Query Languages with Arithmetic and Constraint Databases

Leonid Libkin
Bell Laboratories
600 Mountain Avenue
Murray Hill, NJ 07974
E-mail: libkin@research.bell-labs.com

1 Introduction

Can we store an *infinite set* in a database? Clearly not, but instead we can store a *finite representation* of an infinite set and write queries as if the entire infinite set were stored. This is the key idea behind *constraint databases*, which emerged relatively recently as a very active area of database research. The primary motivation comes from geographical and temporal databases: how does one store a region in a database? More importantly, how does one design a query language that makes the user view a region as it if were an infinite collection of points stored in the database?

Finite representations used in constraint databases are first-order formulae; in geographical applications, one often uses Boolean combinations of linear or polynomial inequalities. One of the most challenging questions in the development of the theory of constraint databases was that of the expressive power: what are the limitations of query languages for constraint databases? These questions were easily reduced to those on the expressiveness of query languages over ordinary relational databases, with additional condition that databases may store numbers and arithmetic operations may be used in queries.

It turned out that the classical techniques for analyzing the expressive power of relational query languages no longer work in this new setting. In the past several years, however, most questions on the expressive power have been settled, by using new techniques that mix the finite and the infinite, and bring together results from a number of fields such as model theory, algebraic geometry and symbolic computation.

In this column we briefly survey of some of the results on expressiveness of query languages for constraint databases. Mathematically, these can be viewed as results on expressiveness of logics over finite or definable sets embedded in certain structures. We first deal with the finite case, that is formalized by *embedded finite models*. We give a new type of expressivity bounds – collapse results – and explain how they can be applied in the setting of constraint databases.

2 Embedded finite models and relational databases

The definition of embedded finite models is rather simple. We consider a first-order language L and an L-structure $\mathcal{M}$ with an infinite universe U. Consider a separate first-order language $SC = (R_1, \ldots, R_n)$ that consists of relational symbols only. (The notation SC comes from the database name "schema".) A SC structure on $\mathcal{M}$ (or an embedded finite model) is a structure

$$\langle U, L^{\mathcal{M}}, R_1^{\mathcal{M}}, \ldots, R_n^{\mathcal{M}} \rangle$$

where each $R_i^{\mathcal{M}}$ is a *finite* subset of U^{k_i}, where k_i is the arity of R_i, and $L^{\mathcal{M}}$ is the interpretation of L in $\mathcal{M}$. In this setting, we want to study the expressive power of first-order logic (FO) with respect to the finite SC-reduct. That is, we look at sentences like $\exists x \exists y \ R(x, y) \wedge y \cdot x = x + 2$ that states the existence of a specific edge (x, y) in a graph whose nodes are numbers. We want to know what one can express over such finite structures embedded in an infinite one.

At the first glance, the problem may appear a bit esoteric, and certainly more appropriate for the Logic Column than for the Database one. To prove that the first impression is false, let me explain how this problem naturally arises in the theory of relational databases. A relational database is just a finite relational structure. Databases are queried using languages that correspond to first-order logic (FO) and its fragment (in fact, the core SQL – minus aggregation – has precisely the power of FO). These languages usually come in two flavors: declarative, like *relational calculus* (which is just FO), and procedural, like relational algebra, that give evaluation mechanism for declarative queries.

When relational algebra and calculus are introduced in textbooks, the conditions for selecting tuples from relations are usually defined to be boolean combinations of $x = y$ and $x < y$, where x and y are variables or constants. This is often followed by a number of examples faithful to the definitions, as well as examples of the kind "find employees who make at least 90% of their manager's salary," which involves conditions of the form $x > 0.9 \cdot y$. This addition corresponds exactly to the formal setting introduced in the first paragraph: we may assume that $\mathcal{M}$ is, for example, the real field $\mathbf{R} = \langle \mathbb{R}, +, \cdot, 0, 1, < \rangle$, and one can thus write queries such as $\varphi(x) \equiv \exists y, s_1, s_2\ (M(x, y) \wedge S(x, s_1) \wedge S(y, s_2) \wedge s_1 > 0.9 \cdot s_2)$, where M records the employee-manager relation, and S has salaries of all employees.

