Semantic Integrity Constraints for Spatial Databases

Loreto Bravo and M. Andrea Rodríguez

Universidad de Concepción, Chile
{lbravo,andrea}@udec.cl

Abstract. This paper introduces a formalization of a set of spatial semantic integrity constraints on an extended-relational database model. The formalization extends traditional notions of functional and inclusion dependencies by adding interaction with spatial attributes. This enables to specify implicit and explicit topological conditions between geometries and impose constraints on thematic attributes that depend on the geometries. We study the consistency problem for this set of integrity constraints, which rises issues about topological consistency and realizability of spatial constraints. We show that the consistency problem is not tractable and provide some conditions under which it is.

1 Introduction

Spatial databases systems enhance traditional databases with models and query languages that support spatially referenced data and provide spatial indexing and efficient algorithms for spatial query processing [8]. The use of spatially referenced data is an essential component of such diverse applications as Geographic Information Systems (GIS), Computer-Aided Design (CAD), multimedia information systems, data warehousing, mobile computing, location-based services, and NASA's Earth Observing System (EOS).

Fundamental to spatial databases are the *spatial data models or geometric models* that represent objects in a space. Examples of such models are the spaghetti [18], topological [9, 18], and polynomial models [14]. Most of the current spatial database management systems (SDBMSs) implement extensions to relational databases, so called object-relational or extended-relational databases, by defining spatial data types that represent the geometry of objects in the Euclidean space (e.g. Postgres+Posgis, Oracle Spatial SQL, and MySql Spatial).

In the context of database consistency, spatial databases offer new challenges due, particularly, to the complex nature of spatial attributes, the derivation of implicit relations from spatial attributes, and the combination of spatial and non-spatial attributes (called thematic attributes). SDBMSs enforce integrity constraints used for preventing structural inconsistency of spatial attributes (i.e. domain constraints in the relational context). For example, they enforce that regions must be represented by closed and simple polylines [2]. The other important type of spatial integrity constraints, which are not implemented in SDBMs, are the spatial semantic integrity constraints (also known as topo-semantic integrity constraints) that relate geometries with semantic conditions. For example, a semantic constraint can enforce that a city's administrative region must be contained within its corresponding city limits [17]. These constraints are less formalized and are highly dependent on the application.

In this work we define a constraint language to specify spatial semantic constraints. This language extends functional and inclusion dependency constraints to combine spatial and thematic attributes.

Example 1. Consider a spatial database that stores information of land parcels and buildings using the schema $\mathcal{R} = \{Landparcel(id,owner,area,g), Building(id, use,g)\}$. In predicate *Landparcel*, the thematic attributes are *id*, *owner*, and *area* (in m^2), and the only spatial attribute is g that stores the geometry of each land parcel. Similarly for *Building*, which has *id* and *use* as thematic attributes and g as spatial attribute. Figure 1 shows an instance of this schema. In it, dark rectangles represent buildings and white rectangles represent land parcels.

Landparcel	*id*	*owner*	*area*	*g*
	$L45$	J. Dupont	840	g_1
	$L96$	J. Smith	800	g_2
	$L33$	A. White	508	g_3

Building	*id*	*use*	*g*
	$B23$	residential	g_4
	$B87$	industry	g_5

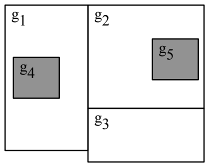

Fig. 1. A spatial database instance

In this database, we need traditional constraints to enforce that attribute *id* is unique in both predicates. But, there are other relations that cannot be captured by traditional functional dependencies or inclusion dependencies. For example, we need spatial constraints to ensure that land parcels do not overlap, that each building is inside a land parcel, and that the value in attribute *area* corresponds to the area of the spatial attribute g in predicate *landparcel*. □

Some studies have addressed the specification of spatial integrity constraints [1, 10, 12]; however, they treat spatial data in isolation from other integrity constraints, this is, without considering the interaction with the thematic attributes. Other studies have analyzed topological [6, 12] and realizability [7] problems. The topological problem analyzes the consistency of topological relations as expressed by inference tables. The realizability problem, on the other hand, analyzes whether or not there is a set of geometries that realizes a set of topological relations. These problems differ from the database consistency problem, or simply consistency problem, in which the goal is to determine if there is a non-empty database instance that satisfies a set of integrity constraints. In the consistency problem the number of geometries in a database instance that must satisfy a set of constraints is unknown. For topological and realizability problems, instead, there exists a pre-defined number of geometries that must satisfy a set of constraints expressed by the possible topological relations between them. To the best of our knowledge, the consistency problem of spatial integrity constraints has not been studied so far.

