Published in: GIScience 2008, T.J. Cova et al. (Eds.)
LNCS 5266, pp. 234–248, 2008.

Reasoning on Spatial Relations between Entity Classes

Stephan Mäs

AGIS - Arbeitsgemeinschaft GIS, University of the Bundeswehr Munich,
Werner Heisenberg Weg 39, 85577 Neubiberg, Germany
{Stephan.Maes}@unibw.de

Abstract. The facilitation of interoperability requires a clear distinction if a relation refers to classes of individuals or to specific instances, in particular when it comes to the logical properties of the involved relations. Class relations are defined whenever the semantics of entire classes are described, independently of single instances. Typical examples are spatial semantic integrity constraints or ontologies of entity classes. The paper continues research on spatial class relations by deepening the analysis of the reasoning properties of class relations. The work is based on a set of 17 abstract class relations defined in [11]. The paper provides a complete composition table for the 17 abstract class relations and redefines the concept of conceptual neighbourhood for class relations. This approach can be used to find conflicts and redundancies in sets of semantic integrity constraints or other applications of spatial class relations.

Keywords: Class Level Relations, Spatial Relations, Reasoning, Compositions, Conceptual Neighbourhood, Constraint Networks

1 Introduction

The definition of class relations and their logical properties did not attract much attention of the scientific community so far. But as already argued by [1] the facilitation of interoperability requires a clear distinction if a relation refers to classes of individuals or to concrete instances, in particular when it comes to the logical properties of the involved relations. The difference becomes obvious through simple natural language statements, like for example: “house #12 is contained by parcel #1234”. This is a simple statement about two entities related by the spatial relation *contained by*. Since *contained by* is the inverse relation of *contains* the statement also implies “parcel #1234 contains house #12”. A statement about the corresponding entity classes is “buildings are contained by a parcel”. Applying the symmetry of the instance relation again it becomes: “a parcel contains buildings”. These statements can be mistaken, since they should be understood as “every building is contained by some parcel”, but NOT as “every parcel contains some building”. This example shows the influence of words like “all” or “some” on the semantic of a statement. They define cardinality restrictions on the applied relation. For a human reader it is very often possible to interpret the correct semantics, but a formal description of such a statement must explicitly contain cardinality information. The example also shows that a relation among the classes is not subject to the same logical properties as a relation between instances and the cardinality information must be considered for reasoning. This has also been pointed out by [1] and [11].

Class relations define a cardinality restriction for a certain relation between the individuals of classes. The restriction is always valid for entire classes or subsets of classes and not exclusive for single instances. A class relation always links the cardinality restriction to an instance relation. Typical applications of class relations are ontologies of classes and semantic integrity constraints. For geographical information class relations are of particular interest, because a semantic description of such data requires class relations which are based on spatial instance relations, like e.g. topological or metric relations. The interoperable exchange of data of different domains and application areas requires semantic descriptions of the data. Class relations are useful for the formalization of these descriptions. The logical properties of the class relations support an automatic processing, querying and comparing of such descriptions.

As shown in previous investigations [1] [11] it is useful to separately analyse the reasoning properties of the class relations and those of the instance relations. Therewith class relations can be flexibly used in combination with any kind of instance relation. In a previous paper a set of 17 abstract class relations has been defined, which is independent of concrete instance relations [11]. For these relations some basic reasoning concepts have been investigated. This paper continues these investigations particularly with regard to the composition of class relations and the conceptual neighbourhood of class relations. The following chapter will recapitulate the definition of basic cardinality properties and based on that the definition of the 17 abstract class relations. In the third chapter the composition of 17 class relations is investigated. The paper provides an overall analysis of all possible compositions, which has never been presented before. Furthermore the concept of conceptual neighbourhood, which has been so far only considered in combination with instance relations, is redefined for class relations. It is shown how these logics can be used to find conflicts in triples of class relations.

2 Definition of Class Level Relations

A class relation defines cardinality restrictions for a certain relation between all instances of the involved classes. In the following subchapters some basic cardinality properties are defined and used for the definition of class relations. It recapitulates the definitions made in [11]. For a more extensive elaboration refer to the original paper.

2.1 General Definitions and Requirements

For the definition of class relations the classes must conform to the following two preconditions. First, the involved classes must have at least one instance, i.e. empty classes are not feasible. As stated before class relations are linked to instance relations. Thus the second condition specifies that if a class relation is defined, there must at least one corresponding instance relation exist among the instances of the involved classes. The investigations in this paper are restricted to binary relations between entire entity classes. Relations between three or more classes or between subsets of classes (e.g. all blue houses as a subset of the class house) are not considered.

In the following definitions small letters 'x', 'y', 'z', … denote variables for instances / individuals. Every instance must be associated to an entity class. For entity classes capital letters 'A', 'B', 'C', … are used. 'Inst(x,A)' means individual x is an instance of class A. The function 'r(x,y)' means instance x has the relation r to instance y; x and y are said to participate in the relationship instance r. The meta-variable r can stand for any relation between instances (e.g. topological relations between areal features [2], which are used in the examples the paper). Instance relationships can be associated to a class relation R. For class relation definitions 'R(A,B)' denotes that R relates the classes A and B. The meta-variable R can stand for any class relationship. Every R is related to an instance relation r. If a class relation R(A,B) is defined, at least one r must exist between the instances of A and B.