How does one analyze this very natural extension of the standard relational calculus? We know many results about expressivity of FO without the interpreted structure $\mathcal{M}$; in fact, much of the development of finite-model theory was motivated by database problems. Standard techniques such as 0-1 laws [15] and locality [16] tell us that queries such as *parity of cardinality* or *the transitive closure of a graph* are inexpressible in FO. However, these techniques become inapplicable in the embedded setting. Other techniques, such as Ehrenfeucht-Fraïssé games, become extremely awkward to apply with interpreted operations present.

The full scope of interaction between finite databases and fixed, possibly infinite, interpreted structure, has not, until recently, been fully explored in the database community, although the question was already raised in the seminal paper of Chandra and Harel [13]. One of the reasons the database community started looking at these problems recently is the emergence of *constraint databases* as a very active area of research. We discuss them next.

3 Constraint Databases

Constraint databases were invented about ten years ago [23] and have become a well-established topic in the database field. The mathematical model is very simple. We consider the setting of embedded finite models, and instead of interpreting each n-ary symbol S from SC as a finite n-relation, we interpret it as *definable* subset of U^n. That is, the interpretation of S is $\{\vec{a} \in U^n \mid \mathcal{M} \models \alpha(\vec{a})\}$ for some formula α in the language L.

The key idea is that the formula α gives a *finitary* representation of an *infinite* objects. Consider, for example, sets definable over the real ordered group $\mathbf{R}_{\text{lin}} = \langle \mathbb{R}, +, -, 0, 1, < \rangle$. These are Boolean combinations of sets definable with linear inequalities of the form $\sum_{i=1}^{n} a_i \cdot x_i \leq b$, with coefficients a_is being rational – this follows from the fact that over $\mathbf{R}_{\text{lin}}$, every formula is equivalent to a quantifier-free one (one also says that $\mathbf{R}_{\text{lin}}$ admits *quantifier-elimination*). Sets arising in such a way are called *semi-linear* [11, 33].

Such representation is very convenient in geographical applications, as regions are often represented as semi-linear sets on the real plane $\mathbb{R}^2$. If linear constraints are not sufficient, one can use *polynomial* ones. That is, one can consider sets definable over the real field $\mathbb{R}$. Such sets, called *semi-algebraic* [11, 33], are Boolean combinations of sets definable by polynomial inequalities $p(x_1, \ldots, x_n) \leq 0$. This again follows from the classical result of Tarski's [32] that the real field has quantifier-elimination.

We use the notation $\mathrm{FO} + \textsc{Poly}$ for FO over the real field and SC (FO with polynomial constraints) and $\mathrm{FO} + \textsc{Lin}$ for FO over the real ordered group and SC (FO with linear constraints). An example of definability in $\mathrm{FO} + \textsc{Poly}$ is the property that all points in a relation S lie on a common circle: $\exists a \exists b \exists r \ (\forall x \forall y \ S(x, y) \rightarrow (x - a)^2 + (y - b)^2 = r^2)$. In general, $\mathrm{FO} + \textsc{Poly}$ can define many useful topological concepts such as closure, interior and boundary. These are definable in $\mathrm{FO} + \textsc{Lin}$ as well. In $\mathrm{FO} + \textsc{Poly}$ one can also define the convex hull of a set. To see how this is done in the two-dimensional case, assume that a set a semi-algebraic set $S \in \mathbb{R}^2$ is given. Then $\varphi(x, y)$ given by

$$\begin{aligned}\exists x_1, y_1, x_2, y_2, x_3, y_3 \ \ \exists \lambda_1 \lambda_2 \lambda_3 \ (S(x_1, y_1) \wedge S(x_2, y_y) \wedge S(x_3, y_3) \wedge \lambda_1 \geq 0 \wedge \lambda_2 \geq 0 \wedge \lambda_3 \geq 0 \wedge \\ (x = \lambda_1 \cdot x_1 + \lambda_2 \cdot x_2 + \lambda_3 \cdot x_3) \wedge (y = \lambda_1 \cdot y_1 + \lambda_2 \cdot y_2 + \lambda_3 \cdot y_3))\end{aligned}$$

is true on (x, y) iff $(x, y) \in \operatorname{conv}(X)$. In general, to definite the convex hull of a set in $\mathbb{R}^n$, one uses Carathéodory's theorem stating that x is in the convex hull of $X \subseteq \mathbb{R}^n$ iff x is in the convex hull of some $n + 1$ points in X, and codes this by a FO formula just as we did above for the case of $\mathbb{R}^2$. We note in passing that multiplication is essential for definability of the convex hull; in fact, even adding a predicate for collinearity to $\mathrm{FO} + \textsc{Lin}$, one gets the full power of $\mathrm{FO} + \textsc{Poly}$.

