University of Huddersfield Repository

Batsakis, Sotiris, Antoniou, Grigoris and Tachmazidis, Ilias

Reasoning over Spatial Orientation Relations Using Rules

Original Citation

Batsakis, Sotiris, Antoniou, Grigoris and Tachmazidis, Ilias (2014) Reasoning over Spatial Orientation Relations Using Rules. In: 18th East-European Conference on Advances in Databases and Information Systems (ADBIS 2014), September 7-10, 2014, Ohrid, Republic of Macedonia. (Unpublished)

This version is available at http://eprints.hud.ac.uk/id/eprint/20910/

Reasoning over Spatial Orientation Relations Using Rules

Sotiris Batsakis, Grigoris Antoniou, and Ilias Tachmazidis

Department of Informatics
University of Huddersfield
Queensgate, Huddersfield, West Yorkshire, HD1 3DH, UK
e-mail:{S.Batsakis,G.Antoniou,Ilias.Tachmazidis}@hud.ac.uk

Abstract. Representation of spatial information for the Semantic Web often involves qualitative defined information (i.e., information described using natural language terms such as “Left”), since precise arithmetic descriptions using coordinates and angles are not always available. A basic aspect of spatial information is directional relations, thus embedding directional spatial relations into ontologies along with their semantics and reasoning rules is an important practical issue. This work proposes a new representation for directional spatial information in ontologies by means of OWL properties and reasoning rules in SWRL embedded into the ontology. The proposed representation is based on the combination of object orientations (e.g., same direction or opposite) and cone shaped directional relations of positions using an egocentric reference (e.g., left or right of an object). The proposed representation is to the best of our knowledge a novel one, and in this work, the proposed representation is analysed, implemented and evaluated.

1 Introduction

Understanding the meaning of Web information requires formal definitions of concepts and their properties, using the Semantic Web Ontology definition language OWL. OWL provides the means for defining concepts, their properties and their relations, and allows for reasoning over the definitions and the assertions of specific individuals using reasoners such as HermiT. Furthermore, reasoning rules can be embedded into the ontology using the SWRL rule language.

Spatial information is an important aspect of represented objects in many application areas. Spatial information in turn can be defined using quantitative (e.g., using coordinates) and qualitative terms (i.e., using natural language expressions such as “Behind”). Qualitative spatial terms have specific semantics which can be embedded into the ontology using reasoning rules. In previous work [1, 6], such a representation is proposed for allocentric (i.e., using an external reference frame, such as North-South) directional relations in OWL.

Current work deals with the case of egocentric directional spatial information, and proposes a new representation for such information. Egocentric directional relations are applied over local reference frames e.g., using terms such as “front” or “left”, that are defined with respect to specific objects and the placement of objects relative to these points of reference. Egocentric orientation relations are analysed into two sets of relations; The first set represents the directional orientation relation between two objects (e.g., “same” or “opposite” direction). This set of relations is a modified form of the relations proposed in [7]. The second set represents the positional orientation relations in terms of the egocentric reference frame of each object (e.g., “front” or “behind”). This set is a modified form of *OPRA* calculi proposed in [8]. Thus, for example if object B is in front of object A and is directed towards it, the following relations hold: *B opposite A* and *B front-of A*. Both relations correspond to cone shaped regions in the plane, and their definitions and semantics are introduced in the current work. Reasoning is applied on directional orientation relations separately, since orientation of directed objects does not depend on the position of one wrt the other (e.g., an object can be directed to an opposite direction wrt another, and simultaneously can be left, right, front or back of it). On the other hand, reasoning over positional orientation relations combines directional orientation relations as well. Current work proposes a reasoning mechanism for the proposed representation. Properties of the reasoning mechanism are analysed and the mechanism is implemented and evaluated. Furthermore, the implementation is based on OWL axioms and SWRL rules embedded into an ontology, thus it is suitable for Semantic Web applications, since reasoning can be achieved using only standard reasoners that support SWRL such as HermiT [9].

Current work is organized as follows: related work in the field of spatial knowledge representation is discussed in Section 2. The proposed representation is presented in Section 3 and the corresponding reasoning mechanism in Section 4 followed by evaluation in Section 5 and conclusions and issues for future work in Section 6.