2.2 Cardinality Properties of Class Level Relations

Cardinalities represent the number of elements of a set. Class relations refer to an instance relation and restrict the cardinality of this relation between the instances of the involved classes. The restrictions are defined through cardinality properties. In [11] the following cardinality properties of class relations have been used.

$$\mathrm{LT}(A,B,r) := \forall x(\mathrm{Inst}(x,A) \rightarrow \exists y(\mathrm{Inst}(y,B) \cap r(x,y))). \quad \text{(CP1)}$$

$$\mathrm{RT}(A,B,r) := \forall y(\mathrm{Inst}(y,B) \rightarrow \exists x(\mathrm{Inst}(x,A) \cap r(x,y))). \quad \text{(CP2)}$$

The cardinality properties (CP1) and (CP2) define a totality for the class A and B respectively. (CP1) holds if every instance of A has the relation r to some instance of B. In set theory such relations are called *left-total*.

(CP2) holds if for each instance of B there is some instance of A which stands in relation r to it. This means that every instance of B has the inverse relation of r to some instance of A. In this case the relation is *right-total*. The concepts of totality have also been used for the class relations defined in [1].

$$\begin{aligned}\mathrm{LD}(A,B,r) := \forall x,y,z[&\mathrm{Inst}(x,A) \cap \mathrm{Inst}(y,B) \cap \mathrm{Inst}(z,A) \cap \\ &r(x,y) \cap r(z,y) \rightarrow x = z] \cap \mathrm{Ex}(A,B,r).\end{aligned} \quad \text{(CP3)}$$

$$\begin{aligned}\mathrm{RD}(A,B,r) := \forall x,y,z[&\mathrm{Inst}(x,A) \cap \mathrm{Inst}(y,B) \cap \mathrm{Inst}(z,B) \cap \\ &r(x,y) \cap r(x,z) \rightarrow y = z] \cap \mathrm{Ex}(A,B,r).\end{aligned} \quad \text{(CP4)}$$

$$\mathrm{Ex}(A,B,r) := \exists x \exists y(\mathrm{Inst}(x,A) \cap \mathrm{Inst}(y,B) \cap r(x,y)). \quad \text{(CP5)}$$

Class relations which hold (CP3) are *left-definite* and specify that for no instance of B there is more than one instance of A which stands in relation r to it. This property restricts the number of r relations an instance of B can participate; the instances of A are not restricted. The last term ensures that at least one instance relation r does exist between the instances of A and B (CP5).

(CP4) specifies that no instance of A participates in a relationship r to more than one instance of B. When this cardinality property is defined in a class relation all instances of A are restricted while the instances of B are not affected. The corresponding class relations are *right-definite*.

Such properties of class relations are well established in data modelling, for example when total participation and cardinality ratio constraints are described using the Entity-Relationship notation. In such models a total participation is represented by a double line for the relation and cardinality ratio for example by a N:1 next to the relation signature (figure 1). In this example all buildings are restricted to be contained by exactly one parcel, while the parcels are allowed to contain an undefined number of buildings. *Contains* is the restricted instance relation. The number of different cardinality ratio constraints of such a notation is indefinite. This approach only considers a cardinality ratio of 0..1, which is representing the concept of unambiguousness.

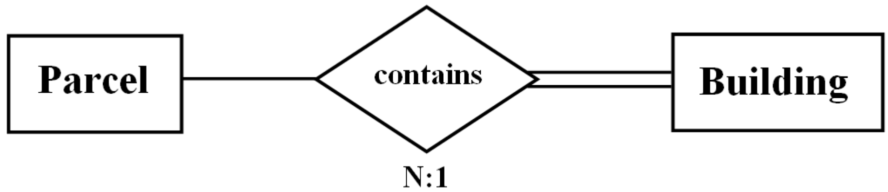

Fig. 1. Constraints in an Entity-Relationship Notation

2.3 Class Level Relations

The formal definition of class relations is based on the above defined cardinality properties *left-definite*, *right-definite*, *left-total* and *right-total*. These properties are independent of each other. This means that no property implies or precludes one of the other properties. If a class relation is only defined as *right-total* there is no information about its left totality and the unambiguousness available.

As the example in figure 1 illustrates, the properties can be combined for the definition of a class relation. The class relation in the example is based on the topological instance relation *contains* and the cardinality properties *left-definite* and *right-total*. The other two properties are not valid. The corresponding class relation $CONTAINS_{LD.RT}(Parcel, Building)$ is based on the class relation defined in (CR1).

$$R_{LD.RT}(A,B) \coloneqq LD(A,B,r) \cap RT(A,B,r) \cap \neg RD(A,B,r) \cap \neg LT(A,B,r). \qquad \text{(CR1)}$$

In the following I will refer such class relations, which are not linked to a particular instance relation, as **abstract class relations** (e.g. $R_{LD.RT}(A,B)$). In analogy to (CR1) the four cardinality properties can be used to define a set of 15 abstract class relations, where for each relation at least one of the four properties holds and the others are excluded, respectively. An investigation of all possible combinations leads to:

– four abstract class relations where one property is valid and the corresponding other three are excluded,
– six abstract class relations where two properties are valid and the other two are excluded,
– four abstract class relations which combine three of the four defined cardinality properties respectively and exclude the corresponding fourth,
– and one abstract class relations where all four properties are valid.

Additionally to these 15 abstract class relations two special cases are considered in [11]. To achieve a jointly exhaustive set of relations one abstract class relation is defined for the situation that none of the four properties is valid but some instances of A stand in relation r to some instances of B (CR2).

Second, a further abstract class relation is defined for the case that all instances of A have a relationship instance of R to all instances of B (CR3). This is a strict case of a *left-total* and *right-total* relation. For class relations it is frequently occurring, for example when no instances of two classes are allowed to intersect: $\text{DISJOINT}_{\text{LT.RT-all}}$(Streets, Lakes).

$$R_{some}(A,B) := Ex(A,B,r) \cap \neg LD(A,B,r) \cap \neg RD(A,B,r) \cap \neg LT(A,B,r) \cap \neg RT(A,B,r). \quad \text{(CR2)}$$

$$R_{LT.RT-all}(A,B) := \forall x \forall y (Inst(x,A) \cap Inst(y,B) \rightarrow r(x,y)). \quad \text{(CR3)}$$

This set of 17 abstract class relations enables the definition of class relations based on any binary instance relation. For further details on the definition of the abstract class relations it is referred to [11]. Figure 2 shows an example for each of the abstract class relations.