Thus, $\mathrm{FO} + \textsc{Poly}$ is a rather expressive language to talk about semi-algebraic sets. In fact, many typical GIS queries can already be expressed in the weaker language $\mathrm{FO} + \textsc{Lin}$. We next turn to a very basic topological property: *connectivity*. Suppose we are given a semi-algebraic or semi-linear set S, and we want to test if it is topologically connected. Can we do this in $\mathrm{FO} + \textsc{Poly}$ or $\mathrm{FO} + \textsc{Lin}$?

The first reaction is to say 'no'. Indeed, it is well known that over finite structures, FO cannot define graph connectivity (see, e.g., [14]), and topological connectivity appears to be a natural continuous analog of graph connectivity. This intuition was made precise in [29] where it was shown that if FO + Poly defines topological connectivity of semi-algebraic sets, then it defines graph connectivity of finite graphs. (The idea of the reduction is to embed a finite graph G into $\mathbb{R}^3$ without self-intersections.) This reduces the problem of topological connectivity of semi-algebraic sets to the problem of definability of graph connectivity of finite graphs. However, by definability we still mean definability in FO + Poly. That is, we now deal with a problem that perfectly fits the setting of embedded finite models: we have a finite graph whose nodes are in $\mathbb{R}$, and we want to see if in FO over the language of the graph and the real field, one can say that the graph is connected.

Questions of this kind were asked in early 90s; after several partial results (all inherently limited to deal with languages like FO + Poly), definite answers appeared a few years ago. In fact, by now we have a very good picture of the expressive power of FO over embedded finite models, depending on the the properties of the underlying structure $\mathcal{M}$. But before we survey those, we must deal with the crucial issue of the semantics of FO in the embedded setting.

4 Semantics of first-order queries

We use two simple examples to review the concepts of *active* and *natural* semantics in the relational model. Assume that we have a finite binary relation S on a set A, which is a subset of some infinite set U. Suppose we want to test if S is reflexive. The most obvious way to write this in FO is $\forall x\ S(x,x)$. How do we interpret $\forall x$ here? It is essential that we interpret it as "for all $x \in A$." Indeed, if by $\forall x$ we mean "for all $x \in U$," then the result of the above query is always false: as A is finite and U is infinite, there is always some $x \in U - A$ for which $S(x,x)$ fails.

The set A in the above example is what is usually called the *active domain* of a database – the set of all constants that occur in relations of a database D. We denote this by $adom(D)$. Thus, one possible semantics of FO queries is the *active semantics*: under this semantics, $\exists x \varphi(x)$ is interpreted as "there exists x in $adom(D)$."

For the next example, consider the real field $\mathbf{R} = \langle \mathbb{R}, +, \cdot, 0, 1, < \rangle$, and let again S be a finite binary relation on a set $A \subset \mathbb{R}$. Suppose we want to ask the following query: do all elements in S, considered as points in $\mathbb{R}^2$, lie on a line? To answer this, one has to test for the existence of a slope a and a base b such that $y = ax + b$ holds for every (x, y) in S. That is, the query is $\exists a \exists b \forall x \forall y\ (S(x, y) \rightarrow y = ax + b)$. In contrast to the previous example, here the quantifiers $\exists a$ and $\exists b$ are to be interpreted as "exist real numbers a and b." Indeed, if we restrict the range of these two quantifiers to A, the query above may produce a wrong result, as there is no guarantee that a and b will be found in A.

The interpretation under which $\exists x$ and $\forall x$ mean "exists $x \in U$" and "for all $x \in U$," where U is the infinite universe of a structure over which databases are interpreted, is called the *natural semantics*, or natural interpretation.