2 Background and Related Work

Definition of ontologies for the Semantic Web is achieved using the Web Ontology Language OWL[1]. The current W3C standard is the OWL 2[2] language, offering increased expressiveness while retaining decidability of basic reasoning tasks. Reasoning tasks are applied both on the concept and property definitions into the ontology (TBox) and the assertions of individual objects and their relations (ABox). Reasoners include among others Pellet[3], and HermiT[4]. Reasoning rules can be embedded into the ontology using SWRL[5]. To guarantee decidability, the rules are restricted to *DL-safe rules* that apply only on named individuals in the ontology ABox. *Horn Clauses* (i.e., a disjunction of atoms with at most one positive literal), can be expressed using SWRL, since Horn clauses can be written as implications (i.e., $\neg A \vee \neg B ... \vee C$ can be written as $A \wedge B \wedge ... \Rightarrow C$).

Qualitative spatial reasoning (i.e., inferring implied relations and detecting inconsistencies in a set of asserted relations) typically corresponds to Constraint Satisfaction problems which are *NP-hard*, but tractable sets (i.e., solvable by polynomial algorithms)

---

[1] http://www.w3.org/TR/owl-ref/

[2] http://www.w3.org/TR/owl2-overview/

[3] http://clarkparsia.com/pellet/

[4] http://hermit-reasoner.com/

[5] http://www.w3.org/Submission/SWRL/

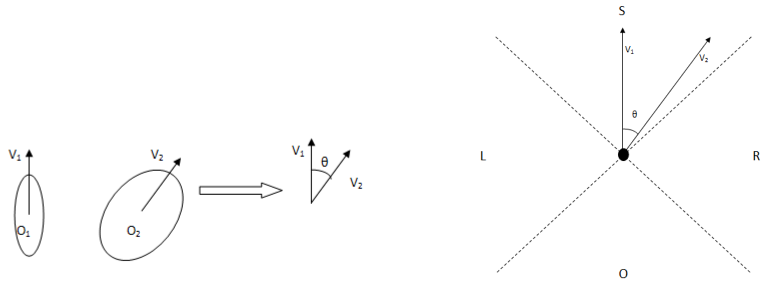
are known to exist [2]. Formal spatial representations have been studied extensively within the Semantic Web community. Relations between spatial entities in ontologies can be topological, directional, or orientation relations. Furthermore, spatial relations are distinguished into qualitative (i.e., relations described using lexical terms such as “Behind”) and quantitative (i.e., relations described using numerical values such as “45 degrees Right”).

Embedding spatial reasoning into the ontology by means of SWRL rules applied on spatial object properties forms the basis of the SOWL model proposed in [1, 6]. Based on the representation proposed in [1] the dedicated Pellet-Spatial reasoner [3] has been extended for directional relations in the CHOROS system [4] (Pellet-Spatial supports only topological relations). None of the above supports orientation relations which typically appear in natural language scene descriptions and in robotics among others. In this work, a representation of orientation relations based on a combination of modified versions of the relations proposed in [7, 8] is proposed. The proposed representation is combined with a tractable reasoning mechanism over specific sets of relations, containing basic relations of both sets that are parts of the mechanism. Furthermore, the reasoning mechanism is implemented by means of OWL axioms and SWRL rules that are fully compliant with existing Semantic Web standards and tools.

3 Spatial Representation

Orientation relations in this work are represented as object properties between OWL objects representing spatial entities. For example if $Object1$ is $Left\ Of\ Object2$, user asserts the binary relation $Object1\ Left\ Object2$, or equivalently *Left(Object1, Object2)*. This approach is similar to the approach used in [1] for cardinal directional relations, as part of the SOWL model. In [7] and [8] orientation relations are defined between objects based on cone-shaped regions around objects. In both cases lines separating the cone-shaped regions are also different relations, similar to the *star calculus* proposed in [5]. Reasoning over *star calculus* have been proven to be *NP-complete*, even if reasoning is restricted over basic relations. On the other hand, lines separating cone-shaped regions can belong to one of these regions instead of being separate basic relations. This calculi is called the *revised star calculus* and it is also presented in [5]. Furthermore reasoning over basic relations of the modified calculi is decided by path consistency and is tractable [5]. This approach is also used for representing cardinal directional relations in [1, 6]. In this work, orientation relations correspond to cone-shaped regions, and lines separating the regions belong to only one of these regions. This is the basic difference to relations proposed in [7] and [8].

