside* constraints and the ABox

In this section, we explain how to use a reasoner to increase the functionalities of the *inside* constraints explained in Section 3.

Let $A = f(loc_{x_1}, \ldots, loc_{x_n})$ be an area concept, we define the r-buffer area concept, $Buffer_r(A)$, as the area concept that extends the area of A in all directions by $r > 0$ units. For example, if $A = \exists loc_x. \geq_5 \sqcap \exists loc_x. \leq_{10}$, then $Buffer_3(A) = \exists loc_x. \geq_2 \sqcap \exists loc_x. \leq_{13}$. It is obvious that it holds that $f(loc_{x_1}, \ldots, loc_{x_n}) \sqsubseteq Buffer_r(f(loc_{x_1}, \ldots, loc_{x_n}))$.

5.1. Extending the constraint inside(r, G, M, type)

As a running example, let us suppose that we would like to retrieve countries that produce red wine and are located within 500 miles from Zaragoza. Note that this type of query cannot be solved with an *inside* constraint since red wine is not a location attribute; instead, it is a type of drink. A simple *inside* constraint only involves objects with a known location. First, the above TBox $\mathcal{T}$ is enlarged with a new axiom to express the buffer area of *Zaragoza*:

$$Buffer\_Zaragoza\_Area \equiv Buffer_{500}(Zaragoza\_Area)$$

Now, a reasoner can help us to deduce, for example:

(1) Which area concepts intersect the buffer area. The following set of area concept names is constructed:

$$Inter_A(G, r) = \{area(G') \mid G' \in M, area(G') \sqcap Buffer_r(area(G)) \text{ is satisfiable}\}$$

In our case, $Inter_A(Zaragoza, 500)$ is equal to *Madrid_Area*, *Zaragoza_Area*, *Spain_Area*, *France_Area*, *Portugal_Area*, and *Europe_Area*.

(2) Which concepts contain the area concept names $Inter_A(G, r)$. The following set of concepts is found:

$$Inter(G,r) = \{H \mid \exists F \in Inter_A(G,r), F \sqsubseteq \exists isContained.H\}$$

In the example, *Inter*(*Zaragoza*, 500) is equal to *Madrid*, *Zaragoza*, *Spain*, *France*, *Portugal*, and *Europe*.

(3) Which concepts of *Inter*(*Zaragoza*, 500) are countries, that is, if it is held that $C \sqsubseteq Country$ for some $C \in Inter(Zaragoza, 500)$. Thus, it is retrieved *Spain*, *France*, and *Portugal*.

(4) And finally, if there exists a concept C in the last step such that $RedWine \sqsubseteq \exists isTypical.\exists isContained.C$. The final result would be *Spain* and *France*.

We can summarize the queries for the reasoner as follows:

$$concepts\ ?x, ?y :?x \sqsubseteq Country; RedWine \sqsubseteq \exists isTypical.\exists isContained.?x$$

$$?x \sqsubseteq ?y;\ \ ?y \sqcap Buffer\_Zaragoza\_Area\ \ is\ satisfiable$$

Observe that the transitivity of the role *isContained* is necessary to infer that red wine is a typical type of wine in *Spain*:

$$RedWine \sqsubseteq \exists isTypical.\exists isContained.Aragon \sqsubseteq$$

$$\sqsubseteq \exists isTypical.\exists isContained.\exists isContained.Spain \sqsubseteq\ \ (by\ transitivity)$$

$$\sqsubseteq \exists isTypical.\exists isContained.Spain$$

5.2. *Extending the constraint* inside(r, l, M, type)

In this case, we need to solve queries like '*Retrieve regions which have red wines and that are located within 100 miles from a location l*'. Let us suppose that the coordinates for location l are (10,15). The TBox is enlarged with axioms

$$Here\_Area = \exists loc_x. =_{10} \sqcap \exists loc_y. =_{15} \tag{5.1}$$

$$Buffer\_Here\_Area \equiv Buffer_{100}(Here\_Area) \tag{5.2}$$

Now, we proceed in the same way as before: first, we retrieve the area concepts that intersect *Buffer_Here_Area*; second, we find what concepts hold that are subsumed by $\exists isContained.H$, where H is an area concept found in the previous step; and, finally, we obtain which of the last retrieved concepts, C, are regions that $RedWine \sqsubseteq \exists isTypical.\exists isContained C$. The queries for the reasoner could be summarize as follows:

$$concepts\ ?x, ?y :?x \sqsubseteq Region :\ \ RedWine \sqsubseteq \exists isTypical.\exists isContained.?x$$

$$?x \sqsubseteq ?y; ?y \sqcap Buffer\_Here\_Area \sqsubseteq\ \ is\ satisfiable$$