Notation: $\mathrm{FO}(\mathcal{M}, SC)$ denotes the class of queries on SC-structures definable in FO over $\mathcal{M}$, under the natural interpretation; likewise, we use the notation $\mathrm{FO}_{\mathrm{act}}(\mathcal{M}, SC)$ for the active interpretation. Since $adom(D)$ is definable in FO, we have $\mathrm{FO}_{\mathrm{act}}(\mathcal{M}, SC) \subseteq \mathrm{FO}(\mathcal{M}, SC)$. When we write $\mathrm{FO} + \textsc{Poly}$ and $\mathrm{FO} + \textsc{Lin}$, we mean the natural interpretation of formulae.

5 Collapse results

Our goal is to obtain general results on limitations of the expressive power of a query language. Our main tools are *collapse results*: these say that query class A is contained in query class B, where A is a class that could a priori be much bigger than B. Such results answer most questions on expressive power in the embedded context.

An important class of collapse result is *generic collapse*, describing expressiveness with respect to 'pure', or generic, queries. Generic queries are those that conform to the principle of *data independence*. Formally, they commute with permutations on the domain. Informally, this means that database entries are just "tags" of objects and one cannot use any extra information about them. Connectivity is an example of a generic query. So are many others: acyclicity, parity of cardinality, etc. In fact, every query that can be expressed in traditional relational languages like relational algebra and datalog, is generic.

We shall deal with three kinds of collapse results:

Natural-Active Collapse We say that a structure $\mathcal{M}$ admits this collapse if $\mathrm{FO}(\mathcal{M}, SC) = \mathrm{FO}_{\mathrm{act}}(\mathcal{M}, SC)$; that is, every query definable under the natural semantics is definable under the active semantics.

Active Generic Collapse Let $\mathcal{M}$ be ordered (that is, $<$ is one of the symbols in L, interpreted as a linear order on the universe U). We say that $\mathcal{M}$ admits the active-generic collapse if the classes of generic queries in $\mathrm{FO}_{\mathrm{act}}(\mathcal{M}, SC)$ and $\mathrm{FO}_{\mathrm{act}}(\langle U, <\rangle, SC)$ are the same; that is, every query definable under the active-domain semantics with $\mathcal{M}$-constraints is definable under the active-domain semantics with just order constraints.

Natural Generic Collapse over ordered $\mathcal{M}$ states the classes of generic queries in $\mathrm{FO}(\mathcal{M}, SC)$ and $\mathrm{FO}(\langle U, <\rangle, SC)$ are the same; that is, every query definable under the natural semantics with $\mathcal{M}$-constraints is definable under the natural semantics with just order constraints.

As an illustration of the power of these collapse results, let us sketch the proof that graph connectivity is not expressible in $\mathrm{FO} + \mathrm{POLY}$ (assuming that we have already proved the above collapse results for the real field $\mathbf{R}$). Assume to the contrary that connectivity is definable in $\mathrm{FO} + \mathrm{POLY}$. By the natural-active collapse, it is then definable in $\mathrm{FO} + \mathrm{POLY}$ under the active semantics; being generic, it is thus definable with just the order relation. However, it well known that graph connectivity is not definable in FO, over finite ordered graphs [21].

We now consider the *active generic collapse*. This is the easiest kind of collapse results, and it turns out that it works over all ordered structures:

Theorem 1 ([6, 27]) *Any ordered structure $\mathcal{M}$ admits the active generic collapse.* □

The proof – relatively easy – is based on Ramsey's theorem. In fact, one shows that for any infinite set $X \subseteq U$ and any $\mathrm{FO}_{\mathrm{act}}(\mathcal{M}, SC)$ formula $\alpha(\vec{x})$, one can find an infinite subset $Y \subseteq X$ and a $\mathrm{FO}_{\mathrm{act}}(\langle U, <\rangle, SC)$ formula $\beta(\vec{x})$ such that $D \models \alpha(\vec{a}) \leftrightarrow \beta(\vec{a})$ holds whenever $adom(D) \subset Y$ and all components of $\vec{a}$ are in Y. Using this, and genericity, one derives the collapse result.

