

On capturing first-order topological properties of planar spatial databases

(extended abstract)

Bart Kuijpers^{*1} and Jan Van den Bussche²

¹ University of Antwerp (UIA), Dept. Math. & Computer Sci.,
Universiteitsplein 1, B-2610 Antwerp, Belgium
Email: kuijpers@uia.ua.ac.be

² Limburgs Universitair Centrum, Dept. WNI,
B-3590 Diepenbeek, Belgium
Email: vdbuss@luc.ac.be

Abstract. Spatial databases are modeled as closed semi-algebraic subsets of the real plane. First-order logic over the reals (expanded with a symbol to address the database) provides a natural language for expressing properties of such databases. Motivated by applications in geographical information systems, this paper investigates the question of which *topological* properties can be thus expressed. We introduce a novel, two-tiered logic for expressing topological properties, called \mathcal{CL} , which is subsumed by first-order logic over the reals. We put forward the question whether the two logics are actually equivalent (when restricting attention to topological properties). We answer this question affirmatively on the class of “region databases.” We also prove a general result which further illustrates the power of the logic \mathcal{CL} .

1 Introduction and summary

A simple yet powerful way of modeling spatial data is using *semi-algebraic sets*. A subset A of n -dimensional Euclidean space \mathbf{R}^n is called semi-algebraic if it can be defined by a Boolean system of polynomial inequalities. First-order logic over the reals, denoted here by $\text{FO}[\mathbf{R}]$, then becomes a spatial query language, fitting in the (by now rather well known) framework of constraint query languages introduced by Kanellakis, Kuper and Revesz [12]. The goal of this paper is to understand the power of this formalism in expressing topological queries.¹

We will work with *planar* spatial databases, whose content are described by semi-algebraic sets S in the plane \mathbf{R}^2 . An example of a first-order query in this context is “is the database bounded?”, which can be expressed in $\text{FO}[\mathbf{R}]$ as

^{*} Post-doctoral research fellow of the Fund for Scientific Research of Flanders (FWO-Vlaanderen).

¹ The work we will present is similar in spirit to work done in topological model theory [9, 11, 17], though the technical focus is quite different.

$(\exists b > 0)\forall x\forall y(S(x, y) \rightarrow (-b < x < b \wedge -b < y < b))$.² We will consider only sets that are closed in the ordinary topology on \mathbf{R}^2 . This assumption is of great help from a technical point of view, and is harmless from a practical point of view.

Topological properties. A property of spatial databases is called *topological* if it is invariant under topological transformations of the plane. More precisely, whenever the property holds for some A , it must also hold for any other A' that is the image of A under a homeomorphism of the plane.³ For example, the above-mentioned property “the database is bounded” is topological, as is the property “the database consists of (curved) lines only”. In contrast, the property “the database contains a straight line” is not. Apart from our interest in topological properties as a natural and mathematically well-motivated class of properties, they are also practically motivated by geographical information systems [6–8, 14, 18].

So far there was not much understanding yet of the class of topological properties that are first-order (i.e., expressible in $\text{FO}[\mathbf{R}]$), except for the feeling that this class must be rather meager. Indeed, many topological properties are *not* first-order; for example, one cannot express in $\text{FO}[\mathbf{R}]$ that the database is topologically connected.⁴ But exactly which topological properties *are* first-order?

Cone Logic. What we do understand quite well is when two given sets A and A' are *topologically elementary equivalent*. This means that any $\text{FO}[\mathbf{R}]$ -sentence that is topological will not distinguish between A and A' . Indeed, Paredaens and the present authors [15] discovered a characterization of topological elementary equivalence in terms of the *cone types* occurring in the two given databases. Semi-algebraic sets are topologically well-behaved in that locally around each point they are “conical” [4]. The cone of a point can either be completely filled (in case of points in the interior of the set), completely empty (in case of isolated points or points not in the set), or consisting of lines and regions arriving in the point. A database can be partitioned according to the cone types of its points. The characterization states that two databases are topologically elementary equivalent if and only if the cardinalities of the equivalence classes of their partitions match.