Table 1. Restrictions of the number of instances of the abstract class relations

Abstract class relation	Minimal required instances A	Minimal required instances B	Comparison number of A / number of B	Abstract class relation	Minimal required instances A	Minimal required instances B	Comparison number of A / number of B
1. $R_{LD.RD}$	2	2	-	10. $R_{RD.RT}$	3	1	A > B + 1
2. R_{LD}	2	3	-	11. R_{LT}	2	3	-
3. R_{RD}	3	2	-	12. R_{RT}	3	2	-
4. R_{some}	3	3	-	13. $R_{LD.RD.LT.RT}$	2	2	A = B
5. $R_{LD.RD.LT}$	1	2	A < B	14. $R_{LD.LT.RT}$	2	3	A < B
6. $R_{LD.RD.RT}$	2	1	A > B	15. $R_{RD.LT.RT}$	3	2	A > B
7. $R_{LD.RT}$	2	2	-	16. $R_{LT.RT}$	2	2	-
8. $R_{RD.LT}$	2	2	-	17. $R_{LT.RT-all}$	1	1	-
9. $R_{LD.LT}$	1	3	A + 1 < B				

Please note that class relations are not depending on fixed numbers of instances and the constellations represented in figure 2 are just exemplarily. Some abstract class relations require a minimum number of instances of A and/or B and a certain ratio between the instances of both classes. These restrictions are represented in table 1.

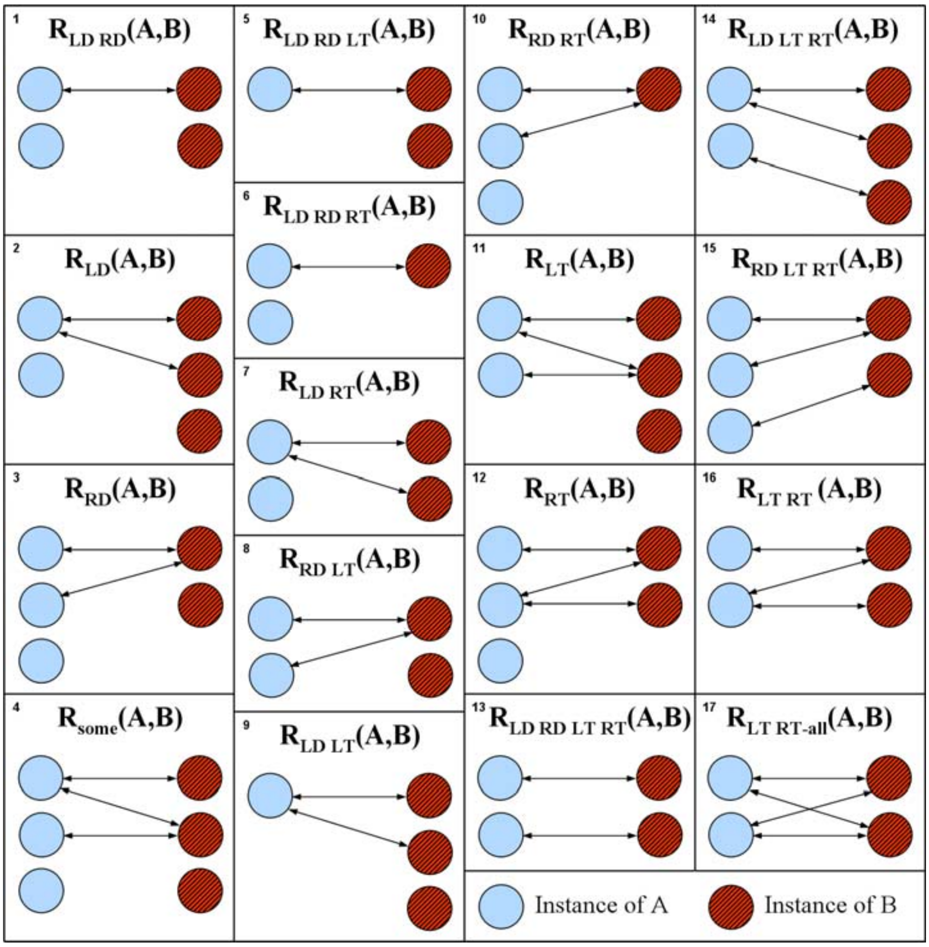

Fig. 2. Examples of the 17 abstract class relations

The set of the 17 abstract class relations is a qualitative representation of the constellation of instance relations between two classes. For every instance relation there is only one class relation valid for each pair of classes. Nevertheless it is possible to define more than one class relation between two classes, even when the applied instance relations are part of the same jointly exhaustive and pair wise disjoint (JEPD) set of instance relations. Based on the definitions of the cardinality properties and the abstract class relations [11] it can be proven that for a JEPD set of instance relations the corresponding class relations are also JEPD (hence each class relation definition excludes the cardinality properties which are not valid, see (CR1))

3 Reasoning on Class Relations

The reasoning methods presented in this chapter can be used to find conflicts and redundancies in sets of class relations. As demonstrated in [1] and [11] the logical properties of class relations derive from the logical properties of the corresponding instance and abstract class relations. It has been shown that it is appropriate to analyse the reasoning properties of the abstract class and instance level relations independently of each other. For the abstract class relations it is reasonable to refer the logical properties to their cardinality definitions. The following sections investigate the symmetry, composition and conceptual neighbourhood of the 17 abstract class relations.