This is why in this paper we formalize a set of spatial semantic integrity constraints that: (i) generalizes traditional functional and inclusion dependencies, (ii) enables to specify topological relations between spatial objects, (iii) enables to relate both spatial and thematic attributes, and (iv) are expressive enough to capture constraints found in practice. Furthermore, we reason about these constraints and determine whether or not the set of integrity constraints is consistent, i.e., if there is a non-empty database instance that satisfies the set of constraints. We show that this problem is not tractable and provide some conditions under which it is.

The organization of this paper is as follows. In Section 2 we present the data model used to represent spatial databases. Then, in Section 3 we present the set of spatial semantic constraints for which we will study the consistency problem in Section 4. Finally, in Section 5 we briefly discuss related work and give conclusions.

2 A Data Model for Spatial Databases

Current models of spatial data are typically seen as extensions of the relational data model (object-relational models), with the definition of abstract data types to specify spatial or geometric attributes. We now introduce a general spatia-relational data model that includes spatia-relational predicates (that can be purely relational) and also spatial ICs. It uses some of the definitions introduced in [15] and is independent of the geometric data model underlying the representation of spatial data types.

A spatia-relational database schema is of the form $\Sigma = (\mathcal{U}, \mathcal{A}, \mathcal{S}, \mathcal{R}, \mathcal{T}, \mathcal{O})$, where: (a) $\mathcal{U}$ is the possibly infinite database domain of atomic thematic values that includes $\mathbb{R}$. (b) $\mathcal{A} = \{A_1, \ldots, A_n\}$ where each A_i is a thematic attribute which takes values in $\mathbb{R}$. (c) $\mathcal{S} = \{S_1, \ldots, \mathcal{S}_n\}$ where each S_i takes admissible values in $\mathcal{P}(\mathbb{R}^2)$, the power set of $\mathbb{R}^2$. (d) $\mathcal{R}$ is a finite set of spatia-relational predicates each of them with a finite and ordered set of attributes belonging to $\mathcal{A}$ or $\mathcal{S}$. We sometimes use $R(A_1, \ldots, A_n)$ to represent the predicate with its set of attributes. (e) $\mathcal{T}$ is a fixed set of binary topological predicates. (f) $\mathcal{O}$ is a fixed set of geometric operators that take spatial and thematic arguments and return a geometry or a value in $\mathcal{U}$.

A database instance D of a spatia-relational schema Σ is a finite collection of ground atoms (or *spatial database tuples*) of the form $R(c_1, \ldots, c_i, \ldots, c_n)$, where (a) $R(A_1, \ldots, A_i, \ldots, A_n) \in \mathcal{R}$, (b) if $A_i \in \mathcal{A}$, then $c_i \in \mathcal{U}$ (c) if $A_i \in \mathcal{S}$, then $c_i \in Ad \subseteq \mathcal{P}(\mathbb{R}^2)$ where Ad is the class of closed regions in the topological space formed by point sets of the Euclidean space. These regions represent real objects that have geographic extent, this is, they are not empty.

The elements of $\mathcal{T}$ are binary topological relations with a fixed semantics. There are eight basic binary relations: Overlaps (O), Equal (EQ), CoveredBy (CB), Inside (IS), Covers (CV), Includes (IC), Touches (TO), and Disjoint (D) [16, 5][1]. In addition to the basic topological relations, SQL languages consider some *derived relations* that can be logically defined in terms of the other basic predicates: Intersects (IT), Within (W), and Contains (C) [13]. A lattice showing how these relations are grouped to derive new relations is shown in Figure 2. The eight basic relations are shown in boxes containing prototypical cases and the relations included in current SQL are represented with thick borders. For example, Contains is obtained by grouping Equal, CoveredBy and Includes, this is, A Contains B if either A is Equal to, is CoveredBy or Includes B. Observe that relation IDisjoint (IDC) (internally disjoint) was added even though it is neither basic nor used in SQL. Nonetheless, it is added because it is useful to represent situations found in practice. In what follows we assume that $\mathcal{T}$ contains all the relations shown in Figure 2.