In this paper we introduce *Cone Logic* (\mathcal{CL}), in which only topological properties can be expressed. The logic \mathcal{CL} is two-tiered: at the bottom tier, there is a first-order logic for expressing properties of cones, which can talk about the lines and regions making up the cone, and their relative order in the cone. At the top tier, any sentence γ from the bottom tier can be used in an “atomic” formula of

² The subformula $-b < x$ is, of course, a shorthand for $(\exists z)(z + b = 0 \wedge z < x)$. Note that formally, we work in an expansion of first-order logic over the reals with a binary relation symbol S to address the content of the database. However, we will use the same notation $\text{FO}[\mathbf{R}]$ to denote this first-order query language.

³ A homeomorphism of the plane is a bijection $f : \mathbf{R}^2 \rightarrow \mathbf{R}^2$ such that both f and f^{-1} are continuous.

⁴ This follows from the combined results of Benedikt, Dong, Libkin and Wong [2] and Grumbach and Su [10].

the form $[\gamma](p)$, where p is a point variable; this formula expresses that the cone of p satisfies property γ . The only other atomic predicate at the top tier is the symbol S to address the database; the top tier is then closed under the standard first-order operations. An example of a sentence in \mathcal{CL} is

$$\forall p[\exists xR(x) \rightarrow \exists!xL(x)](p),$$

which expresses that at every point bordering a region (R), there can be at most one line (L) entering that region. Another example is

$$\exists p[\exists x\exists y\exists z\exists u(R(x) \wedge L(y) \wedge R(z) \wedge L(u) \wedge B(x, y, z) \wedge B(z, u, x))](p),$$

which expresses that there is a point where two regions meet, and through which a line runs between the two regions. (The predicate $B(x, y, z)$ denotes that cone element y lies between cone elements x and z .)

Note that while $\text{FO}[\mathbf{R}]$ talks about points in terms of their coordinates, \mathcal{CL} can only talk about points directly and does not even have access to their coordinates. Every property expressible in \mathcal{CL} is also expressible in $\text{FO}[\mathbf{R}]$. We investigate the question of the converse: *is \mathcal{CL} first-order complete? That is, is every first-order topological property expressible in \mathcal{CL} ?*

Circular languages. As a first illustration of the power of \mathcal{CL} , we show that *any* property of cones expressible in $\text{FO}[\mathbf{R}]$ can also be expressed in \mathcal{CL} . Since a non-trivial cone can be represented as a circular list of L 's and R 's, an arbitrary property of cones can be represented as a set of such circular lists; we call such a set a *circular language*. We prove for any circular language T that if “the cone of point (x, y) satisfies T ” is expressible in $\text{FO}[\mathbf{R}]$, then “the cone of point p satisfies T ” is expressible in \mathcal{CL} .

Region databases. A database is called a *region database* if, intuitively, it only contains “filled” figures. More precisely, the cone of every point in the database must either be completely full or consist exclusively of R 's (regions). Region databases appear often in geographical information systems.

With each region database we can associate an abstract directed graph of a very simple form. For each singular point p in the database there is a “parent” node in the graph with outgoing edges to n “child” nodes, where n is the number of R 's in the cone of p . The sets of child nodes for different parent nodes are disjoint. Importantly, by the above-mentioned characterization of topological elementary equivalence, any two topologically elementary equivalent region databases have the same associated abstract graph.

Our second main result is then that a topological property of region databases is expressible in $\text{FO}[\mathbf{R}]$ if and only if it is expressible in standard first-order logic when looking at the abstract graph of a database instead of at the database itself.⁵ Using a quantifier elimination procedure, we obtain as a corollary the first-order completeness of \mathcal{CL} on the class of region databases.

⁵ With “standard” first-order logic of graphs we mean first-order logic over one binary relation E , used to address the edges of the graph.

The general question of first-order completeness of \mathcal{CL} the class of all planar spatial databases remains open. Other open questions are to extend our results to databases consisting of multiple semi-algebraic sets (rather than just one), or to non-planar (e.g., 3D) databases.

Lifting collapse theorems. In the proofs of our completeness results we make heavy use of a powerful tool: a “collapse theorem” by Benedikt, Dong, Libkin and Wong [2]. This theorem says that any $\text{FO}[\mathbf{R}]$ -definable property of *finite* databases that is invariant under monotone bijections from \mathbf{R} to \mathbf{R} , is already expressible by a sentence that uses no arithmetic, except for the order predicate. So, this sentence mentions only the predicate $<$ and the relation symbol S for the database content.