Note that representations based on projections on orthogonal 2D axis and reasoning over the pairs of relations on these one-dimensional spaces, instead of cone shaped regions in bi-dimensional space have been proposed as well in [2]. Projection based representations have different semantics than the proposed cone-shaped representation, thus it can not be consider as an alternative to it. For example, using the projection based approach, if a point is located far left relatively to another point and slightly behind it, following the projection based approach relations $Left$ and $Behind$ will hold at the horizontal and the vertical axis respectively. Following the cone-shaped approach



(a) Directional Orientation of Objects (b) Directional Orientation Relations

Fig. 1. (a) Egocentric Directional Orientation of Objects (b) Egocentric Directional Orientation Relations

only the relation $Left$ holds, which is conceptually right according to the way humans usually refer to orientation relations.

The basic directional orientation relations are: *Same, Opposite, Left and Right* as presented in Figure 1(b). These relations are abbreviated as *S,O,L,R* respectively. The relations are defined as follows, if 2D vectors v_1 and v_2 represent the orientation of objects o_1 and o_2 on the 2D plane, then the angle θ_o between vector v_1 and v_2 specify the egocentric orientation relation between o_1 and o_2 as illustrated in Figure 1(a). Note that lines separating the cone shaped regions belong only to one of these regions according to definitions of relations. Specifically:

$$-\frac{\pi}{4} \leq \theta_o < \frac{\pi}{4} \equiv S(o_1, o_2)$$

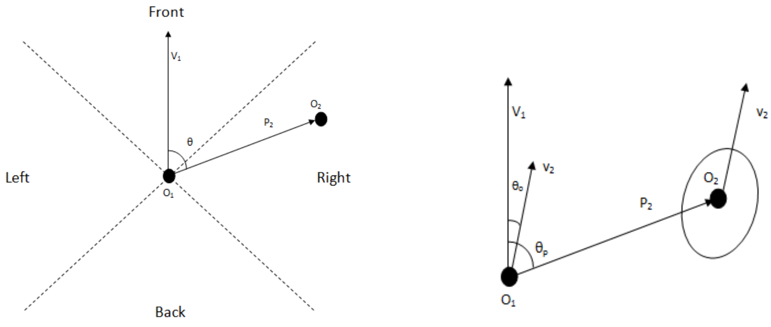
$$\frac{\pi}{4} \leq \theta_o < \frac{3\pi}{4} \equiv R(o_1, o_2)$$

$$\frac{3\pi}{4} \leq \theta_o < \frac{5\pi}{4} \equiv O(o_1, o_2)$$

$$\frac{5\pi}{4} \leq \theta_o < \frac{7\pi}{4} \equiv L(o_1, o_2)$$

Positional orientation relations are *Front, Back, Left and Right*, presented in Figure 2(a) (note that terms *Back* and *Behind* can be used interchangeably). Lines separating the cone-shaped regions belong to only one of the adjacent regions, as in the case of directional orientation relations. By convention, they also belong to the region to the right of the line, moving clockwise (but other conventions are valid as long as each line belongs to exactly one adjacent cone shaped region). Although positional orientation relations seem similar to directional orientation relations, their definition and semantics are different. Specifically, positional orientation relations are defined as follows, if 2D vector p_2 represents the position (and not orientation, as for directional orientation relations) of object o_2 on the 2D plane, that has o_1 position as reference frame, y-axis defined using v_1 (orientation of o_1) and x-axis perpendicular to y-axis, then the angle θ_p between vector v_1 and p_2 specify the egocentric positional orientation relation between o_1 and o_2 as illustrated in Figure 2(a). Relations are defined as follows:

$$
\begin{aligned}
-\frac{\pi}{4} \leq \theta_p < \frac{\pi}{4} &\equiv Front(o_1, o_2) \\
\frac{\pi}{4} \leq \theta_p < \frac{3\pi}{4} &\equiv Right(o_1, o_2) \\
\frac{3\pi}{4} \leq \theta_p < \frac{5\pi}{4} &\equiv Back(o_1, o_2) \\
\frac{5\pi}{4} \leq \theta_p < \frac{7\pi}{4} &\equiv Left(o_1, o_2)
\end{aligned}
$$