And what about the collapse to pure FO, without an order relation? In general, it cannot be guaranteed, as $\mathrm{FO}_{\mathrm{act}}(SC) \neq \mathrm{FO}_{\mathrm{act}}(\langle U, <\rangle, SC)$ (see, e.g., [1] for a separating example). However, for several classes of structures the collapse to equality can be proved: for example, for $\langle \mathbb{R}, +, \cdot, e^x \rangle$ [9].

We next deal with the *natural-active collapse*. Can it possibly be proved for all structures? The answer, as was pointed out in [20], is clearly 'no': over $\mathcal{N} = \langle \mathbb{N}, +, \cdot \rangle$ every computable property of finite SC-structures is expressible by a FO sentence (simply by coding), while we saw that graph connectivity, for example, is not in $\mathrm{FO}_{\mathrm{act}}(\mathcal{N}, SC)$.

To find a suitable class of structures over which the collapse holds, we consider the model theoretic notion of o-minimality [31, 33]. An ordered structure $\mathcal{M}$ is called *o-minimal* if every definable set is a finite union of points and open intervals. A definable set is one of the form $\{c \mid \mathcal{M} \models \varphi(c)\}$ where φ is a formula in the language of $\mathcal{M}$, possibly supplemented by constants for some elements of $\mathcal{M}$. Classical examples of o-minimal structures include the real field $\mathbf{R}$ and the real ordered group $\mathbf{R}_{\mathrm{lin}}$ (as an easy consequence of Tarski's quantifier-elimination and the fundamental theorem of algebra). More recently, it was shown that the exponential field $\langle \mathbb{R}, +, \cdot, e^x \rangle$ is o-minimal [34].

Theorem 2 ([8, 9]) *Let $\mathcal{M}$ be an o-minimal structure admitting quantifier elimination. Then $\mathcal{M}$ admits the natural-active collapse:* $\mathrm{FO}(\mathcal{M}, SC) = \mathrm{FO}_{\mathrm{act}}(\mathcal{M}, SC)$. □

Thus, the natural-active collapse holds for FO + POLY and FO + LIN. The first result on the natural-active collapse was the Hull-Su theorem [22], treating the case of FO without any interpreted structure, but over an infinite universe. The relatively complicated proof of [22] was later simplified in [9, 12]. Even before [22], the "4 Russians" paper [3] showed that quantification over finite domains suffices to recover the full power of $\mathrm{FO}(\langle U, \emptyset \rangle, SC)$.

The natural-active collapse for FO + LIN (that is, over $\mathbf{R}_{\mathrm{lin}}$) was established – by constructive means – in [30]. Theorem 2 was first proved nonconstructively in [8], and then [9] gave an *algorithm* for converting every $\mathrm{FO}(SC, \mathcal{M})$ formula into an equivalent $\mathrm{FO}_{\mathrm{act}}(SC, \mathcal{M})$ one (assuming that the quantifier elimination is effective). In general, the complexity of such transformation depends on the complexity of the quantifier elimination procedure. For the case of FO + POLY, a partial complexity analysis was given in [5], which handled the case of schemas SC containing only unary predicates.

The *natural generic collapse* can now be obtained as a corollary of the two previous theorems:

Corollary 1 *Any o-minimal structure $\mathcal{M}$ admits the natural generic collapse.* □

Note that we no longer need the quantifier-elimination assumption, as we can always take a definable extension that admits quantifier-elimination (extra predicates do not hurt when we deal with the active semantics, as all ordered structures admit the active generic collapse).

Corollary 1 first appeared in [6] where it was proved using techniques from nonstandard analysis. It was recently generalized to other structures, including quasi-o-minimal ones (see [7] for the definition and the proof) and those of finite *VC dimension*. We first recall the definition (cf. [2]). Suppose X is an infinite set, and $\mathcal{C} \subseteq 2^X$ is a family of subsets of X. Let $F \subset X$ be finite; we say that $\mathcal{C}$ *shatters* F if the collection $\{F \cap C \mid C \in \mathcal{C}\}$ is 2^F. The *Vapnik-Chervonenkis (VC) dimension* of $\mathcal{C}$ is the maximal cardinality of a finite set shattered by $\mathcal{C}$, or ∞ if arbitrarily large finite sets are shattered by $\mathcal{C}$.