Now \mathcal{CL} is subsumed by first-order logic over $(<, S)$. Hence, our first-order completeness result for \mathcal{CL} “lifts” collapse to the level of infinite, semi-algebraic, sets, which are much more relevant in the spatial context than finite databases.

Acknowledgment. We thank Jan Paredaens for a number of inspiring discussions we had with him in the initial stage of this work.

2 Preliminaries

Spatial databases. We denote the real numbers by \mathbf{R} , so \mathbf{R}^2 denotes the real plane. A *semi-algebraic set in \mathbf{R}^2* is a set of points that can be defined as

$$\{(x, y) \in \mathbf{R}^2 \mid \varphi(x, y)\},$$

where $\varphi(x, y)$ is a formula built using the Boolean connectives \wedge , \vee , and \neg from atoms of the form $P(x, y) > 0$, where $P(x, y)$ is a polynomial in the variables x and y with integer coefficients. Observe that $P = 0$ is equivalent to $\neg(P > 0) \wedge \neg(-P > 0)$, so equations can be used as well as inequalities.

In this paper, a *database* is defined as a semi-algebraic set in \mathbf{R}^2 that is closed in the ordinary topological sense. It is known [4] that these are precisely the finite unions of sets of points that can be defined as

$$\{(x, y) \in \mathbf{R}^2 \mid P_1(x, y) \geq 0 \wedge \dots \wedge P_m(x, y) \geq 0\}.$$

In other words, we disallow the essential use of strict inequalities in the definition of a database.

First-order logic over the vocabulary $(0, 1, +, \times, <, S)$, with S a binary relation symbol, is denoted by $\text{FO}[\mathbf{R}]$. An $\text{FO}[\mathbf{R}]$ -formula φ can be evaluated on a database A by letting variables range over \mathbf{R} , interpreting the arithmetic symbols in the obvious way, and interpreting $S(x, y)$ to mean that the point (x, y) is in A .

To formalize what it means for two databases A and B to be topologically the same, we use the notion of *isotopy*. An isotopy is a continuous deformation

of the plane;⁶ A and B are called *isotopic* if there is an isotopy h such that $h(A) = B$.⁷

An FO[\mathbf{R}]-sentence φ is called *topological* if whenever databases A and B are isotopic, then $A \models \varphi$ if and only if $B \models \varphi$. Finally, two databases A and B are called *topologically elementary equivalent* if for each topological sentence φ , $A \models \varphi$ if and only if $B \models \varphi$.

Cones. A known topological property of semi-algebraic sets [4] is that locally around each point they are conical. This is illustrated in Figure 1. For every point p of a semi-algebraic set A there exists an $\varepsilon > 0$ such that $D(p, \varepsilon) \cap A$ is isotopic to the planar cone with top p and base $C(p, \varepsilon) \cap A$.⁸ We thus refer to *the cone of p in A* .

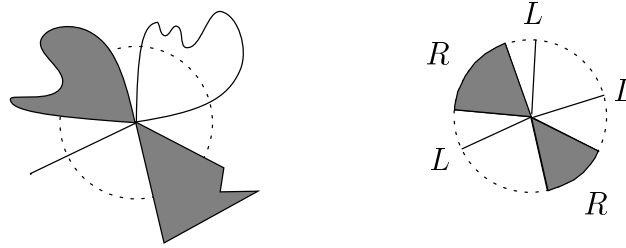


Fig. 1. A database and the cone of one of its points.

A database is also conical around the point at infinity.⁹ More precisely, there exists an $\varepsilon > 0$ such that $\{(x, y) \mid x^2 + y^2 \geq \varepsilon^2\} \cap A$ is isotopic to $\{\lambda \cdot (x, y) \mid (x, y) \in C((0, 0), \varepsilon) \cap A \wedge \lambda \geq 1\}$. We can indeed view the latter set as the cone with top ∞ and base $C((0, 0), \varepsilon) \cap A$, and call it *the cone of ∞ in A* .