(a) Positional Orientation Relations (b) Orientation Example

Fig. 2. (a) Positional Orientation Relations (b) Orientation Example

An example presenting both relations is illustrated in Figure 2(b). Directional orientation relation is defined by angle θ_o between vector v_1 representing the orientation of object o_1 and vector v_2 representing the orientation of object o_2. Positional orientation relation is defined by angle θ_p between vector v_1 representing orientation of object o_1 and vector p_2 representing position of object o_2. In this example, object o_2 has the same orientation as object o_1 and it is located at the right of object o_1.

Additional OWL axioms required for the proposed representation; basic relations of each set are pairwise disjoint e.g., $Left$ is disjoint with $Front$. Also $Left$ is $inverse$ of $Right$ (in both sets) and $Front$ is $inverse$ of $Back$. On the other hand, directional orientation relations $same$ and $opposite$ are symmetric. Note also that if two objects are identical then the equality relation holds between them. Instead of using a separate equality relation the OWL $sameAs$ keyword can be used instead for this case as in [1].

4 Spatial Reasoning

Reasoning is realized by introducing a set of SWRL rules operating on spatial relations. Reasoners that support DL-safe rules such as HermiT can be used for inference and consistency checking over orientation relations. Defining compositions of relations is a basic part of the spatial reasoning mechanism. Table 1 represents the result of the composition of two directional orientation relations of Figure 1(b) (relations $Same$, $Right$, *Opposite* and *Left*, are denoted by "S","R","O", "L" respectively).

Relations	$S(Same)$	$R(Right)$	$O(Opposite)$	$L(Left)$
S	S, R, L	S, R, O	R, O, L	S, O, L
R	S, R, O	R, O, L	S, O, L	S, R, L
O	R, O, L	S, O, L	S, R, L	S, R, O
L	S, O, L	S, R, L	S, R, O	R, O, L

Table 1. Composition Table for Directional Orientation Relations.

Composition Table can be interpreted as follows: if relation R_1 holds between object o_2 and object o_1 and relation R_2 holds between object o_3 and object o_2, then the entry of the Table 1 corresponding to line R_1 and column R_2 denotes the possible relation(s) holding between object o_3 and object o_1. For example, if object o_2 is at $Same$ direction to object o_1 and object o_3 is *Right* (in terms of directional orientation) to object o_2 then object o_3 is *right, same direction or opposite* to object o_1. Entries in the above composition table are determined using the following observation: composition of two relations corresponds to the addition of two angles representing the relative directional orientation of $point2$ to $point1$ and $point3$ to $point2$, forming angles θ_1 and θ_2 respectively with the reference (vertical) axis. Combining this observation with the definition of relations in Section 3 the above compositions of Table 1 are defined. So for example composition of $same$ and $opposite$ is interpreted as adding $\frac{7\pi}{4} \leq \theta_1 < \frac{\pi}{4}$ and $\frac{3\pi}{4} \leq \theta_2 < \frac{5\pi}{4}$ which yields $\frac{2\pi}{4} \leq \theta_{12} < \frac{6\pi}{4}$ which corresponds to a cone shaped region into the region defined by the disjunction of *Right, Opposite, Left*.

Composing positional orientation relations (Figure 2(a)) requires combining also directional orientation relations (Figure 1(b)). Specifically, composing orientation relations can be defined as follows: if object *o1* is related with object *o2* with directional orientation relation $R_{o_{21}}$ and positional orientation relation $R_{p_{21}}$ and object *o2* is related with object *o3* with directional orientation relation $R_{o_{32}}$ and positional orientation relation $R_{p_{32}}$ then between object *o1* and object *o3* the directional orientation relation $R_{o_{31}}$ is defined using the compositions of Table 1 as: $R_{o_{31}} \equiv R_{o_{21}} \circ R_{o_{32}}$ ($\circ$ denotes composition).