Let $\varphi(\vec{x}, \vec{y})$ be a formula in the language of $\mathcal{M}$ with $\mid \vec{x} \mid = n, \mid \vec{y} \mid = m$. For each $\vec{a} \in U^n$, define $\varphi(\vec{a}, \mathcal{M}) = \{\vec{b} \in U^m \mid \mathcal{M} \models \varphi(\vec{a}, \vec{b})\}$, and let $F_\varphi(\mathcal{M})$ be $\{\varphi(\vec{a}, \mathcal{M}) \mid \vec{a} \in U^n\}$. Families of sets arising in such a way are called *definable families*. We say that $\mathcal{M}$ has finite VC dimension if every definable family over $\mathcal{M}$ has finite VC dimension. Examples include all o-minimal structures [26] as well as quasi-o-minimal ones [4] (including, for example, $\langle \mathbb{N}, +, < \rangle$).

Theorem 3 (see [4]) *Let $\mathcal{M}$ have finite VC dimension. Then $\mathcal{M}$ admits the natural generic collapse.* $\square$

When the collapse fails We have seen one example of the failure of the natural-active collapse: $\mathcal{N} = \langle \mathbb{N}, +, \cdot \rangle$. However, this structure is highly undecidable. Can one find a structure with a decidable first-order theory over which the natural-active collapse fails?

To see an example, let $\mathcal{RG} = \langle U, E \rangle$ be the random graph on a countably infinite set U: that is, any model that satisfies every sentence that is true in almost all finite graphs. Here 'almost all' is with respect to the uniform probability distribution: $E(a, b)$ holds of nodes a, b with probability one half, independently for each pair a, b. It is known [15] that the set of all such sentences forms a complete theory with infinite models, and that this theory is decidable.

Proposition 1 (see [27]) *The natural-active collapse fails over the random graph.* $\square$

The idea of the proof is to use the extension axioms to simulate monadic second-order logic over SC structures, using nodes of the random graph. As monadic second-order logic is more expressive than FO, the result follows.

6 Expressive power of constraint query languages

We now return to the question of expressive power of languages such as $\mathrm{FO} + \textsc{Lin}$ or $\mathrm{FO} + \textsc{Poly}$ over constraint databases, that is, definable sets. We indicated earlier that with topological connectivity, one can express graph connectivity over finite databases, and thus the former should not be expressible. Other reductions of this kind are summarized below.

Proposition 2 (see [20]) *Assume that testing of any of the following properties of semi-linear sets in $\mathbb{R}^2$ is expressible in $\mathrm{FO} + \textsc{Lin}$ or $\mathrm{FO} + \textsc{Poly}$: topological connectivity, existence of exactly one/at least one hole, homeomorphism. Then the same language can express evenness.* $\square$

Since testing for even cardinality is generic, and not definable in $\mathrm{FO}_{\mathrm{act}}(\langle U, < \rangle, SC)$, we conclude the following from the natural generic collapse:

Corollary 2 *None of the problems mentioned in Proposition 2 is definable in $\mathrm{FO} + \textsc{Poly}$, even when inputs are restricted to semi-linear sets in $\mathbb{R}^2$.* $\square$

Most reductions to the finite case presented in [20] were quite ad-hoc. More systematic ways of getting expressivity bounds for queries over semi-algebraic sets are described in [24]. Those techniques work for *topological* queries, that is, queries invariant under homeomorphisms of input databases, which are infinite semi-algebraic sets on the real plane (cf. [28]). In such a set X, every point $\vec{x}$ has a small neighborhood $B_\epsilon(\vec{x}) = \{\vec{y} \mid d(\vec{x}, \vec{y}) \leq \epsilon\}$ whose intersection with X is conic, that is, homeomorphic to the cone with the center $\vec{x}$ and the base $S_\epsilon(\vec{x}) \cap X$, where $S_\epsilon(\vec{x}) = \{\vec{y} \mid d(\vec{x}, \vec{y}) = \epsilon\}$. For example, in the set $X = \{(x, y) \mid x^2 + y^2 \leq 1\}$, there are two types of cones. Points in the interior of X have full cones (that is, for small enough ϵ, $B_\epsilon(\vec{x}) \subseteq X$), and points on the boundary have "half-filled" cones (homeomorphic to the half-disc). Both types of cones have infinitely many points in the disk realizing them. A remarkable result of [24] says that the number of realizers of cone types is all one needs to know to find out if $\mathrm{FO} + \textsc{Poly}$ can express a given topological property.