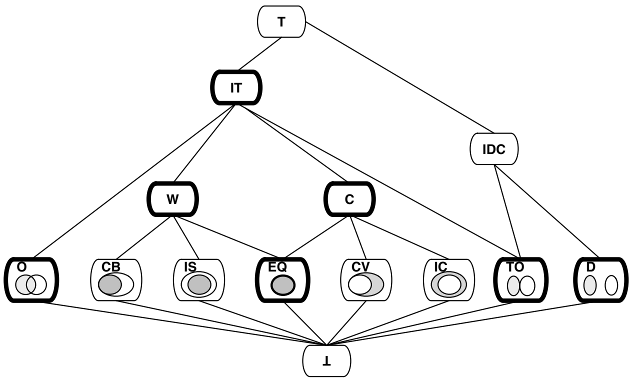

Fig. 2. Subsumption lattice of relations O (Overlaps), CB (CoveredBy), IS (Inside), EQ (Equal), CV (Covers), IC (Includes), TO (Touches), D (Disjoint), IT (Intersects), IDC (IDisjoint), W (Within), and C (Contains).

Given a database instance, additional spatial information is usually computed from the explicit geometric data by means of the spatial operators in $\mathcal{O}$. Some relevant operators are: *Area*, *Union*, *Intersection* and *Difference* (cf. [13] for the complete set of spatial predicates defined within the Open GIS Consortium).

3 Spatial Semantic Integrity Constraints

In this section we present a set of spatial semantic integrity constraints that generalizes classic integrity constraints and that are capable of dealing with spatial data.

To simplify the presentation we will use the following notation: (i) variables x, y, z, x_1, x', ... represent values in $\mathcal{U}$ or geometries, (ii) variables g, g_1, g', ...

[1] The names of relations given here are in agreement with the names used in current SQL languages [13], but differ slightly from the ones in the research literature.

represent geometries, and (iii) variables u, v, w, u_1, u', ... represent values in $\mathcal{U}$. (iv) symbols $\bar{x}$, $\bar{u}$, $\bar{x}_1$, ... denote a possibly empty sequence of distinct variables or, with a slight abuse of notation, a set of variables.

Definition 1. A *spatial semantic integrity constraint* (SSIC) is one of the following:
Functional Dependency (FD):
$\forall \bar{u}\bar{y}_1\bar{y}_2 v_1 v_2 (P(\bar{u}, \bar{y}_1, v_1) \wedge P(\bar{u}, \bar{y}_2, v_2) \rightarrow v_1 = v_2)$
where $\bar{y}_1 \cap \bar{y}_2 = \emptyset$.
Inclusion Dependency (IND):
$\forall \bar{u}\bar{x}(P(\bar{u}, \bar{x}) \rightarrow \exists \bar{y} R(\bar{u}, \bar{y}))$
Topological Dependency (TD):
$\forall \bar{u}\bar{y}_1\bar{y}_2\bar{g}_1\bar{g}_2(P(\bar{u}, \bar{y}_1, g_1) \wedge R(\bar{u}, \bar{y}_2, g_2) \bigwedge_{i=1}^{n} v_i \neq w_i \rightarrow \mathcal{T}(g_1, g_2))$
where $\bar{y}_1 \cap \bar{y}_2 = \emptyset$ and for every i, variable $v_i \in \bar{y}_1$ and variable $w_i \in \bar{y}_2$.
Spatial Referential Dependency (RD):
$\forall \bar{u}\bar{x}g_1(P(\bar{u}, \bar{x}, g_1) \rightarrow \exists \bar{y} g_2 R(\bar{u}, \bar{y}, g_2) \wedge \mathcal{T}(g_1, g_2))$
Comparison Constraint (CC):
$\forall \bar{x}(P(\bar{x}) \rightarrow t_1 \odot t_2)$
where $\odot \in \{>, =, \neq\}$ and terms t_1 and t_2 are constructed from variables in $\bar{x}$ and operators in $\mathcal{O}$. □

FDs and INDs correspond to the traditional constraints since there are no joins nor references between geometrical attributes. Indeed, spatial attributes are not used in keys nor in referenced attributes since that would have a high computational cost associated in terms of space and time. Also, from a practical point of view, geometries by themselves do not identify spatial objects, so they need some kind of thematic key. TDs enforce topological relations between spatial attributes and RDs enforce inclusion dependencies where the spatial attributes of the referenced table should have some particular spatial properties. Finally, comparison constraints can be used to compare values within a tuple in P. Note that none of the constraints considers joins on geometric attributes.