5.3. Exploiting the ABox

So far, the ABox has not been exploited, since areas have been described as concepts in the TBox. Another feasible way to describe locations is through instances in an ABox. An instance o of a semantic granule in our model is: first, since $o \in area(G)$, it has a value for $loc_{x_1}, \ldots, loc_{x_n}$, and we can see o as a location point; second, since $o \in semGranule(G)$, it has some other qualities (like Region, Country), so we can consider o as a *semantic location point*. Let us show an example where the ABox is used. *Thierry* is a Frenchman visiting Zaragoza city. He is in front of a wine shop and would like to know if he can buy a bottle of wine of similar quality to wines from his home city, Libourne. We define a transitive role, *hasSimilarWine*, to express that two types of wines have similar quality.

If we consider the TBox defined by axioms (4.1)–(4.41) and we assert in the ABox the *wineshop*1 instance:

$$WineShop(wineshop1); loc_x(wineshop1) = 27; loc_y(wineshop1) = 28$$

then, the reasoner deduces (see Figure 3) that the instance *wineshop*1 is located in *Zaragoza_Area*, and therefore in *Aragon_Area*, which is contained in *Aragon* (as a region). Moreover, by axiom 4.41, *wineshop*1 sells some Aragon's wines which are similar to Bourdeaux's wines and Libourne's wines by axioms 4.36 and 4.37. Therefore, *wineshop*1 sells wines similar to the ones from Libourne. Let us note that we did not assert that wines from *Zaragoza* and *Libourne* are of similar quality, but we have just introduced an axiom in the TBox to express that the *Aragon* and *BordeauxWR* wines are similar. The system, by reasoning over the relations *isContained*, *hasSimilarWine*, and *isA* relations, has inferred the rest.

We would like to remark the difference between an instance and a concept in our model. An instance would be any specific location point with attached semantics. For example, an instance o of the concept $Spain \equiv Spain\_Area \sqcap Country$ would be any point in Spain being considered as a location in a country. So, the instance o, in fact, only has the information of the coordinates of the point. However, we can access all the information about the *Spain* concept because it is asserted that o belongs to it. Thus, we could retrieve the boundaries of the country, the different defined properties, and so on.

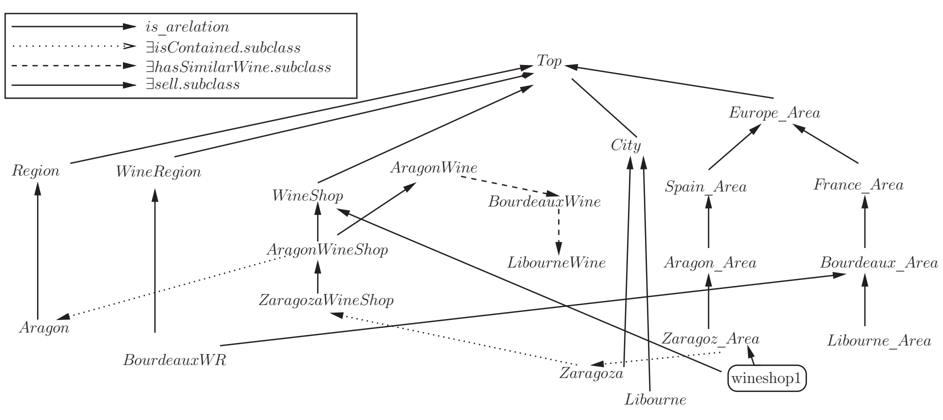

Figure 3. Subset of the ontology to infer Thierry's query.

5.4. Other spatial relationships

So far, we have focused on the inside constraint, as this is a very useful and typical constraint for location-dependent queries. However, other relationships can also be easily supported by our proposed framework. In this section, we briefly overview some of these relationships.

A formal definition for binary topological relationships in geographic databases is given in Egenhofer (1989), Egenhofer and Franzosa (1991), and Egenhofer and Herring (1991). According to these definitions, the authors distinguish a set of basic relations between regions depending on the interior, exterior, and frontier of the regions (see Figure 4, i.e., the interior, exterior, and frontier of the granules). The next proposition states how we can use a reasoner to discover some of these relationships. The proof is straightforward from the definitions.