Theorem 4 *Assume that SC consists of a single binary predicate. Let Φ be a sentence in $\mathrm{FO} + \textsc{Poly}$ which expresses a topological property (that is, a property invariant under homeomorphisms). Let X_1 and X_2 be two semi-algebraic sets in $\mathbb{R}^2$ such that for any cone type, there are equally many points in X_1 and X_2 realizing this type. Then $X_1 \models \Phi$ iff $X_2 \models \Phi$.* $\square$

We now show how to use this result to obtain an alternative proof that topological connectivity is not expressible in FO + POLY. Assume that it is expressible by a sentence Φ. Let X_1 be a disk, and X_2 be a disjoint union of two disks. Since there are only two types of cones realized in X_1 and X_2 (full cone, and half-a-disk cone), and both have continuum many realizers, Theorem 4 implies $X_1 \models \Phi$ iff $X_2 \models \Phi$. However, X_1 is connected, X_2 is not, and thus Φ cannot define topological connectivity.

Other techniques for proving expressivity bounds Complexity bounds were used to prove some early expressivity bounds for FO + LIN. In particular, it was proven that, assuming all coefficients used in linear (in)equalities are integers, the complexity of query evaluation is AC^0 [19]. This implies inexpressibility of queries such as parity of cardinality. However, the technique fails for FO + POLY, as multiplication is not in AC^0.

The following growth bounds result was established as a by-product of the work on query safety in constraint databases, and led to new expressivity bounds. Let $\varphi(\vec{x})$ be an $\mathrm{FO}_{\mathrm{act}}(\mathcal{M}, SC)$ formula. Define $growth_\varphi : \mathbb{N} \to \mathbb{N}$ by letting $growth_\varphi(n)$ be the maximum cardinality of $\{\vec{a} \mid D \models \varphi(\vec{a})\}$ as D ranges over SC-structures of size n.

Theorem 5 (see [10]) *Let $\mathcal{M}$ be o-minimal, based on a dense order. Then for any $\varphi(\vec{x})$ there exists a polynomial p_φ such that, for any $n \in \mathbb{N}$, either $growth_\varphi(n) < p_\varphi(n)$, or $growth_\varphi(n) = \infty$.*

As a geometric application, one derives from here that polygons cannot be triangulated with arbitrary precision in FO + POLY. Note that o-minimality is essential for the dichotomy theorem: over $\langle \mathbb{N}, +, <, 1 \rangle$ one can define φ with $growth_\varphi(n) = 2^n$.

Further reading For a general overview of the field of constraint databases, see the forthcoming book [25]. Chapters 1 and 2 provide an introduction to constraint databases, while chapters 3 and 4 contain an extensive treatment of expressive power. Other issues addressed in [25] include query safety (including preservation of geometric properties), dealing with spatial aggregation, datalog in constraint databases, topological queries, and semi-linear databases in spatial applications. Another interesting method of combining finite and infinite structures is the metafinite model theory of [17]; it is, however, outside of the scope of this column. For practical aspects, see a report on the implementation of a constraint database system for linear constraints [18].

References

[1] S. Abiteboul, R. Hull and V. Vianu. *Foundations of Databases.* Addison-Wesley, 1995.

[2] M. Anthony and N. Biggs. *Computational Learning Theory.* Cambridge Univ. Press, 1992.

[3] A.K. Ailamazyan, M.M. Gilula, A.P. Stolboushkin and G.F. Shvarts. Reduction of a relational model with infinite domains to the finite-domain case. *Doklady Akademii Nauk SSSR*, 286 (1) (1986), 308–311. Translation in *Soviet Physics – Doklady*, 31 (1986), 11–13.