3.1 Symmetry of Class Relations

Logical properties class relations, like symmetry and transitivity, have been researched in [1]. As pointed out by [11] in particular the symmetry is of interest, since this property has to be proven to ensure the arc consistency of a class relation network. It has been shown that every class relation has an inverse relation, if the corresponding instance relation has an inverse relation or is symmetric. Most spatial relations fulfil this requirement. The symmetry properties of the class relations can be derived from the symmetry of the applied instance relations and the cardinality definitions of the abstract class relations ((CP1)-(CP5), (CR2) and (CR3)). The inverse of a class relation is also based on the inverse of the applied instance relation. If an abstract class relation is *left-total* / *left-definite* the inverse relation is *right-total* / *right-definite* and vice versa. R_{some} and $R_{LT.RT\text{-}all}$ are symmetric. The following two examples demonstrate the derivation of inverse class relations. Here the class relations are based on the symmetric instance relation *disjoint* and the inverse relations *contains* and *inside*:

$$(\text{DISJOINT}_{\text{RD.LT}}(A,B))^{i} = \text{DISJOINT}_{\text{LD.RT}}(B,A).$$
$$(\text{CONTAINS}_{\text{LD.RD.LT}}(A,B))^{i} = \text{INSIDE}_{\text{LD.RD.RT}}(B,A).$$

3.2 Composition of Class Relations

The composition of binary relations enables the derivation of implicit knowledge about a triple of entities. If two binary relations are known the corresponding third one can potentially be inferred or some of the possible relations can be excluded. This knowledge can also be used to find conflicts in case all of the three relations are known. The compositions rules of a set of relations are usually represented in a composition table like it has been done for the topological relations between areal entities in [4][8] and for directional/orientation relations in [6][9]. Many other sets of binary spatial relations also allow for such derivations.

The composition of class relations is hardly researched yet, but as shown in [11] it is also possible. This paper proposes a two level reasoning formalism, which separates the compositions of the abstract class relations from those of the instance relations (see figure 3). Therewith the composition of the abstract class relations can be defined independently of a concrete set of instance relations.

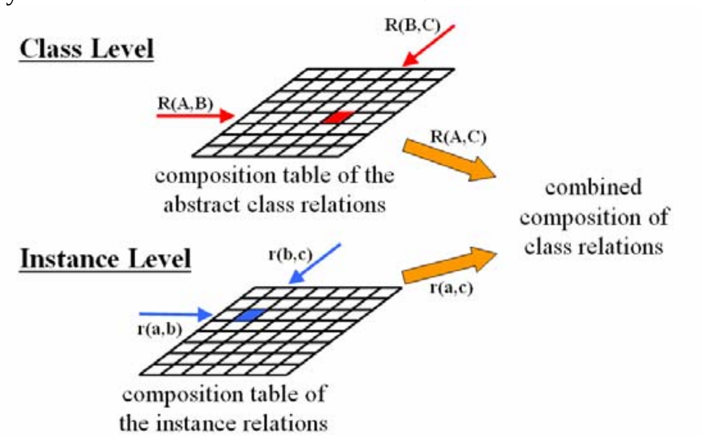

Fig. 3. Two levels composition of class relations

The following example demonstrates how the compositions of the abstract class relations are inferred. With the class relations $R1_{LT.RT\text{-}all}(A,B)$ and $R2_{LD.RD.LT.RT}(B,C)$ given, the relation between the classes A and C shall be derived. Such a situation is schematically represented in figure 4. All instances of A have the same instance relation r1 to all instances of B and all instances of B are related by r2 to one instance of C. To infer the composition of the abstract class relations every possible triple of A, B and C instances has to be separately analysed. Whenever the relation r1 between the instance of A and the instance of B and the relation r2 between the instance of B and the instance of C is given, the relation r3 (or a disjunction of possible relations) between the A and C instances can be inferred. The combination of the inferences of all possible triples of instances leads to the abstract class relation between A and C. If no triple of instances with r1 and r2 relations exists, then no inference for r3 is possible. Therewith the composition of the class relations leads to a universal disjunction $\mathcal{U}$ of all possible class relations. For the example shown in figure 4, each instance of A is related to every instance of C via some instance of B. Therewith it is obvious that all instances of A must have the same instance relation to all instances of C. Thus the composition of the abstract class relations must be:

$$\mathrm{R1}_{\mathrm{LT.RT-all}}(\mathrm{A,B});\mathrm{R2}_{\mathrm{LD.RD.LT.RT}}(\mathrm{B,C}) \Rightarrow \mathrm{R3}_{\mathrm{LT.RT-all}}(\mathrm{A,C}).$$

For the composition of the class relations this result must be combined with the composition of the instance relations, for example (taken from the composition table in [8]):

$$\mathrm{meet}(a,b);\mathrm{covers}(b,c) \Rightarrow \mathrm{disjoint}(a,c) \cup \mathrm{meet}(a,c).$$

The combination of the compositions of the two levels results in:

$$\mathrm{MEET}_{\mathrm{LT.RT-all}}(\mathrm{A,B});\mathrm{COVERS}_{\mathrm{LD.RD.LT.RT}}(\mathrm{B,C}) \Rightarrow \left[\mathrm{DISJOINT} \cup \mathrm{MEET}\right]_{\mathrm{LT.RT-all}}(\mathrm{A,C}).$$