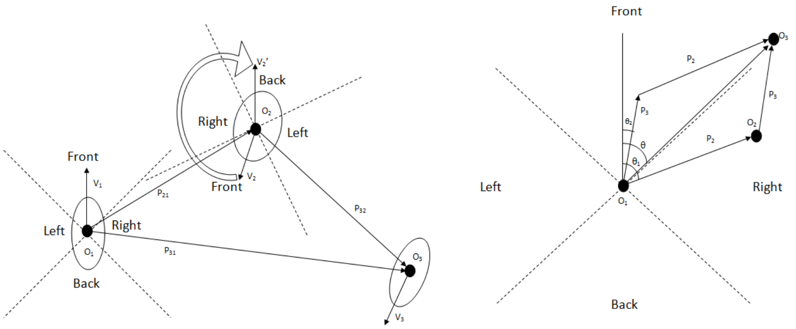
Composition of positional relations is more complex: when composing relations $R_{p_{21}}$ and $R_{p_{32}}$, the fact that object *o2* may have a different directional orientation wrt object *o1* must be also taken into account. For example, if object $o2$ is $Right$ of object $o1$, object $o3$ is $Left$ of object $o2$, but because object $o2$ is $opposite$ of object $o1$ (see Figure 3(a)), directional orientation of objects $o1$ and $o2$ must be aligned, before composing positional relations. Specifically, after rotating object $o2$ wrt object $o1$ (i.e., aligning their directional orientation, by changing direction of object $o2$ from v_2 to $v_2\prime$), then object $o3$ is not considered to be located $Left$ of object $o2$, but $Right$, $Front$ or

$Back$ of object $o2$ (see Table 2). Then we can compose positional orientation relations and infer possible relations holding between objects $o1$ and $o3$.

Intuitively, for the composition of $R_{p_{21}}$ and $R_{p_{32}}$ object *o1* is now the reference point for both object *o2* and object *o3*, thus before composing the two relations, $R_{p_{32}}$ must be adjusted to the reference frame of object *o1* and this is achieved by performing the rotation specified by relation $R_{o_{21}}$. The result of this rotation, which will be described in detail is denoted by $R_{o_{21}} \Diamond R_{p_{32}}$. The resulting relation which is a positional orientation relation of Figure 2(a) can be composed with relation $R_{p_{21}}$, thus $R_{p_{31}} \equiv R_{p_{21}} \circ (R_{o_{21}} \Diamond R_{p_{32}})$. Since rotation is defined as an addition of two angles, the result of the rotation is similar to compositions of Table 1. Specifically, rotations are defined in Table 2. Given a directional orientation relation R_o and a positional orientation relation R_p, each entry in Table 2 corresponding to row R_o and column R_p represent the result of the rotation of relation R_p with respect to relation R_o, yielding a set of positional orientation relations.

Relations	$F(Front)$	$R(Right)$	$B(Back)$	$L(Left)$
$S(Same)$	F, R, L	F, R, B	R, B, L	F, B, L
$R(Right)$	F, R, B	R, B, L	F, B, L	F, R, L
$O(Opposite)$	R, B, L	F, B, L	F, R, L	F, R, B
$L(Left)$	F, B, L	F, R, L	F, R, B	R, B, L

Table 2. Rotation Table for Positional Orientation with respect to Directional Orientations



(a) Composition Example (b) Positional Composition Example

Fig. 3. (a) Composition Example (b) Positional Composition Example

After performing the rotation of the second positional orientation relation, the two positional orientation relations can be composed. Table 3 represents the result of the composition of two positional orientation relations of Figure 2(a) (relations $Front$, $Right$, *Back* and *Left*, are denoted by "Fr","Ri","Ba", "Le" respectively, and All denotes the disjunction of all relations).

Relations	Fr	Ri	Ba	Le
Fr	Fr	Fr, Ri	All	Fr, Le
Ri	Fr, Ri	Ri	Ri, Ba	All
Ba	All	Ri, Ba	Ba	Ba, Le
Le	Fr, Le	All	Ba, Le	Le

Table 3. Composition Table for Positional Orientation Relations.