Since all the topological relations in $\mathcal{T}$ have their converse (or inverse) relation in $\mathcal{T}$, there is no need to have an RD with $\mathcal{T}(g_2, g_1)$ in replacement of $\mathcal{T}(g_2, g_1)$. Indeed the constraint $\forall x g_1(P(x, g_1) \rightarrow \exists g_2 R(x, g_2) \wedge \mathsf{CoveredBy}(g_2, g_1)$ can be replaced by $\forall x g_1(P(x, g_1) \rightarrow \exists g_2 R(x, g_2) \wedge \mathsf{Covers}(g_1, g_2)$.

Example 2. (example 1 cont.) The following SSICs can be defined:

$$\bar{\forall}\,(\text{Landparcel}(u, v, w, g) \wedge \text{Landparcel}(u, v', w', g') \rightarrow v = v') \quad (1)$$
$$\bar{\forall}\,(\text{Landparcel}(u, v, w, g) \wedge \text{Landparcel}(u, v', w', g') \rightarrow w = w') \quad (2)$$
$$\bar{\forall}\,(\text{Landparcel}(u, v, w, g) \wedge \text{Landparcel}(u, v', w', g') \rightarrow \mathsf{Equal}(g, g')) \quad (3)$$
$$\bar{\forall}\,(\text{Building}(u, v, g) \wedge \text{Building}(u, v', g') \rightarrow v = v') \quad (4)$$
$$\bar{\forall}\,(\text{Building}(u, v, g) \wedge \text{Building}(u, v', g') \rightarrow \mathsf{Equal}(g = g')) \quad (5)$$
$$\bar{\forall}\,(\text{Landparcel}(u, v, w, g) \wedge \text{Landparcel}(u', v', w', g') \wedge u \neq u' \rightarrow \mathsf{IDisjoint}(g, g')) \quad (6)$$
$$\bar{\forall}\,(\text{Building}(u, v, g) \wedge \text{Building}(u', v', g') \wedge u \neq u' \rightarrow \mathsf{IDisjoint}(g, g')) \quad (7)$$

$$\bar{\forall}\,(\text{Building}(u,v,g) \rightarrow \exists xyzg'\,(\text{Landparcel}(x,y,z,g') \wedge \mathsf{Within}(g,g'))) \quad (8)$$

$$\bar{\forall}\,(\text{Landparcel}(u,v,w,g) \rightarrow \mathit{Area}(g) = w) \quad (9)$$

Functional dependencies (1-2) together with topological dependency (3) ensure that *id* is a key of table Landparcel. In the same way, constraints (4-5) enforce that *id* is a key of Building. The topological dependencies (6) and (7) ensure that land parcels and buildings, respectively, are either adjacent or disjoint. Finally, the comparison constraint (9) forces attribute *area* to contain the area of the spatial attribute g. □

4 Consistency Problem

Before checking if a database satisfies a set of spatial semantic constraints, we need to make sure that the constraints themselves are not in conflict. Thus, we first need to check if the integrity constraints are consistent.

The *consistency problem* for a database schema Σ and a set of ICs IC, consists on determining whether there exists a non-empty database instance D over Σ that satisfies IC. This is, checking if there exists a nontrivial database that satisfies the constraints.

This consistency problem is related but different from the *topological* and *realizability* problems [6, 7]. A set of topological relations defined over a set of objects is said to be topologically consistent if there are no contradictions among the relations. For example three objects A, B and C and the relations $\mathsf{Covers}(A,B)$, $\mathsf{Covers}(B,C)$, $\mathsf{Disjoint}(A,C)$ are topologically inconsistent since the first two relations imply that A and C cannot be disjoint. There are some cases in which even if the topological relations are not inconsistent, the geometries may not be realizable due to reasons of planarity, this is, there are no geometries in $\mathbb{R}^2$ that satisfy those relations. The problem of determining if those geometries exist is the *realizability problem*.

Those problems differ from our consistency problem for two reasons. First, since we do not know in which tables we have tuples nor how many tuples there are, it is unknown the number of objects there are in the spatial database. Second, we do not know the topological relations that will be imposed since that will depend on the values not only in the spatial attributes but in the thematic attributes. Even though the problems differ from the consistency problem, some settings of the topological and realizability problem can be reduced to the consistency problem.