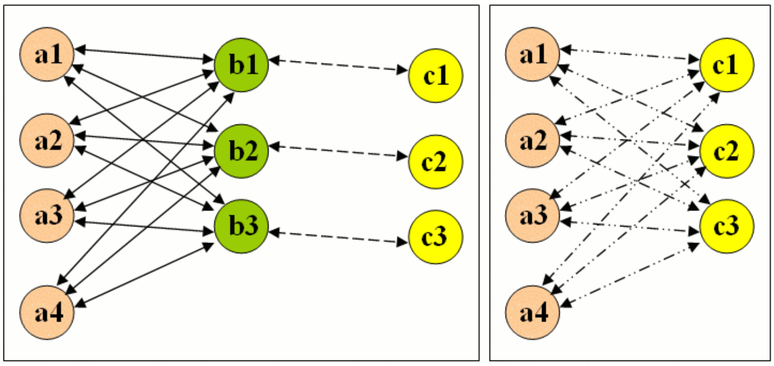

Fig. 4. Possible scene defined by the class relations $R1_{LT.RT\text{-}all}(A,B)$ and $R2_{LD.RD.LT.RT}(B,C)$ and their composition $R3_{LT.RT\text{-}all}(A,C)$

For the example shown in figure 4 the derived composition is independent of the number of instances. This means that the composition of the given abstract class relations will always lead to the same result. The 17 abstract class relations have 289 possible compositions. Many of them have differing results, depending on the number of instances of the three classes and the relative arrangement of the instance relations. This must be considered when the compositions are calculated.

The influence of the relative arrangement of the instance relations on the composition is illustrated in figure 5. The two boxes show possible constellations of the $R1_{LT.RT}$; $R2_{LT.RT}$ composition. They only differ in an instance relation between the classes B and C: in the left box b1 and c2 are related whereas in the right box the instances b2 and c1 are related. The abstract class relations and the total amount of instance relations are the same. Nevertheless this difference will lead to different compositions. For the left constellation the relation between the instances a2 and c1 can not be inferred and the composition is $R3_{LT.RT}$ (relation #16). The right constellation allows for a deduction of all four instance relations between A and C and thus the composition is $R_{LT.RT\text{-}all}$ (relation #17). The disjunction of all possible results leads to the composition of the abstract class relations:

$$R1_{LT.RT}(A,B); R2_{LT.RT}(B,C) \Rightarrow R3_{LT.RT}(A,C) \cap R3_{LT.RT-all}(A,C).$$

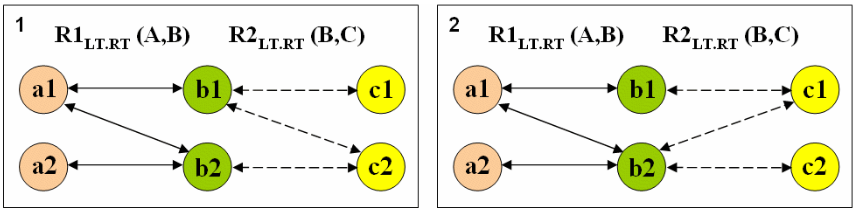

Fig. 5. Example of how the relative arrangement of the instance relations influences the composition of the abstract class relations.

In previous publications the composition of class relations has only been exemplarily investigated; an overall analysis of all possible compositions has never been presented. The calculation of this composition table is complex and costly. For the defined abstract class relations it requires an analysis of all possible arrangements of instance relations for up to 6 instances for each of the three classes. If both classes of one relation have 6 instances than there are about 68,7 billion arrangements possible. Each of these has to be separately analysed with all possible arrangements of the second relation. An analysis of classes with 7 or more instances does not lead to additional results in the composition. Making use of some heuristics this calculation can be further optimised. The overall composition table is shown in figure 6.

1. Relation \ 2. Relation		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	LD.RD	u	u	u	u	1	u	u	1,3	1,2	u	1,2, 3,4	u	1	1,2	1,3, 6,10	1,2,3, 4,6,7, 10,12	6, 7, 10,12
2	LD	u	u	u	u	2	u	u	1,2, 3,4	2	u	1,2, 3,4	u	2	2	1,2,3,4, 6,7,10, 12	1,2,3, 4,6,7, 10,12	6, 7, 10,12
3	RD	u	u	u	u	3	u	u	3	3,4	u	3,4	u	3	3,4	3,10	3,4, 10,12	10,12
4	some	u	u	u	u	4	u	u	3,4	4	u	3,4	u	4	4	3,4, 10,12	3,4, 10,12	10,12
5	LD.RD.LT	u	u	u	u	5	u	u	5,8	5,9	u	5,8, 9,11	u	5	5,9	5,8, 13,15	5,8,9, 11,13,14, 15,16,17	17
6	LD.RD.RT	1	2	3	4	1	6	7	3	2	10	4	12	6	7	10	12	6, 7, 10,12
7	LD.RT	1,2	2	1,2, 3,4	2,4	2	6,7	7	1,2, 3,4	2	6,7, 10,12	2,4	7,12	7	7	6,7, 10,12	7,12	6, 7, 10,12
8	RD.LT	u	u	u	u	8	u	u	8	8,11	u	8,11	u	8	8,11	8,15	8,11,15, 16,17	17
9	LD.LT	u	u	u	u	9	u	u	5,8, 9,11	9	u	5,8, 9,11	u	9	9	5,8,9,11, 13,14,15 16,17	5,8,9,11, 13,14,15, 16,17	17
10	RD.RT	1,3	2,4	3	4	3	6,10	7,12	3	4	10	4	12	10	12	10	12	10,12
11	LT	u	u	u	u	11	u	u	8,11	11	u	8,11	u	11	11	8,11, 15,16, 17	8,11, 15,16, 17	17
12	RT	1,2 3,4	2,4	1,2, 3,4	2,4	4	6,7, 10,12	7,12	3,4	4	6,7, 10,12	4	7,12	12	12	10,12	12	10,12
12	RT	1,2 3,4	2,4	1,2, 3,4	2,4	4	6,7, 10,12	7,12	3,4	4	6,7, 10,12	4	7,12	12	12	10,12	12	10,12
13	LD.RD.LT.RT	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
14	LD.LT.RT	1,2, 5,9	2,9	1,2,3, 4,5,8, 9,11	2,4, 9,11	9	6,7, 13,14	7,14	5,8, 9,11	9	6,7,10, 12,13,14, 15,16,17	9,11	7,12, 14,16, 17	14	14	13,14, 15,16, 17	14,16, 17	17
15	RD.LT.RT	1,3	2,4	3	4	8	6,10	7,12	8	11	10	11	12	15	16	15	16	17
16	LT.RT	1,2,3, 4,5,8, 9,11	2,4, 9,11	1,2,3, 4,5,8, 9,11	2,4, 9,11	11	6,7,10, 12,13,14, 15,16,17	7,12, 14,16, 17	8,11	11	6,7,10, 12,13,14, 15,16,17	11	7,12, 14,16, 17	16	16	15,16, 17	16,17	17
17	LT.RT-all	5,8, 9,11	9,11	5,8, 9,11	9,11	5,8, 9,11	17	17	5,8, 9,11	9,11	17	9,11	17	17	17	17	17	17