Composition Table can be interpreted as follows: if relation R_1 holds between $point2$ and $point1$ and relation R_2 holds between $point3$ and $point2$, then the entry of the Table 3 corresponding to line R_1 and column R_2 denotes the possible relation(s) holding between $point3$ and $point1$ (points represent the centroid of corresponding objects). For example, if $point2$ is $Front$ of $point1$ and $point3$ is $Left$ to $point2$ (after rotating object $o2$ to so as to point to the same direction as object $o1$) then $point3$ is $Front$ or $Left$ to $point1$. Entries in the above composition tables are determined using the following observation: composition of two relations corresponds to the addition of two vectors representing the relative placement of $point2$ to $point1$ and $point3$ to $point2$ (after performing the aforementioned rotation) forming angles θ_1 and θ_2 respectively with the reference axis. The resulting vector represents the relative placement of $point3$ to $point1$, i.e., the composition of two vectors, as illustrated in Figure 3(b). When adding the two vectors the resulting vector forms an angle θ with the reference axis such that $\theta_2 \leq \theta \leq \theta_1$. Angle θ defines the relation between $point1$ and $point3$. Using this observation it can be concluded for example that composing relations $Right$ and $Front$ yields the disjunction of these two relations as a result.

A series of compositions of relations may yield relations which are inconsistent with existing ones (e.g., the above example will yield a contradiction if *point3 back of of point1* has been also asserted into the ontology). Consistency checking is achieved by ensuring path consistency by applying formula:

$$\forall x, y, k \; R_s(x, y) \leftarrow R_i(x, y) \cap (R_j(x, k) \circ R_k(k, y))$$

representing intersection of compositions of relations with existing relations (symbol $\cap$ denotes intersection, symbol $\circ$ denotes composition and R_i, R_j, R_k, R_s denote directional relations). The formula is applied until a fixed point is reached (i.e., the application of the rules above does not yield new inferences) or until the empty set is reached, implying that the ontology is inconsistent. Implementing path consistency formula requires rules for both compositions and intersections of pairs of relations.

Compositions of relations R_1, R_2 yielding a unique relation R_3 as a result are expressed in SWRL using rules of the form:

$$R_1(x, y) \wedge R_2(y, z) \rightarrow R_3(x, z)$$

Note that for directional orientation relations of Figure 1(b), rules of the above form apply, but for positional orientation relations of Figure 2(a), rules are of the form:

$$R_1(x,y) \wedge R_{o_1}(x,y) \wedge R_2(y,z) \to R_3(x,z)$$

where R_{o_1} is a relation of the set presented in Figure 1(b).
The following is an example of such a composition rule:

$$Front(x,y) \wedge Front(y,z) \to Front(x,z)$$

Rules yielding a set of possible relations cannot be represented directly in SWRL, since disjunctions of atomic formulas are not permitted as a rule head. Instead, disjunctions of relations are represented using new relations, whose compositions must also be defined and asserted into the knowledge base. For example, the composition of relations $Front$ and *Right* yields the disjunction of two possible relations (*Front* and *Right*) as a result:

$$Front(x,y) \wedge Right(y,z) \rightarrow (Front \vee Right)(x,z)$$

If the relation $Front\_Right$ represents the disjunction of relations *Front* and *Right*, then the composition of $Front$ and $Right$ can be represented using SWRL as follows:

$$Front(x,y) \wedge Right(y,z) \rightarrow Front\_Right(x,z)$$

A set of rules defining the result of intersecting relations holding between two points must also be defined in order to implement path consistency. These rules are of the form:

$$R_1(x,y) \wedge R_2(x,y) \to R_3(x,y)$$

where R_3 can be the empty relation. For example, the intersection of relations $Left$ and $Right$ yields the empty relation, and an inconsistency is detected:

$$Left(x,y) \wedge Right(x,y) \to \perp$$

Intersection of relations $Right$ and $Right\_Back$ (representing the disjunction of $Right$ and *Back* yields relation $Right$ as a result:

$$Right(x,y) \wedge Right\_Back(x,y) \to Right(x,y)$$

Thus, path consistency is implemented by defining compositions and intersections of relations using SWRL rules and OWL axioms for inverse relations as presented in Section 3. Another important issue for implementing path consistency is the identification of the additional relations, such as the above mentioned $Right\_Back$ relation, that represent disjunctions. Specifically *minimal* sets of relations required for defining compositions and intersections of all relations that can be yielded when applying path consistency on the basic relations of Figures 1(b) and 2(a) are identified. The identification of the additional relations is required for the construction of the corresponding SWRL rules.

In this work, the *closure method* [2] of Table 4 is applied for computing the minimal relation sets containing the set of basic relations: starting with a set of relations, intersections and compositions of relations are applied iteratively until no new relations are yielded forming a set closed under composition, intersection and inverse. Since compositions and intersections are constant-time operations (i.e., a bounded number of table lookup operations at the corresponding composition tables is required) the running time of closure method is linear to the total number of relations of the identified set. This method is applied over both sets of relations.

```
Input: Set S of tractable relations
Table C of compositions
WHILE S size changes
    BEGIN
        Compute C:Set of compositions of relations in S
        $S=S \cup C$
        Compute I:set of intersections of relations in S
        $S= S \cup I$
    END
RETURN S
```

Table 4. Closure method

A reduction to required relations and rules can be achieved by observing that the disjunction of all basic relations when composed with other relations yields the same relation, while intersections yield the other relation. Specifically, given that All represents the disjunction of all basic relations and R_x is a relation in the supported set, then the following holds for every R_x:

$$All(x, y) \land R_x(x, y) \rightarrow R_x(x, y)$$

$$All(x, y) \land R_x(y, z) \rightarrow All(x, z)$$

$$R_x(x, y) \land All(y, z) \rightarrow All(x, z)$$

Since relation All always holds between two points, because it is the disjunction of all possible relations, all rules involving this relation, both compositions and intersections, do not add new relations into the ontology and they can be safely removed. Also, all rules yielding the relation All as a result of the composition of two supported relations R_{x1}, R_{x2}:

$$R_{x1}(x, y) \land R_{x2}(y, z) \rightarrow All(x, z)$$

can be removed as well. Thus, since intersections yield existing relations and the fact that the disjunction over all basic relations must hold between two objects, all rules involving the disjunction of all basic relations, and consequently all rules yielding this relation, can be safely removed from the knowledge base. After applying the closure method and optimizations the required number of relations for representation and reasoning (basic and disjunctive) is 23 (14 directional and 9 positional).

5 Evaluation

In the following the proposed representation and reasoning mechanism is evaluated both theoretically and experimentally.

5.1 Theoretical Evaluation

The required expressiveness of the proposed representation is within the limits of OWL 2 expressiveness. Reasoning is achieved by employing DL-safe rules expressed in SWRL that apply on named individuals in the ontology ABox, thus retaining decidability.

Specifically, any object can be related with every other object with two basic directional relations (one of each set presented in Figures 1(b) and 2(a)). Since relations of each set are mutually exclusive, between n objects, at most $2n(n-1)$ relations can be asserted. Furthermore, path consistency has $O(n^5)$ time worst case complexity (with n being the number of points). In the most general case where disjunctive relations are supported, in addition to the basic ones, any object can be related with every other object by at most k relations, where k is the size of the set of supported relations. Therefore, for n objects, using $O(k^2)$ rules, at most $O(kn^2)$ relations can be asserted into the knowledge base.

The $O(n^5)$ upper limit for path consistency running time referred to above is obtained as follows: At most $O(n^2)$ relations can be added in the knowledge base. At each such addition step, the reasoner selects 3 variables among n objects which corresponds to $O(n^3)$ possible different choices. Clearly, this upper bound is pessimistic, since the overall number of steps may be lower than $O(n^2)$, because an inconsistency detection may terminate the reasoning process early, or the asserted relations may yield a small number of inferences. Also, forward chaining rule execution engines employ several optimizations, thus the selection of appropriate variables usually involves fewer than $O(n^3)$ trials. Nevertheless, since the end user may use any reasoner supporting SWRL, a worst case selection of variables can be assumed in order to obtain an upper bound for complexity. Also, retaining control over the order of variable selection and application of rules yields an $O(n^3)$ upper bound for path consistency [3].