Proposition 5.1: *Let G_1 and G_2 be granules of a granule map M,* $(M, \mathcal{K}, area, semGranule)$ *a semantic granule map, and A_1 and A_2 are the area concepts in $\mathcal{K}$ such that* $area(G_i) = A_i$ $(i = 1, 2)$. *Then:*

(1) *if $A_1 \equiv A_2$, then the area of the granule G_1 is equal to the area of the granule G_2, that is, $equal(G_1, G_2)$ is true.*
(2) *if $A_2 \sqsubseteq A_1$ and $\neg A_2 \sqcap A_1$ are satisfiable, then $contain(G_1, G_2)$ or $cover(G_1, G_2)$ holds.*
(3) *if $(A_1 \sqcap A_2) \sqsubseteq \bot$, then $disjoint(G_1, G_2)$ holds.*
(4) *if $A_1 \sqcap A_2$, $\neg A_1 \sqcap A_2$ and $A_1 \sqcap \neg A_2$ are satisfiable, then $overlap(G_1, G_2)$ or $meet(G_1, G_2)$ holds.*

Let us note that we cannot easily use the reasoner to distinguish between contain/cover and overlap/meet relationships, since the reasoner should automatically obtain the frontier of the area concepts. In fact, the *isContained* relation used in this article is the logical disjunction of the *contain* and the *cover* relationships in Figure 4. Also note that we have not considered the inside/coveredBy relationships in the proposition, as these relationships are the inverse of contain/cover. Finally, it is necessary that A_1 and A_2 in the above proposition be area concepts. For example, if A_1 is equal to *VaticanCity* (the intersection of the area concept *Vatican_Area* and the concept *City*) and A_2 is *VaticanCountry* (the intersection of

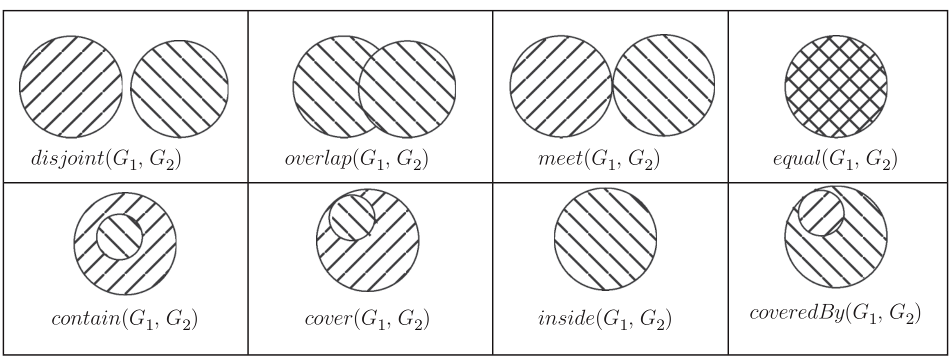

Figure 4. Set of binary topological relationships. Adapted from Egenhofer (1989), used with permission.

concepts *Vatican_Area* and *Country*), we get that $A_1 \sqcap A_2 \sqsubseteq \bot$ (since concepts *City* and *Country* are disjoint). So, applying the above proposition, we would obtain that the areas A_1 and A_2 are disjoint, which is obviously false. If we consider only the area concepts, we will get the right conclusion: A_1 and A_2 are equal.

6. Related work

As we have seen, the problem of formalizing semantic granules is closely related to spatial reasoning with DL languages. There are different approaches in the literature to tackle the spatio-semantic reasoning problem with DLs. In Haarslev *et al.* (1998), it is defined a new role forming operation in DLs that leads to a new type of DL: $\mathcal{ALCRP}(D)$. The authors use this DL for spatioterminological reasoning with the concrete domain of polygons. Some constraints are imposed on the new role forming operator to obtain the decidability of $\mathcal{ALCRP}(D)$. The *contains* relation is then defined using the new operator. However, problems would arise if a real reasoner should be used, since no one uses the role forming operator and the polygons concrete domain.

In Wessel and Möller (2009), the authors present a practical framework for building ontology-based systems applied to GIS. They propose to associate the ABox with an SBox (spatial box) which includes geometric information, called *map substrate*. Then, a special query language is used to obtain information from it. Contrary to our model, they make an intensive use of the ABox to infer geometrical information, leaving the TBox mainly for semantic inferences.