We use the following finite representation for cones. The cone having a full circle as its base (which appears around interior points) is represented by the letter F . Any other cone can be represented by a circular list of L 's and R 's (for “line”

⁶ Formally, an isotopy is a homeomorphism of the plane that is isotopic to the identity. Two homeomorphisms f and g are isotopic if there is a continuous function $F : \mathbf{R}^2 \times [0, 1] \rightarrow \mathbf{R}^2$ such that for each $t \in [0, 1]$, the function $F_t : \mathbf{R}^2 \rightarrow \mathbf{R}^2 : p \mapsto F(p, t)$ is a homeomorphism and F_0 is f and F_1 is g .

⁷ A more relaxed notion of “being topologically the same” is to simply require that B is the image of A under a homeomorphism rather than an isotopy. The only difference between the two notions is that the latter considers mirror images to be the same, while the former does not. Indeed, every homeomorphism either is an isotopy itself, or is isotopic to a reflection [13]. All the results we will present under isotopies have close analogues under homeomorphisms.

⁸ $D(p, \varepsilon)$ is the closed disk with center p and radius ε ; $C(p, \varepsilon)$ is its bordering circle.

⁹ If we project \mathbf{R}^2 stereographically onto a sphere, the point at infinity corresponds to the missing point on the sphere.

and “region”) which describes the cone in a complete clockwise turn around the top. For example, the cone of Figure 1 is represented by $(LLRLR)$. The cone with empty base (which appears around isolated points) is represented by the empty list $()$. The set of all cones, represented in the way just explained, will be denoted by \mathbf{C} .

Let A be a database. The *point structure of A* is the function $\Pi(A)$ from $A \cup \{\infty\}$ to \mathbf{C} that maps each point to its cone in A . It can be shown that $\Pi(A)^{-1}$ is empty on all but a finite number of cones. Moreover, there are only three cones where $\Pi(A)^{-1}$ can be infinite: F , (LL) (the cone around points on curves), and (R) (the cone around points on the smooth border of a region). It can indeed be shown that in each database, the points with a cone different from these three are finite in number. The points are called the *singular* points of the database.

Let A and B be databases. We say that $\Pi(A)$ is *isomorphic* to $\Pi(B)$, denoted by $\Pi(A) \cong \Pi(B)$, if there is a bijection f from $A \cup \{\infty\}$ to $B \cup \{\infty\}$ with $f(\infty) = \infty$, such that $\Pi(A) = \Pi(B) \circ f$. Paredaens and the present authors gave the following characterization [15]:

Theorem 1. *Two databases A and B are topologically elementary equivalent if and only if $\Pi(A) \cong \Pi(B)$.*

3 Cone logic

In this section we introduce the logic \mathcal{CL} (cone logic). This is a two-tiered logic. At the bottom tier we have a first-order logic for expressing properties of cones. At the top tier we can use sentences from the bottom tier to talk about points in the database and their cones.

Logical properties of cones. Consider the vocabulary \mathcal{C} consisting of the propositional symbols F and E , the unary relation symbols L and R , and the ternary relation symbol B . First-order logic sentences over \mathcal{C} will be called \mathcal{C} -sentences.

An arbitrary cone can be viewed as a finite \mathcal{C} -structure as follows. The full cone F is viewed as the empty structure where proposition F is true (and proposition E is false); the empty cone $()$ is viewed as the empty structure where E is true (and F false). A cone of the form $(c_0 \dots c_{n-1})$, where each c_i is L or R , is viewed as the structure with domain $\{0, \dots, n-1\}$ in which propositions F and E are false; relation L equals $\{i \mid c_i = L\}$; relation R equals $\{i \mid c_i = R\}$; and relation B equals $\{(i, j, k) \mid 0 \leq (j - i) \bmod n < (k - i) \bmod n\}$. Relation B stands for “betweenness”: $B(i, j, k)$ holds if when we walk around the cone in clockwise order starting from element nr. i , we meet element nr. j before we meet element nr. k .

Under the above view we can evaluate \mathcal{C} -sentences on cones. For example, the cone $(RLLRL)$ satisfies the \mathcal{C} -sentence

$$\exists x \exists y \exists z \exists u (R(x) \wedge L(y) \wedge R(z) \wedge L(u) \wedge B(x, y, z) \wedge B(z, u, x)).$$

The logic \mathcal{CL} . Cone logic is first-order logic over the infinite vocabulary consisting of the constant symbol ∞ , the unary relation symbol S , and all unary relation symbols of the form $[\gamma]$, with γ a \mathcal{C} -sentence.