The following example illustrates the case when the topological relations imposed by spatial integrity constraints contradict topological consistency.

Example 3. (example 2 cont.) Consider the set of integrity constraints imposed in example 2. The database administrator could have, by mistake, written constraints (6) and (7) as follows:

$$\bar{\forall}\,(\text{Landparcel}(u,v,w,g) \wedge \text{Landparcel}(u',v',w',g') \rightarrow \mathsf{IDisjoint}(g,g'))$$

$$\bar{\forall}\,(\text{Building}(u,v,g) \wedge \text{Building}(u',v',g') \rightarrow \mathsf{IDisjoint}(g,g'))$$

The first constraint will also be checked when $u = u'$ and therefore it requires that a land parcel should be disjoint of or adjacent to itself. This, of course, is not possible. The second constraint enforces the same for *Building*. Since none of the predicates can contain tuples, the set of ICs is inconsistent. □

The following example shows a situation in which, even though the topological relations are consistent, it is not possible to find geometries in $\mathbb{R}^2$ that satisfy them, this is, the geometries are not realizable.

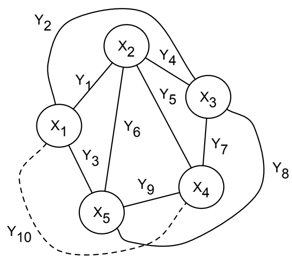

Fig. 3. The non-planar graph K_5

Example 4. A graph is said to be non-planar if it is not possible to draw it in the plane without edge intersection. A well known example of non-planar graph is K_5, the complete graph with five nodes (see Figure 3). In this graph, even though all the topological relations are satisfied, it is not realizable in the plane. We can map the topological relations on K_5 as a set of spatial integrity constraints IC_5 defined over a schema $\mathcal{S}_5$ in such a way that the set of constraints is consistent iff K_5 can be drawn in the plane. Let $\mathcal{S}_5$ contain five and ten spatia-relational predicates of the form $X_i(id, g_i)$ and $Y_j(id, g_j)$ respectively, which represent the nodes and edges of K_5, respectively, and where id is a thematic attribute, and g_i and g_j are spatial attributes. Now, let IC_5 contain:

1. For all $Z \in \{X_1, \ldots, X_5, Y_1, \ldots, Y_{10}\}$
   $\forall u g_1 g_2 (Z(u, g_1) \wedge Z(u, g_2) \rightarrow \mathsf{Equal}(g_1, g_2))$
2. For all $X_i \in \{X_1, \ldots, X_5\}$ and $Y_j \in \{Y_1, \ldots, Y_{10}\}$, where X_i and Y_j are incident in K_5:
   $\forall u g_i\, (\mathrm{X_i}(u, g_i) \rightarrow \exists g_j\, \mathrm{Y_j}(u, g_j))$
   $\forall u g_j\, (\mathrm{Y_j}(u, g_j)) \rightarrow \exists g_i\, \mathrm{X_i}(u, g_i))$
   $\forall u g_i g_j\, (\mathrm{X_i}(u, g_i) \wedge \mathrm{Y_j}(u, g_j) \rightarrow \mathsf{Overlaps}(g_i, g_j))$
3. For all $X_i \in \{X_1, \ldots, X_5\}$ and $Y_j \in \{Y_1, \ldots, Y_{10}\}$, where X_i and Y_j are not incident in K_5
   $\forall u g_i g_j\, (\mathrm{X_i}(u, g_i) \wedge \mathrm{Y_j}(u, g_j) \rightarrow \mathsf{Disjoint}(g_i, g_j))$
4. For all $X_i, X_j \in \{X_1, \ldots, X_5\}$ with $i \neq j$:
   $\forall u g_i v g_j\, (\mathrm{X_i}(u, g_i) \wedge \mathrm{X_j}(v, g_j) \rightarrow \mathsf{Disjoint}(g_i, g_j))$
5. For all $Y_i, Y_j \in \{Y_1, \ldots, Y_{10}\}$ with $i \neq j$:
   $\forall u g_i g_j\, (\mathrm{Y_i}(u, g_i) \wedge \mathrm{Y_j}(u, g_j) \rightarrow \mathsf{Disjoint}(g_i, g_j))$

The INDs enforce that if there is a tuple in any predicate, there will be at least one tuple in all the rest. Since a database is consistent if there is at least one tuple in the database, this constraints will ensure that there is at least one tuple in every predicates. The TDs enforce the topological relations between the different nodes and edges. It is easy to see that IC_5 is consistent iff K_5 is planar. Since K_5 is not planar, there is no non-trivial database that satisfies IC_5. This situation is interesting since the topological relations enforced by the TDs are not contradictory, but it is not possible to find tuples with objects in two dimensions that satisfy them. □

The constraints we have defined are able to express most of the constraints commonly needed in spatial databases. However, this expressiveness comes with a cost: the consistency problem for SSICs is not tractable.