Another way to capture semantics and spatial information together is using Region Connection Calculus (RCC, Randell *et al.* (1992)). The relations it proposes have been an intensive case of study. However, there is no unique way of modeling them using ontologies. In Katz and Grau (2005), RCC relations are presented as concept axioms in OWL DL (Bechhofer *et al.* 2004). In this case, the *isContained* relation is defined as a subsumption relation $\sqsubseteq$, which implies, as we have seen before, unexpected inferences. RCC relations considered as roles in DLs have been researched more extensively. In Wessel (2002), new families of DLs using RCC are defined, $\mathcal{ALCI}_{\mathcal{RCC}8}$ and $\mathcal{ALCI}_{\mathcal{RCC}5}$, demonstrating that they can be undecidable. If we express RCC with roles, it is natural to express relations between RCC components with role axioms like $R \circ S \subseteq R_1 \sqcup \ldots R_n$. It is proved in Horrocks and Sattler (2004) that $\mathcal{SHIQ}$ ($\mathcal{ALC}$ extended with role hierarchy, inverse roles and qualified cardinality restrictions) with complex role inclusions is undecidable. An algorithm is used to refine the ABox and obtain new knowledge in Grütter *et al.* (2008). The algorithm asserts new RCC relations $P(x, y)$ (part of, contains) in the ABox, if the instances related hold an hypothesis and the new assertion is consistent with the TBox. However, with this approach, we need an external entity to assert facts that in our model are inferred automatically.

A different approach to realize spatial relations trying to avoid the undecidability problems is the use of rules. In Chen *et al.* (2005), SWRL (Sematic Web Rule Language) is used to encode spatial relations. SWRL is composed of Horn rules which do not have enough expressivity for complex relations and can only be applied in ABoxes.

Regarding spatial databases, the problem of modeling spatial data at different levels of detail has been addressed from a different point of view (e.g., Fonseca *et al.* 2002, Camossi *et al.* 2003a). These works focus on the different possible levels of detail/specification of spatial entities, leaving aside their locations and their implicit semantics. There are also works that have considered locations at different granularity levels but with no semantic information attached, such as Camossi *et al.* (2003b, 2008). However, the authors of these works focus on inner data representation, the access to it, and its conversion at different representation levels, thus leaving aside the semantics and inferring capabilities that our model exhibits. In Belussi *et al.* (2009), the authors provide a framework to define spatio-temporal granularities (as well as a set of relationships between granularities) and use them in Database Management Systems, but again they are more focused on data representation than on capturing the semantics and enabling reasoning capabilities.

Regarding pervasive computing, many efforts have also been done in modeling locations, as the location is an important part of the user's context. In Stevenson *et al.* (2009), the authors propose *Ontonym*, a set of ontologies that model different aspects of context. In particular, they adopt the location model presented in Ye *et al.* (2007), which unifies both *hierarchical* and *geometrical* dimensions of locations in the same model. However, they focus on representing locations and spatial relationships, leaving aside the nonspatial semantics that might be behind them. Moreover, these proposals delegate to the developer (or to an external system) the effort of establishing the relationships and modeling the containment hierarchies.

Summing up, as opposed to our model, most of the proposed models in the literature use essentially the ABox to encode spatial relations, which implies the need of having a method outside the reasoner to assert many of the spatial relations, or they are mainly focused on data representation, neglecting the potential semantics behind locations expressed at different granularities.

7. Conclusions and future work

In this article, we have proposed a formal model that comprises both the spatial and semantic dimensions of locations seamlessly. For this purpose, we have adopted DLs as a base formalism to have all its expressive power and reasoning capabilities. In particular, our approach has the following features:

- We give a formalization for semantic granules and semantic granule maps using a type of DLs $\mathcal{ALC}(D)$ with transitive roles, which is supported by almost all the existing reasoners (Pellet, Racer Pro, HermiT, etc.).
- Our interpretation of *isContained* and *intersects* relations enables inferring new knowledge while avoiding obtaining wrong conclusions due to the geographic inclusion of concepts.
- It uses a reasoner to extend the expressivity of *inside* constraints.
- It exploits the capability of reasoners to infer implicit knowledge over a TBox better than other models based on asserting relations in an ABox explicitly.
- It allows us to express non-geographical information about the different locations and reason about it along with the spatial information.

As future work, we plan to study how to integrate more spatial relationships in the model to enable automatic reasoning about it. Above all, we will analyze how to introduce RCC relations other than the *isContained* relation, like inner and outer tangential relations and, what is more interesting, how to model the dependency between all the RCC relations.

Another interesting question to solve is how to introduce distances between areas in our model. Since we consider that granules are concepts instead of instances, it is difficult to formalize a distance between granules. We would need a function from concepts to real numbers, and, as far as we know, there is no defined DLs in the literature with such type of functions.