A \mathcal{CL} -formula can be evaluated on a database A in the following way: ∞ is interpreted by the point at infinity; $S(p)$ means that p is a point belonging to A ; and $[\gamma](p)$ means that the cone of p in A satisfies γ . Variables and quantifiers range over the points in the plane.

Since the cone structure of a database is left invariant by isotopies, we have:

Proposition 1. *Every property expressed by a \mathcal{CL} -sentence is topological.*

We also note: (proof delayed to the next section)

Proposition 2. *For every \mathcal{CL} -formula there is an equivalent $\text{FO}[\mathbf{R}]$ -formula.*

The natural question now arises: *is every topological property expressible in $\text{FO}[\mathbf{R}]$ also expressible in \mathcal{CL} ?* We investigate this problem, which we call the *first-order completeness of \mathcal{CL}* , in the following sections.

4 Circular languages

Let us call a *circular language* any cone property (i.e., a set of cones) that does not contain the two special cases of the full cone and the empty cone (these can be treated separately). So a circular language is a set of non-empty circular lists of L 's and R 's.

A circular language T is called $\text{FO}[\mathbf{R}]$ -*definable* if there is an $\text{FO}[\mathbf{R}]$ -formula $\varphi(x, y)$ such that for each database A and each point $(x_0, y_0) \in A$, $A \models \varphi[x_0, y_0]$ iff the cone of (x_0, y_0) in A belongs to T . We are going to show:

Theorem 2. *Every $\text{FO}[\mathbf{R}]$ -definable circular language T is definable by a \mathcal{C} -sentence.*

Before we sketch the proof, we remark that it is easy to characterize the \mathcal{C} -definable circular languages. Let T be an arbitrary set of words over the alphabet $\{L, R\}$. We can turn T into a circular language T^{circ} by circularizing every word in T . It is well known [19] how words over the alphabet $\{L, R\}$ can be viewed as finite structures over the vocabulary consisting of the unary relation symbols L and R , and the order predicate $<$. The first-order definable sets of words are then precisely the star-free regular languages. Using this fact, the following is not difficult to see:

Proposition 3. *The circular languages definable by \mathcal{C} -sentences are precisely those of the form T^{circ} , with T a star-free regular language over $\{L, R\}$.*

Using this fact, we can prove Proposition 2 using induction on the star-free regular expressions. We omit the details.

We now present:

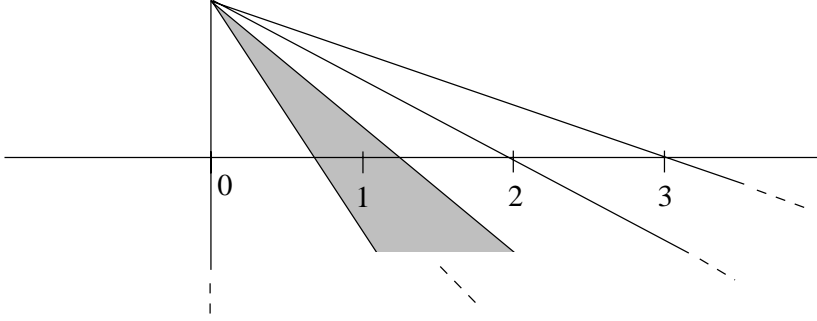


Fig. 2. Construction of $A(\Delta)$.

Proof of Theorem 2. (Sketch) Let φ be the $\text{FO}[\mathbf{R}]$ -formula defining T . Let T^{word} be the set of all words over $\{L, R\}$ that belong to T when viewed as circular lists.

Consider finite (L, R) -structures Δ over the reals, with L and R unary relation symbols. We first show that “ $\Delta \in T^{\text{word}}$ ” (meaning that the elements of L^Δ and R^Δ , when scanned from right to left, spell out a word in T^{word}) is expressible by a first-order sentence over the vocabulary $(L, R, <)$, where $<$ is the real ordering. Indeed, we can find an $\text{FO}[\mathbf{R}]$ -formula (over the vocabulary $(0, 1, +, \times, <, L, R)$) that defines, on any Δ , a database $A(\Delta)$ that is conical with top $(0, 1)$ and that contains through each element of L^Δ or R^Δ (embedded on the x -axis of \mathbf{R}^2) a line or triangular strip. This is illustrated in Figure 2 for $L^\Delta = \{0, 2, 3\}$ and $R^\Delta = \{1\}$.