Proposition 1. The consistency problem for SSIC is NP-*hard*. □

Proof. Hardness can be proven by reduction of the *explicit low level conjunctive realizability problem* [7] which is NP-*complete*. This problem consists in determining if there exists a planar model that satisfies a conjunctive topological expression like $\psi = (A \textit{ intersects } B) \wedge (B \textit{ disjoint } C) \wedge (A \textit{ disjoint } C)$. More formally it consists on determining if for a set of objects O there exists a planar model for the topological expression $\psi = \bigwedge_{i=1}^{n}(A_i \circ_i B_i)$ where (1) each $\circ_i \in \{\textit{intersects,disjoint}\}$[2], (2) if $(A_i \circ_i B_i)$ and $(A_i \circ_j B_i)$ belong to ψ, $\circ_i = \circ_j$, and (3) for every $A, B \in O$ such that $A \neq B$ there exists $\circ \in \{\textit{intersects,disjoint}\}$ such that $(A \circ B) \in \psi$. The problem is said to be "explicit" since for every pair of objects there always exists a relationship which is unique.

We need to construct a set of constraints IC_ψ defined over a schema Σ_ψ such that IC_ψ is consistent iff there exists a planar model of ψ for objects in O. Let $\Sigma_\psi = (\mathbb{R}, \mathcal{A}, \mathcal{S}, \mathcal{R}, \mathcal{T}, \mathcal{O})$ where $\mathcal{A} = \{id\}$, $\mathcal{S} = \{g\}$ and $\mathcal{R} = \{P_A(id, g) | A \in O\}$, $\mathcal{T} = \{\mathsf{Intersects}, \mathsf{Disjoint}\}$ and $\mathcal{O} = \emptyset$. Let IC_ψ contain:

1. For every $P_A \in \mathcal{R}$, a topological dependency:
   $\forall u g_1 g_2 (P_A(u, g_1) \wedge P_A(u, g_2) \rightarrow \mathsf{Equal}(g_1, g_2))$
2. For every $A, B \in \mathcal{O}$ where $A \neq B$, the spatial referential dependency:
   $\forall u g_1 (P_A(u, g_1) \rightarrow \exists g_2 (P_B(u, g_2) \wedge \mathcal{T}(g_1, g_2)))$
   where $\mathcal{T} = \begin{cases} \mathsf{Intersects} & \text{if } (A \textit{ intersects } B) \in \psi \text{ or } (B \textit{ intersects } A) \in \psi \\ \mathsf{Disjoint} & \text{if } (A \textit{ disjoint } B) \in \psi \text{ or } (B \textit{ disjoint } A) \in \psi \end{cases}$

Intuitively, each relation represents one of the objects in ψ, the TDs enforce that for each relation and value u in the first attribute there is a unique geometry, and RDs enforce that all the objects sharing the same value in attribute id satisfy the topological constraints given in ψ.

First let us show that if IC_ψ is consistent, then there is a planar model for ψ. If IC_ψ is consistent, there exists a database D defined over Σ_ψ that satisfies the constraints and that has at least one tuple in one of the relations. Then, from constraints in (2), it follows that every relation will have at least one tuple. Furthermore, for every predicate P_A there exists a constant c and a geometry g_A such that $(c, g_A) \in P_A$. Since this geometries will also satisfy the topological

[2] The topological relation called *intersects* in this paper is referred as *overlaps* in [7]

constraints, if we assign to each object A_i geometry g_{A_i}, we have found a planar model for ψ.

Now, let us prove that if there is planar model of ψ, IC_ψ is consistent. For every object $A \in O$, let g_A be the geometry of A in the planar model of ψ. Now, choose a constant $c \in \mathcal{U}$ and let D be such that $(c, g_A) \in P_A$ for every $A \in \mathcal{O}$. Clearly D satisfies IC_ψ and IC_ψ is consistent. □

It is an open problem to determine if the realizability problem used in the reduction is in NP since in some cases the objects need to be of exponential size.