Acknowledgement

This work has been supported by the CICYT (Comisión Interministerial de Ciencia y Tecnología) project TIN2010-21387-C02-02 and DGA-FSE (Diputación General de Aragón-Fondo Social Europeo).

Note

1. To keep explanations easier to follow, we represent geographic areas in the TBox by simple rectangles instead of the real geographic limits.

References

Baader, F., 2007. *The description logic handbook: theory, implementation, and applications*. Cambridge: Cambridge University Press.

Baader, F. and Hanschke, P., 1991. A scheme for integrating concrete domains into concept languages. *In*: J. Mylopoulos and R. Reiter (Eds.), *Proceedings of the 12th international joint conference on artificial intelligence (IJACAI'91)*. Burlington, MA: Morgan Kaufmann, 452–457.

Bechhofer, S., *et al*., 2004. *OWL web ontology language reference* [online]. Available from: http://www.w3.org/TR/owl-ref/ [Accessed April 2011].

Belussi, A., Combi, C., and Pozzani, G., 2009. Formal and conceptual modeling of spatiotemporal granularities. *In*: B.C. Desai, D. Sacca and S. Greco (Eds.), *Proceedings of the 13th international database engineering and applications symposium (IDEAS'09)*. New York: ACM, 275–283.

Bobed, C., Ilarri, S., and Mena, E., 2010. Exploiting the semantics of location granules in location-dependent queries. *In*: B. Catania, M. Ivanovic and B. Thalheim (Eds.), *Proceedings of the 14th East-European conference on advances in databases and information systems (ADBIS'10)*. LNCS, vol. 6295. Berlin: Springer Verlag, 79–93.

Camossi, E., Bertolotto, M., and Bertino, E., 2008. Querying multigranular spatiotemporal objects. *In*: S.S. Bhowmick, J. Kung and R. Wagner (Eds.), *Proceedings of the 19th international conference on database and expert systems applications (DEXA'08)*. LNCS, vol. 5181. Berlin: Springer Verlag, 390–403.

Camossi, E., *et al*., 2003a. Issues on modeling spatial granularities. *In*: *Workshop: fundamental issues in spatial and geographic ontology (COSIT'03)*, September, Ittingen, Switzerland.

Camossi, E., *et al*., 2003b. A multigranular spatiotemporal data model. *In*: D. Pfoser, I.F. Cruz and M. Ronthaler (Eds.), *Proceedings of the 11th ACM international symposium on advances in Geographic Information Systems (GIS'03)*. New York: ACM, 94–101.

Chen, H., Fellah, S., and Bishr, Y.A., 2005. Rules for geospatial semantic web applications. *In*: *Rule languages for interoperability W3C*.

Couclelis, H., 2010. Ontologies of geographic information. *International Journal of Geographical Information Science*, 24 (12), 1785–1809.

Egenhofer, M., 1989. A formal definition of binary topological relationships. *In*: W. Litwin, H.-J. Schek (Eds.), *Foundations of data organization and algorithms*. LNCS, vol. 367. Berlin: Springer, 457–472.

Egenhofer, M.J. and Franzosa, R.D., 1991. Point set topological relations. *International Journal of Geographical Information Systems*, 5 (2), 161–174.

Egenhofer, M.J. and Herring, J., 1991. *Categorizing binary topological relationships between regions, lines, and points in geographic databases*. Orono, ME: Department of Surveying Engineering, University of Maine. NCGIA Technical Report 91-7 and NCGIA Technical Report 94-1.

Fonseca, F., *et al*., 2002. Semantic granularity in ontology-driven geographic information systems. *Annals of Mathematics and Artificial Intelligence*, 36 (1–2), 121–151.

Frank, A., 2003. Ontology for spatio-temporal databases. *In*: M. Koubarakis, T.K. Sellis, A.U. Frank, S. Grumbach , R. Hartmut Guting, C.S. Jensen, N.A. Lorentzos, Y. Manolopoulos, E. Nardelli, B. Pernici, H.-J. Schek, M. Scholl, B. Theodoulidis and N. Tryfona (Eds.), *Chapter 2: Spatio-temporal databases*. Berlin: Springer Verlag, 9–77.

Gruber, T., 1993. A translation approach to portable ontology specifications. *Knowledge Acquisition, An International Journal of Knowledge Acquisition for Knowledge-Based Systems*, 5(2), 199–220.