Then “ $\Delta \in T^{\text{word}}$ ” is equivalent to $A(\Delta) \models \varphi[0, 1]$. The latter sentence is also invariant under monotone bijections from \mathbf{R} to \mathbf{R} . Hence, by a collapse theorem by Benedikt, Dong, Libkin and Wong [2], the sentence is equivalent to a sentence ψ over $(L, R, <)$. Moreover, we may assume without loss of generality that the quantifiers in ψ range only over the active domain of the given structure [16, 3].

Consider now the \mathcal{C} -sentence $\gamma = \exists f \psi'$, where ψ' is obtained from ψ by replacing all occurrences of $x < y$ by $B(f, x, y)$. Then γ defines the circular language T . \square

5 Region databases

In this section we focus on *region databases*, defined as databases in which the cone of every point is either F or consists exclusively of R 's. Intuitively, such databases contain only “filled” figures. We are going to show:

Theorem 3. \mathcal{CL} is first-order complete on the class of region databases.

We will prove Theorem 3 by establishing a connection between topological properties and logical properties of abstract graphs.

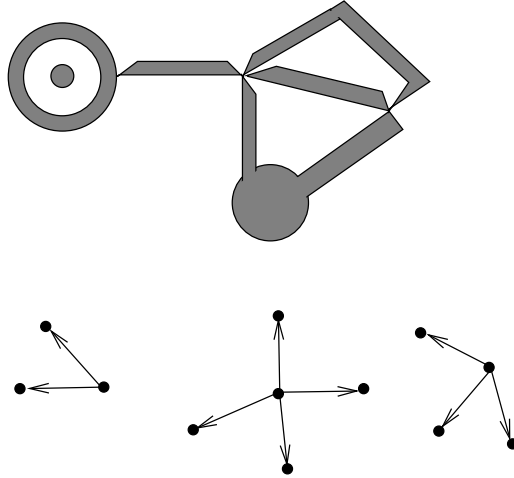


Fig. 3. At the top, a database A ; at the bottom, $\text{graph}(A)$.

Let A be a (fully two-dimensional) database, and let p be a singular point in A . The number of R 's in the cone of p in A is called the *degree* of p in A and denoted by $\text{deg } p$. We can now associate an abstract directed graph to A , denoted by $\text{graph}(A)$, as follows (an illustration is given in Figure 3). The set of nodes of $\text{graph}(A)$ equals the union of the set of singular points in A with the set $\{(p, i) \mid p \text{ a singular point in } A \text{ and } 1 \leq i \leq \text{deg } p\}$.¹⁰ The nodes in the first set are called *parent nodes*; the nodes in the second set are called *child nodes*. There is an edge from each node p to each node (p, i) .

The graphs that equal $\text{graph}(A)$ for some A are called the *depth-one forests*. Also note that, by Theorem 1, if $\text{graph}(A)$ and $\text{graph}(B)$ are isomorphic then A and B are topologically elementary equivalent.

We view directed graphs in the usual manner as structures over the vocabulary consisting of a single binary relation symbol E ; the domain equals the set of nodes, and relation E equals the set of edges. We refer to first-order logic over this vocabulary $\{E\}$ as *FO of graphs*. To prove Theorem 3, we will also need to talk about ordered graphs. These are graphs with an additional order predicate $<$, which is an arbitrary linear order on all nodes. First-order logic over $(E, <)$ will be referred to as *FO of ordered graphs*.