5.2 Experimental Evaluation

Measuring the efficiency of the proposed representation requires the spatial ontology of Section 3, containing instances, thus a data-set of 10K to 100K objects generated randomly was used for the experimental evaluation. Reasoning response times of the spatial orientation reasoning rules are measured as the average over 10 runs. HermiT 1.3.8 running as a library of a Java application was the reasoner used in the experiments. All experiments where run on a PC, with Intel Core CPU at 2.4 GHz, 6 GB RAM, and Windows 7.

Measurements illustrate that the proposed approach can efficiently represent thousands of objects and reason over them in a few seconds, without using specialized software besides a standard OWL reasoner such as HermiT.

6 Conclusions and future work

In this work, a representation framework for handling orientation spatial information in ontologies is introduced. The proposed framework handles both, egocentric directional and positional information using an inference procedure based on path consistency.

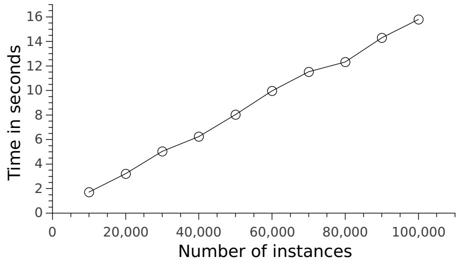


Fig. 4. Average reasoning time for orientation relations as a function of the number of objects

The proposed representation is fully compliant with existing Semantic Web standards and specifications, which increases its applicability. Being compatible with W3C specifications, the proposed framework can be used in conjunction with existing editors, reasoners and querying tools such as Protégé and HermiT, without requiring any additional specialized software. Therefore, information can be easily distributed, shared and modified. Directions of future work include the development of applications based on the proposed mechanism. Such applications could combine temporal and topological spatial representations with the proposed orientation representation and reasoning mechanism.

References

1. S. Batsakis and E.G.M. Petrakis. “SOWL: A Framework for Handling Spatio-Temporal Information in OWL 2.0”. *5th International Symposium on Rules: Research Based and Industry Focused (RuleML’ 2011)*, pages 242–249, 2011.
2. J. Renz, and B. Nebel. “Qualitative Spatial Reasoning using Constraint Calculi”. In Handbook of Spatial Logics, Springer, Netherlands , pp. 161-215, 2007.
3. M. Stocker, and E. Sirin “PelletSpatial: A Hybrid RCC-8 and RDF/OWL Reasoning and Query Engine” In: CEUR Workshop Proceedings, vol. 529-OWLED 2009, pp. 2-31, 2009.
4. G. Christodoulou, E.G.M. Petrakis, and S. Batsakis “Qualitative Spatial Reasoning using Topological and Directional Information in OWL” In Proc. of 24th International Conference on Tools with Artificial Intelligence (ICTAI 2012), November 7-9, 2012.
5. J. Renz, and D. Mitra “Qualitative Direction Calculi with Arbitrary Granularity” In PRICAI 2004: Trends in Artificial Intelligence: 8th Pacific Rim International Conference on Artificial Intelligence, Auckland, New Zealand, August 9-13, 2004.
6. S. Batsakis “Reasoning over 2D and 3D directional relations in OWL: a rule-based approach” In Theory, Practice, and Applications of Rules on the Web (RuleML2013), p.37-51, 2013.
7. R. Baruah, and S. M. Hazarika “Qualitative directions in egocentric spatial reference frame.” Intelligent Systems Design and Applications (ISDA), 2012 12th International Conference on. IEEE, 2012.
8. T. Mossakowski, and R. Moratz “Qualitative reasoning about relative direction of oriented points.” Artificial Intelligence 180 (2012): 34-45.
9. R. Shearer, B. Motik, and I. Horrocks. “HermiT: A Highly-Efficient OWL Reasoner.” OWLED. Vol. 432. 2008.