A natural question is if we can find a set of constraints which are less expressive than SSIC, but that is still useful and tractable. We now explore the use of constraints that do not consider geometric operators $\mathcal{O}$ nor built-ins $\mathcal{B}$.

Definition 2. A *basic spatial integrity constraints* (BSIC) is one of the following:
Functional Dependency (FD):
$\forall \bar{u}\bar{y}_1\bar{y}_2 v_1 v_2 (P(\bar{u}, \bar{y}_1, v_1) \wedge P(\bar{u}, \bar{y}_2, v_2) \rightarrow v_1 = v_2)$
where $\bar{y}_1 \cap \bar{y}_2 = \emptyset$.
Inclusion Dependency (IND):
$\forall \bar{u}\bar{x}(P(\bar{u}, \bar{x}) \rightarrow \exists \bar{y} R(\bar{u}, \bar{y}))$
Topological Dependency ($\mathbf{TD_0}$):
$\forall \bar{u}\bar{y}_1\bar{y}_2\bar{g}_1\bar{g}_2(P(\bar{u}, \bar{y}_1, g_1) \wedge R(\bar{u}, \bar{y}_2, g_2) \rightarrow \mathcal{T}(g_1, g_2))$
where $\bar{y}_1 \cap \bar{y}_2 = \emptyset$.
Spatial Referential Dependency ($\mathbf{RD_0}$):
$\forall \bar{u}\bar{x}g_1(P(\bar{u}, \bar{x}, g_1) \rightarrow \exists \bar{y} g_2 R(\bar{u}, \bar{y}, g_2) \wedge \mathcal{T}(g_1, g_2))$ □

Even though these constraints are much simpler, the consistency problem is still NP-hard. This follows directly from the proof in the general case since only TD_0s and RD_0s are used.

Corollary 1. *The consistency problem for BSIC constraints is* NP*-hard.* □

However, these constraints have better properties in some cases. For example consistency of a set of FDs and TD_0s can be checked in polynomial time.

Proposition 2. *Consistency of a set of FDs and TD_0s can be checked in polynomial time.* □

Proof. First, observe that if a set IC of FDs and TD_0s is consistent, then there will exist a database with only one tuple in one predicate that satisfies the constraints. Thus, a possible algorithm consists on checking in each predicate P if a tuple with fresh constants in each attribute satisfies the FDs and TD_0s defined over it. If this is true in any of the predicates, then the set IC is consistent. Otherwise, it is not. □

A more general an interesting result is that if there are no cycles through INDs and RD_0s, consistency can be checked in polynomial time.

Definition 3. Given a schema $\mathcal{S}$ and a set of BSIC IC, let the *dependency graph* $\mathcal{G}$ of IC be a digraph where the nodes corresponds to predicates in $\mathcal{S}$ with an edge from predicate P to R if there is an IND $\forall \bar{u}\bar{x}(P(\bar{u}, \bar{x}) \rightarrow \exists \bar{y} R(\bar{u}, \bar{y}))$ or a $\mathrm{RD}_0 \forall \bar{u}\bar{x}g_1(P(\bar{u}, \bar{x}, g_1) \rightarrow \exists \bar{y}g_2 R(\bar{u}, \bar{y}, g_2) \wedge \mathcal{T}(g_1, g_2))$ in IC. A set of BSICs is said to be *acyclic* if its dependency graph is acyclic. □

Proposition 3. Consistency of an acyclic set of BSICs can be checked in polynomial time. □

Proof. If a set IC of acyclic BSICs is consistent, then there will exist a database with only one tuple in a leaf predicate that satisfies the constraints. Thus, IC is consistent iff one of the leafs satisfies its FDs and the TD_0. Since consistency of FDs and the TD_0 can be done in polynomial time, consistency of IC can be checked in polynomial time. □

5 Related Work and Conclusion

In the spatial domain, some related studies address the specification of topological constraints (domain constraints) [1] and spatial semantic constraints [12, 10]. There have been also studies concerning models for checking topological consistency at multiple representations and for data integration [4, 19, 6, 11]. In all these studies, however, the consistency problem has not been addressed, and spatial integrity constraints have been treated without considering the interaction with thematic attributes in a database model. The specification of constraints involving topological relations and alphanumeric attributes was studied at the conceptual level in [3]. This study, however, does not address the association of the constraints at the conceptual level with already existing semantic constraints at the logical level.