Let \mathbf{G} be some class of graphs. An FO sentence φ of ordered graphs is called *order-invariant over \mathbf{G}* if it does not distinguish between different orderings of the same graph in \mathbf{G} . It is well known (e.g., [1, Exercise 17.27], [5, Proposition 2.5.6]) that in general, order-invariant FO sentences of ordered graphs are

¹⁰ For simplicity of presentation in this section, we ignore the point at infinity. It can be accommodated for by adding a few technicalities.

more powerful than standard FO sentences of graphs. However, for \mathbf{G} the class of depth-one forests, we can prove:

Proposition 4. *Every FO sentence of ordered graphs that is order-invariant over depth-one forests is equivalent (on depth-one forests) to a standard FO sentence of graphs.*

Proof. (Sketch) We rewrite FO formulas working on depth-one forests in a “many-sorted normal form” where variables range either only over parent nodes or only over child nodes. An ordering of a depth-one forest is called “canonical” if all parent nodes come first in the order, ordered by increasing number of children, followed by the child nodes ordered according to their parents.

We can show that on canonically ordered depth-one forests,¹¹ every FO formula is equivalent to a quantifier-free one over the expansion of the vocabulary $(E, <)$ with the following constants, functions and predicates. For each n , we have constants for the minimal parent having at least n children and the maximal parent having at most n children, as well as constants for the globally minimal and maximal parent. For each n , we have the binary predicate among parent nodes that the number of nodes between them in the order is at least n . We have functions giving the minimal and maximal child of a parent node. We have the sibling relation among child nodes. We have functions giving the minimal and maximal sibling of a child node. Finally, for each n , we have the binary predicate among child nodes that the number of nodes between them in the order is at least n .

The quantifier elimination procedure proceeds inductively as usual, distinguishing between parent variables and child variables in eliminating quantifiers, and adding the necessary extra predicates where needed to preserve equivalence. A quantifier-free sentence in this expanded logic can only talk about bounds on cardinalities of sets of parents defined by bounds on their number of children. Such properties are already expressible in standard FO of graphs. \square

Having Proposition 4 at our disposal, we can now establish the following upper bound on the topological properties expressible in $\text{FO}[\mathbf{R}]$:

Lemma 1. *For every topological $\text{FO}[\mathbf{R}]$ -sentence φ there exists an FO sentence ψ of graphs such that for each database A , $A \models \varphi$ iff $\text{graph}(A) \models \psi$.*

Proof. (Sketch) Given a depth-one forest G embedded in the reals, we can construct a database $A(G)$ topologically elementary equivalent to any database A for which $\text{graph}(A)$ and G are isomorphic. This construction is illustrated in Figure 4. Actually, we can even find an $\text{FO}[\mathbf{R}]$ -formula θ (mentioning the relation symbol E) that performs this construction (i.e., that defines $A(G)$ given any G). Hence, the sentence ψ being the composition of θ with φ is the wanted sentence, if it were not for the fact that ψ is not in FO of graphs but rather

¹¹ Considering order-invariant sentences, it is sufficient to restrict attention to depth-one forests that are canonically ordered.

in the much richer logic $\text{FO}[\mathbf{R}]$. However, ψ is order-generic, so by the collapse theorem already used in the proof of Theorem 2, we may assume it to be over the vocabulary $(E, <)$ only. Again, we may even assume that quantifiers in ψ range over the active domain only; hence ψ is an FO sentence of ordered graphs. Finally, since ψ is order-invariant, applying Proposition 4 yields the desired FO sentence of graphs. \square

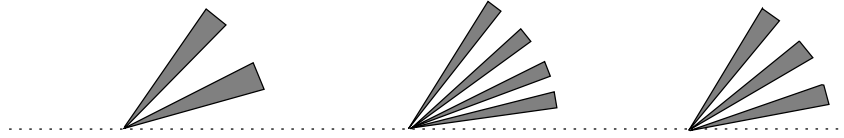


Fig. 4. The database $A(G)$ for some real embedding G of the graph of Figure 3. The parent nodes are placed on the real axis at the positions of their corresponding real numbers. The children are then drawn around the parents as regions, in the directions obtained from their corresponding real numbers (after a scaling from \mathbf{R} to \mathbf{R}^+). Note that $A(G)$ is topologically elementary equivalent to the database of Figure 3; this is a crucial property of the construction.