Fig. 6. Composition table of the 17 abstract class relations

Some of the compositions can be summarized by general rules, which deduce the composition directly from the cardinality properties. This allows a more convenient use of the composition. Some obvious rules are:

- If the first abstract class relation is not *right-total* and the second relation is not *left-total* the composition is always a universal disjunction $\mathcal{U}$.
- If the first relation is $R_{LD.RD.RT}$ (relation #6) and the second is not *left-total* the composition is equal to the second abstract class relation.
- If the first relation is $R_{LD.RD.RT}$ (relation #6) and the second is *left-total* the composition has the same cardinality properties as the second relation, but it is not *left-total.* For $R_{LT.RT\text{-}all}$ (relation #17) this can be relation #6, #7, #10 or #12.
- If the first relation is not *right-total* and the second is $R_{LD.RD.LT}$ (relation #5) the composition is equal to the first abstract class relation.
- If the first relation is *right-total* and the second is $R_{LD.RD.LT}$ (relation #5) the composition has the same cardinality properties as the first relation, but it is not *right-total.* For $R_{LT.RT\text{-}all}$ (relation #17) this can be relation #5, #8, #9 or #11.
- If one of the relations is $R_{LD.RD.LT.RT}$ (relation #13) the composition is always equal to the corresponding other abstract class relation. Because of this property $R_{LD.RD.LT.RT}$ can represent the identity relation of classes if it is combined with a identity instance relation, e.g. $EQUAL_{LD.RD.LT.RT}(A,A)$.
- If the first relation is $R_{LT.RT\text{-}all}$ (relation #17) and the second is *right-total* the composition is always $R_{LT.RT\text{-}all}$.
- If the first relation is *left-total* and the second is $R_{LT.RT\text{-}all}$ (relation #17) the composition is always $R_{LT.RT\text{-}all}$.

The compositions which are defined by these rules are highlighted in grey in figure 6. A set of rules which completely represents the composition table is a subject of further research.

For the abstract class relations and their composition table the properties of a relation algebra [12] have been computationally checked. Some properties of the presented composition of abstract class relations are:

- The inverse of an inverse relation is equal to the original relation: $(R^i)^i = R$.
- All compositions with the identity relation (relation #13) are idempotent: $R;R_{LD.RD.LT.RT} \rightarrow R$ and $R_{LD.RD.LT.RT};R \rightarrow R$.
- The inverse of a composition is equal to the composition of the inverses of the two relations in reverse order: $(R1 ; R2)^i = R2^i ; R1^i$.
- The associative property $(R1;R2);R3 = R1;(R2;R3)$ and the semiassociative property $R; (\mathcal{U};\mathcal{U}) = (R;\mathcal{U});\mathcal{U}$ [10] are not valid. Therewith the composition of the abstract class relations is nonassociative.

3.3 Conceptual Neighbourhood of Class Relations

The conceptual neighbourhood represents continuous transformations between relations through linking relations that are connected by an atomic change. [6] defines two relations in a representation as conceptual neighbours, “if an operation in the represented domain can result in a direct transition from one relation to the other.” Examples of conceptual neighbourhood networks of instance relations can be found for temporal interval relations in [7], for topological relations between regions in [3] and between regions and lines in [5].

The conceptual neighbourhood of class relations has not yet been researched. In this approach two class relations are considered as conceptually neighboured if they are linked to the same instance relation and only differ by a single instance relation between two entities. The number of instances of the class is considered as fixed. In figure 7 the conceptual neighbourhood of R_{some} and $R_{LT.RT}$ is exemplarily illustrated. All arrows symbolize one instance relation of the same kind *r*. The addition of a further instance relation, represented by the dashed arrow in the right box, leads to a transition of the abstract class relation from R_{some} to $R_{LT.RT}$.