Grütter, R., Scharrenbach, T., and Bauer-Messmer, B., 2008. Improving an RCC-derived geospatial approximation by OWL axioms. *In*: A.P. Sheth, S. Staab, M. Dean, M. Paolucci, D. Maynard, T.W. Finin and K. Thirunarayan (Eds.), *Proceedings of the 7th international conference on the semantic web (ISWC'08)*. Berlin: Springer Verlag, 293–306.

Haarslev, V., Lutz, C., and Möller, R., 1998. Foundations of spatioterminological reasoning with description logics. *In*: *Proceedings of the 6th international conference on principles of Knowledge Representation and Reasoning (KR'98)*. Burlington, MA: Morgan Kaufmann, 112–123.

Haarslev, V., Möller, R., and Wessel, M., 2001. The description logic ALCNHR+ extended with concrete domains: a practically motivated approach. *In*: *Proceedings of the international joint conference on automated reasoning (IJCAR'01)*. Berlin: Springer Verlag, 29–44.

Horrocks, I. and Sattler, U., 2004. Decidability of SHIQ with complex role inclusion axioms. *Artificial Intelligence*, 160 (1–2), 79–104.

Ilarri, S., Bobed, C., and Mena, E., 2011a. An approach to process continuous location-dependent queries on moving objects with support for location granules. *Journal of Systems and Software*, 84 (8), 1327–1350.

Ilarri, S., Mena, E., and Illarramendi, A., 2010. Location-dependent query processing: where we are and where we are heading. *ACM Computing Surveys*, 42 (3), 1–73.

Ilarri, S., *et al*., 2011b. Semantics in location-based services – guest editors' introduction for special issue. *IEEE Internet Computing*, 15 (6), 10–14.

Katz, Y. and Grau, B.C., 2005. Representing qualitative spatial information in OWLDL. *In*: *Proceedings of OWL: experiences and directions workshop (OWLED'05)*, Galway, Ireland.

Lutz, C., 2003. Description logics with concrete domains – a survey. *In*: P. Balbiani, N.-Y. Suzuki, F. Wolter and M. Zakharyaschev (Eds.), *Advances in modal logics*, vol. 4. London : King's College Publications.

Lutz, C. and Möller, R., 1997. Defined topological relations in description logics. *In*: R. Brachman, F. Donini, E. Franconi, I. Horrocks, A. Levy and M.-C. Rousset (Eds.), *Proceedings of the international workshop on Description Logics (DL'97)*. Burlington, MA: Morgan Kaufmann, 27–29.

Lutz, M. and Klien, E., 2006. Ontology-based retrieval of geographic information. *International Journal of Geographical Information Science*, 20 (3), 233–260.

Parsia, B. and Sattler, U., 2009. *OWL 2 web ontology language data range extension: linear equations* [online]. Available from: http://www.w3.org/TR/owl2-dr-linear [Accessed September 2012].

Randell, D.A., Cui, Z., and Cohn, A., 1992. A spatial logic based on regions and connection. *In*: B. Nebel, C. Rich and W.R. Swartout (Eds.), *Principles of knowledge representation and reasoning: proceedings of the third international conference (KR'92)*. Burlington, MA: Morgan Kaufmann, 165–176.

Rigaux, P., Scholl, M., and Voisard, A., 2002. *Spatial databases with application to GIS*. Burlington, MA: Morgan Kaufmann.

Schiller, J. and Voisard, A., 2004. *Location based services*. Burlington, MA: Morgan Kaufmann.

Shekhar, S. and Chawla, S., 2002. *Spatial databases: a tour*. Upper Saddle River, NJ: Prentice Hall.

Stevenson, G., *et al*., 2009. Ontonym: a collection of upper ontologies for developing pervasive systems. *In*: J. Manuel Gomez-Perez, P. Haase, M. Tilly, and P. Warren (Eds.), *Proceedings of the 1st workshop on context, information and ontologies (CIAO'09)*, vol. 9. New York: ACM, 1–9:8.

Wessel, M., 2002. On spatial reasoning with description logics. *In*: I. Horrocks and S. Tessaris (Eds.), *Proceedings of the international workshop in description logics (DL'02)*. CEUR Workshop Proceedings, 156–163.

Wessel, M. and Möller, R., 2009. Flexible software architectures for ontology-based information systems. *Journal of Applied Logic*, 7 (1), 75–99.

Ye, J., *et al*., 2007. A unified semantics space model. *In*: J. Hightower, B. Schiele and T. Strang (Eds.), *Proceedings of the 3rd international symposium on location- and context-awareness (LoCA'07)*. LNCS, vol. 4718. Berlin: Springer Verlag, 103–120.