Theorem 3 now follows immediately from Lemma 1 and the following counterpart to it:

Lemma 2. *For every FO sentence ψ of graphs there exists a \mathcal{CL} -sentence φ such that for each database A , $\text{graph}(A) \models \psi$ iff $A \models \varphi$.*

Proof. (Sketch) From the proof of Proposition 4 (which dealt with order-invariant FO sentences of ordered graphs and thus certainly applies to standard FO sentences of graphs), we may assume ψ to be quantifier-free, provided extra predicates are provided for talking about bounds on cardinalities of sets of parents defined by bounds on the number of their children. Translated from $\text{graph}(A)$ to A this represents bounds on the cardinalities of sets of singular points defined by bounds on their degree. But such properties are expressible in \mathcal{CL} . \square

6 Discussion

The most obvious direction for further research is to extend our completeness result for \mathcal{CL} from the class of region databases to the general class of all (closed) databases. The point where our proof fails for the general case is the construction illustrated in Figure 4, where we construct, in $\text{FO}[\mathbf{R}]$, from any real embedding of $\text{graph}(A)$, for any region database A , a database topologically elementary equivalent to A . If A is not a region database, we cannot simply draw the children as regions emanating from their parents, as done in the figure; now some children

have to be drawn as lines. The endpoints of these lines must be pairwise connected; they are unwanted extra singular points and cannot be left “dangling.” The problem, however, is that this seems impossible to do in FO[**R**].

References

1. S. Abiteboul, R. Hull, and V. Vianu. *Foundations of Databases*. Addison-Wesley, 1995.
2. M. Benedikt, G. Dong, L. Libkin, and L. Wong. Relational expressive power of constraint query languages. *Journal of the ACM*, 45(1):1–34, 1998.
3. M. Benedikt and L. Libkin. On the structure of queries in constraint query languages. In *Proceedings 11th IEEE Symposium on Logic in Computer Science*, pages 25–34. IEEE Computer Society Press, 1996.
4. J. Bochnak, M. Coste, and M.-F. Roy. *Géométrie Algébrique Réelle*. Springer-Verlag, 1987.
5. H.-D. Ebbinghaus and J. Flum. *Finite Model Theory*. Springer, 1995.
6. M. Egenhofer and R. Franzosa. Point-set topological spatial relations. *Int. J. Geographical Information Systems*, 5(2):161–174, 1991.
7. M. Egenhofer and R. Franzosa. On the equivalence of topological relations. *Int. J. Geographical Information Systems*, 9(2):133–152, 1995.
8. M. Egenhofer and D. Mark. Modeling conceptual neighborhoods of topological line-region relations. *Int. J. Geographical Information Systems*, 9(5):555–565, 1995.
9. J. Flum and M. Ziegler. *Topological Model Theory*, volume 769 of *Lecture Notes in Mathematics*. Springer-Verlag, 1980.
10. S. Grumbach and J. Su. Queries with arithmetical constraints. *Theoretical Computer Science*, 173(1):151–181, 1997.
11. C.W. Henson, C.G. Jockusch, Jr., L.A. Rubel, and G. Takeuti. *First order topology*, volume CXLIII of *Dissertationes Mathematicae*. Polska Akademia Nauk, 1977.
12. P.C. Kanellakis, G.M. Kuper, and P.Z. Revesz. Constraint query languages. *Journal of Computer and System Sciences*, 51(1):26–52, August 1995.
13. E.E. Moise. *Geometric Topology in Dimensions 2 and 3*, volume 47 of *Graduate Texts in Mathematics*. Springer, 1977.
14. C.H. Papadimitriou, D. Suciú, and V. Vianu. Topological queries in spatial databases. In *Proceedings 15th ACM Symposium on Principles of Database Systems*, pages 81–92. ACM Press, 1996.
15. J. Paredaens, B. Kuijpers, and J. Van den Bussche. On topological elementary equivalence of spatial databases. In F. Afrati and Ph. Kolaitis, editors, *Database Theory—ICDT’97*, volume 1186 of *Lecture Notes in Computer Science*, pages 432–446. Springer, 1997.
16. J. Paredaens, J. Van den Bussche, and D. Van Gucht. First-order queries on finite structures over the reals. *SIAM Journal on Computing*, 27(6):1747–1763, 1998.
17. A. Pillay. First order topological structures and theories. *Journal of Symbolic Logic*, 52(3), September 1987.
18. D. Thompson R. Laurini. *Fundamentals of Spatial Information Systems*. Number 37 in APIC Series. Academic Press, 1992.
19. W. Thomas. Languages, automata, and logic. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Language Theory*, volume III. Springer, 1997.