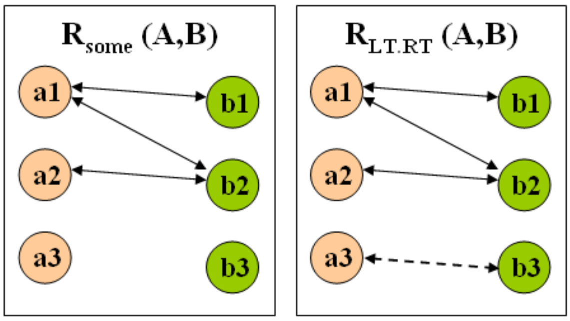

Fig. 7. Conceptual neighbourhood between R_{some} and $R_{LT.RT}$

The computation of all conceptual neighbourhoods between class relations requires an analysis of all possible arrangements of instance relations for up to 4 instances for both classes. A higher number of instances does not lead to additional results. For the 17 class relations are all together 45 neighbourhoods existing. Since the neighbourhood is defined through adding or removal of a single instance relation all neighbourhoods are directed. Table 2 represents the neighbourhoods which result from an addition by a “+” and those which result from a removal by a “-”. If a class relation has changed though an addition / removal of an instance relation, it is not possible to get the same class relation again by further adding / removing of instance relations. The adding of instance relations will ultimately lead to $R_{LT.RT\text{-}all}$ (relation #17). A removal will lead to $R_{LD.RD}$ (relation #1)[1]. The numbering of the abstract class relations has been chosen such that for all class relations the relations which result from an addition have a higher number and all inverse relations are successive. Because of this order all removal neighbourhoods appear at the left bottom and all addition neighbourhoods at the right top in table 2.

[1] For this relation both involved classes must have at least two entities, see table 1.

Table 2. Conceptual neighbourhood of the class relations; +/- represents neighbourhood through addition/removal of an instance relation

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1 LD.RD		+	+	+	+	+	+	+					+				
2 LD	-			+			+		+		+			+			
3 RD	-			+				+		+		+			+		
4 some	-	-	-								+	+				+	
5 LD.RD.LT	-								+		+			+			+
6 LD.RD.RT	-									+		+			+		+
7 LD.RT	-	-										+				+	
8 RD.LT	-		-								+					+	
9 LD.LT		-			-						+			+			+
10 RD.RT			-			-						+			+		+
11 LT		-		-	-			-	-							+	
12 RT			-	-		-	-			-						+	
13 LD.RD.LT.RT	-															+	
14 LD.LT.RT		-			-				-							+	
15 RD.LT.RT			-			-				-						+	
16 LT.RT				-			-	-			-	-	-	-	-		+
17 LT.RT-all					-	-			-	-						-	

The following example shall illustrate the practical use of the conceptual neighbourhood of class relations. Three class relations are defined for the classes A, B and C: $MEET_{some}(A,B)$, $CONTAINS_{LD.RD.LT.RT}(B,C)$ and $DISJOINT_{LT}(A,C)$. These relations shall be analysed for conflicts through comparing the composition of the class relations A to B and B to C with the given third relation between A and C. The compositions of the corresponding instance and abstract class relations are:

$$\text{meet}(a,b);\text{contains}(b,c) \Rightarrow \text{disjoint}(a,c).$$
$$R1_{some}(A,B);R2_{LD.RD.LT.RT}(B,C) \Rightarrow R3_{some}(A,C).$$

Thus the combination of the compositions of the two levels results in:

$$MEET_{some}(A,B);CONTAINS_{LD.RD.LT.RT}(B,C) \Rightarrow DISJOINT_{some}(A,C).$$

This result seems to be in conflict to the given third relation $DISJOINT_{LT}(A,C)$. Figure 8 exemplarily illustrates this situation. The first box shows the given class relations and the second the inferred relation between A and C. In comparison with this the third box shows that the given relation $DISJOINT_{LT}(A,C)$ possibly differs from $DISJOINT_{some}(A,C)$ by only one *disjoint* instance relation (in this case a3 to c2). Therewith $DISJOINT_{some}(A,C)$ and $DISJOINT_{LT}(A,C)$ are conceptual neighbours. In figure 8 the three *disjoint* instance relations of $DISJOINT_{LT}(A,C)$ are implied by the class relations A to B and B to C. About further relations between the instances of A and C the composition does not allow for any conclusion. It can also not be excluded that further pairs of A and C instances are *disjoint*. Hence the composition of $MEET_{some}(A,B)$ and $CONTAINS_{LD.RD.LT.RT}(B,C)$ does not conflict $DISJOINT_{LT}(A,C)$ and the triple of class relation is consistent. Beside $DISJOINT_{LT}(A,C)$ also the class relations $DISJOINT_{RT}(A,C)$ and $DISJOINT_{LT.RT}(A,C)$ as direct conceptual neighbours of $DISJOINT_{some}()$ and $DISJOINT_{LT.RT\text{-}all}(A,C)$ as conceptual neighbour of $DISJOINT_{LT.RT}()$ would not conflict.

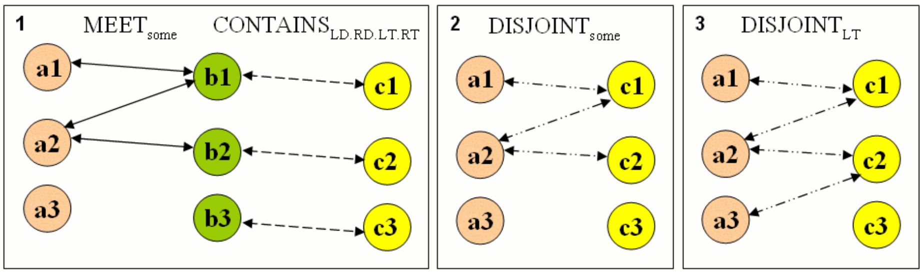

Fig. 8. Use of the conceptual neighbourhood for the composition of class relations

In general, a class relation R3 is not in conflict with a composition R1 ; R2 → R3* if R3* and R3 are based on the same instance relation r3 and the addition of further r3 instance relations to R3* can lead to a transition to class relation R3. For this the result of the composition R3* and R3 don't need to be direct conceptual neighbours. There can also be further class relation transitions between the two class relations. Nevertheless the conceptual neighbourhood points out which R3 class relations are valid, since it shows which transitions are possible for a certain class relation R3*.