Unlike previous works, we have presented a formalization of integrity constraints that combine thematic with spatial attributes in a database model and that extends classical notions of functional and inclusion dependency. We have analyzed the consistency problem and found that it is not tractable in general. However, we have identified an interesting case for which the problem can be solved in polynomial time.

We plan to extend our results by analyzing other interesting cases in which the problem is tractable and to develop heuristics for those cases that it is not. In this direction, we would like to study the complexity of the consistency problem if we remove the requirement of realizability and only require topological consistency. This can be easily achieved by assuming that the spatial attributes take values in $\mathbb{R}^3$ instead of $\mathbb{R}^2$. Since the topological consistency problem is tractable in most settings, this can be a good approximation of the consistency problem.

Acknowledgements. This work is funded by Bicentenario Program PSD 57. Andrea Rodríguez is also funded by Fondecyt 1080138, and Loreto Bravo by Fondecyt 11080260, Conicyt - Chile.

References

1. K. Borges, A. Laender, and C. Davis. Spatial Integrity Constraints in Object Oriented Geographic Data Modeling. In *ACM-GIS*, pages 1–6, 1999.
2. S. Cockcroft. A Taxonomy of Spatial Integrity Constraints. *GeoInformatica*, 1(4):327–343, 1997.
3. M. Duboisset, F. Pinet, M. Kang, and M. Schneider. Precise Modeling and Verification of Topological Integrity Constraints in Spatial Databases: From an Expressive Power Study to Code Generation Principles. In *ER*, Springer LNCS 3716, pages 465–482, 2005.
4. M. Egenhofer, E. Clementine, and P. Di Felice. Evaluating Inconsistency among Multiple Representations. In *Spatial Data Handling*, pages 901–920, 1995.
5. M. Egenhofer and R. Franzosa. Point Set Topological Relations. *IJGIS*, 5:161–174, 1991.
6. M. Egenhofer and J. Sharma. Assessing the Consistency of Complete and Incomplete Topological Information. *Geographical Systems*, 1:47–68, 1993.
7. M. Grigni, D. Papadias, and C.H. Papadimitriou. Topological Inference. In *IJCAI*, pages 901–907, 1995.
8. R. Güting. An Introduction to Spatial Database Systems. *VLDB Journal*, 3:357–399, 1994.
9. R. Hartmut Güting. GraphDB: Modeling and Querying Graphs in Databases. In *VLDB*, pages 297–308, 1994.
10. T. Hadzilacos and N. Tryfona. A Model for Expressing Topological Integrity Constraints in Geographic Databases. In *Spatio-Temporal Reasoning*, Springer LNCS 639, pages 252–268, 1992.
11. B. Kuijpers, J. Paredaens, and J. Van den Bussche. On Topological Elementary Equivalence of Spatial Databases. In *ICDT*, Springer LNCS 1186, pages 432–446, 1997.
12. S. Mäs. Reasoning on Spatial Semantic Integrity Constraints. In *COSIT*, Springer LNCS 4736, pages 285–302, 2007.
13. OpenGis. Opengis Simple Features Specification for SQL. Technical report, Open GIS Consortium, 1999.
14. J. Paredaens, J. Van den Bussche, and D. Van Gucht. Towards a Theory of Spatial Database Queries. In *PODS*, pages 279–288. ACM Press, 1994.
15. J. Paredaens and B. Kuijpers. Data Models and Query Languages for Spatial Databases. *Data and Knowledge Engineering*, 25(1-2):29–53, 1998.
16. D. Randell, Z. Cui, and A. Cohn. A Spatial Logic based on Regions and Connection. In *KR*, pages 165–176, 1992.
17. S. Servigne, T. Ubeda, A. Puricelli, and R. Laurini. A Methodology for Spatial Consistency Improvement of Geographic Databases. *GeoInformatica*, 4(1):7–34, 2000.
18. D. Thompson and R. Laurini. *Fundamentals of Spatial Information Systems*. Number 37 in APIC. Academic Press, 1992.
19. N. Tryfona and M. Egenhofer. Consistency among Parts and Aggregates: A Computational Model. *Transactions on GIS*, 1(1):189–206, 1997.