Thus the check of conflicts in a triple of class relations consists of two steps: first the comparison of the composition of two relations with the given third. If they are equal the triple of relations is conform to the introduced composition of class relations and there is no obvious conflict. If these two relations are not equal the second step checks their conceptual neighbourhood as described above. If the given third relation is not a corresponding conceptual neighbour of the composition the triple of class relations is conflicting.

4 Conclusion and Open Issues

The scientific investigation of class relations is currently still in the early stages. This work continues research into spatial class relation by deepening the analysis of the reasoning properties of the class relations. It is based on a set of 17 abstract class relations defined in [11]. The paper focuses on the composition and conceptual neighbourhood of class relations. The definitions and reasoning rules of the class relations are described independently of a specific set of instance relations. The introduced two levels composition of class relations allows for a separate analysis of instance relations and abstract class relations. Therewith the overall reasoning formalism can be used with any spatial or non-spatial set of instance relations. The only requirements imposed on the instance relations are that they are part of a JEPD set of relations and have defined inverse relations and compositions.

With the described logics it is possible to find conflicts and redundancies in networks of class relations. This can for example be applied to prove consistency of sets of spatial semantic integrity constraints or spatial relations between classes in an ontology.

This approach is restricted to binary relations between entire entity classes. Relations between three or more classes or between subsets of classes are not considered. Further more, only total participation and a cardinality ratio of 0..1 are included as cardinality properties of the class relations. Nevertheless this framework provides a basis, which can be extended for other possibly more complex types of class relations. For an extension by further cardinality ratio constraints (e.g. 0..2) it has to be considered, that this will increase the calculation cost of the compositions exponentially.

Further work can also deal with the direct derivation of the reasoning properties of the class relations from their cardinality properties. This will deepen the understanding of the logics and support possible extensions by additional cardinality properties. The two levels composition of class relations separates the compositions of the abstract class relations from those of the instance relations (figure 3). However, some combinations of instance and abstract class relations lead to conflicts which can not be found this way. For example the combination of $EQUAL_{LD.RT}(A,B)$ and $R_{LD.RD.LT.RT}(B,C)$ (see figure 2: relation #7; relation #13) is not possible. This is due to the specific properties of the *equal* identity instance relation and the cardinality properties of the two abstract class relations. A general description of such conflicts is unsolved.

As pointed out in table 1 some abstract class relations require a minimum number of instances in A and/or B and a certain ratio between the instances of both classes. For many entity classes the number of existing individuals is unknown or variable. For these classes the restriction of the number of individuals is irrelevant. However, for classes with a small and well defined number of instances (e.g. earth surface or continents) the designer of a data model or an ontology is in many cases aware of these numbers. The knowledge about these numbers and their correlation to the class relations can be included in reasoning about class relations. The restrictions on the number of instances also lead to restrictions of the composition and the conceptual neighbourhood.

References

1. Donnelly, M., Bittner, T.: Spatial Relations Between Classes of Individuals. In: Cohn, A.G., Mark, D.M. (eds.) COSIT 2005. LNCS, vol. 3693, pp. 182–199. Springer, Heidelberg (2005)
2. Egenhofer, M., Herring, J.: Categorizing Binary Topological Relationships Between Regions, Lines, and Points in Geographic Databases. Technical Report, Department of Surveying Engineering, University of Maine, Orono, ME (1991)
3. Egenhofer, M., Al-Taha, K.K.: Reasoning about Gradual Changes of Topological Relationships. In: Frank, A.U., Formentini, U., Campari, I. (eds.) Theories and Methods of Spatio-Temporal Reasoning in Geographic Space. LNCS, vol. 639, pp. 196 – 219. Springer, Berlin (1992)
4. Egenhofer, M.: Deriving the Composition of Binary Topological Relations. Journal of Visual Languages and Computing 5(2), 133–149 (1994)

5. Egenhofer, M., Mark, D.: Modeling Conceptual Neighborhoods of Topological LineRegion Relations. International Journal of Geographical Information Systems, 9, pp. 555 - 565. (1995)
6. Freksa, C.: Using Orientation Information for Qualitative Spatial Reasoning. In: Frank, A.U., Formentini, U., Campari, I. (eds.) Theories and Methods of Spatio-Temporal Reasoning in Geographic Space. LNCS, vol. 639, pp. 162–178. Springer, Berlin (1992)
7. Freksa, C.: Temporal reasoning based on semi-intervals. In: Artificial Intelligence, 54, 199 – 227, 1992.
8. Grigni, M., Papadias, D., Papadimitriou, C.: Topological inference. In: Proceedings of the International Joint Conference of Artificial Intelligence (IJCAI), pp. 901–906 (1995)
9. Hernandez, D.: Qualitative Representation of Spatial Knowledge. In: Hernández, D. (ed.) Qualitative Representation of Spatial Knowledge. LNCS, vol. 804, Springer, Heidelberg (1994)
10. Maddux, R.: Some varieties containing relation algebras. In: Transactions of the American Mathematical Society 272, no. 2, pp.501-526 (1982)
11. Mäs, S.: Reasoning on Spatial Semantic Integrity Constraints. In: Winter, S., Duckham, M., Kulik, L., Kuipers, B. (eds.) COSIT 2007. LNCS, vol. 4736, pp. 285–302. Springer, Heidelberg (2007)
12. Tarski, A.: On the Calculus of Relations. In: The Journal of Symbolic Logic, vol. 6, no. 3, pp. 73-89 (1941)