[4] J. Baldwin and M. Benedikt. Embedded finite models, stability theory and the impact of order. In *LICS'98*, pages 490–500.

[5] S. Basu. New results on quantifier elimination over real closed fields and applications to constraint database. *J. ACM*, 46 (1999). Extended abstract in *FOCS'97*, pages 56–65.

[6] M. Benedikt, G. Dong, L. Libkin and L. Wong. Relational expressive power of constraint query languages. *Journal of the ACM* 45 (1998), 1–34. Extended abstract in *PODS'96*, pages 5–16.

[7] O. Belagradek, A. Stolboushkin, M.Tsaitlin. Extended order-generic queries. *Annals of Pure and Applied Logic*, to appear.

[8] M. Benedikt and L. Libkin. On the structure of queries in constraint query languages. *LICS'96*, pages 25–34.

[9] M. Benedikt and L. Libkin. Relational queries over interpreted structures. *Journal of the ACM*, to appear. Extended abstract in *PODS'97*, pages 87–98.

[10] M. Benedikt and L. Libkin. Safe constraint queries. *SIAM J. Comput.*, to appear. Extended abstract in *PODS'98*, pages 99–108.

[11] J. Bochnak, M. Coste, M.-F. Roy. *Real Algebraic Geometry.* Springer Verlag, 1998.

[12] L. Cabibbo and J. Van den Bussche. Converting untyped formulas to typed ones. *Acta Informatica*, 35(8) (1998), 637–643.

[13] A. Chandra and D. Harel. Computable queries for relational databases. *JCSS* 21(2):156–178, 1980.

[14] H.-D. Ebbinghaus and J. Flum. *Finite Model Theory.* Springer Verlag, 1995.

[15] R. Fagin. Probabilities on finite models. *Journal of Symbolic Logic* 41 (1976), 50–58.

[16] H. Gaifman, On local and non-local properties, *in* "Proceedings of the Herbrand Symposium, Logic Colloquium '81," North Holland, 1982.

[17] E. Grädel and Y. Gurevich. Metafinite model theory. *Information and Computation*, 140 (1998), 26–81.

[18] S. Grumbach, P. Rigaux, L. Segoufin. The DEDALE system for complex spatial queries. In *SIGMOD'98*, pages 213–224.

[19] S. Grumbach, J. Su, and C. Tollu. Linear constraint databases. In *Proceedings of Logic and Comput. Complexity, 1994*, pages 426–446.

[20] S. Grumbach and J. Su. Queries with arithmetical constraints. *Theor. Comp. Sci.* 173 (1997), 151–181.

[21] Y. Gurevich. Toward logic tailored for computational complexity. In *Computation and Proof Theory*, Springer, 1984, pages 175–216.

[22] R. Hull, J. Su. Domain independence and the relational calculus. *Acta Informatica* 31 (1994), 513–524.

[23] P. Kanellakis, G. Kuper, and P. Revesz. Constraint query languages. *JCSS*, 51 (1995), 26–52. Extended abstract in *PODS'90*, pages 299–313.

[24] B. Kuijpers, J. Paredaens and J. Van den Bussche. On topological elementary equivalence of spatial databases. In *ICDT'97*, pages 432–446.

[25] G. Kuper, L. Libkin and J. Paredaens, eds. *Constraint Databases.* Springer Verlag, 1999.

[26] M. C. Laskowski. Vapnik-Chervonenkis classes of definable sets. *J. London Math. Soc.*, 45:377–384, 1992.

[27] M. Otto and J. Van den Bussche. First-order queries on databases embedded in an infinite structure. *Information Processing Letters*, 60(1996), 37–41.

[28] C. Papadimitriou, D. Suciu and V. Vianu. Topological queries in spatial databases. In *PODS'96*, pages 81–92.

[29] J. Paredaens, J. Van den Bussche, and D. Van Gucht. Towards a theory of spatial database queries. In *PODS'94*, pages 279–288.

[30] J. Paredaens, J. Van den Bussche, and D. Van Gucht. First-order queries on finite structures over the reals. *SIAM J. Comput.* 27 (1998), 1747–1763. Extended abstract in *LICS'95*.

[31] A. Pillay, C. Steinhorn. Definable sets in ordered structures. *Bulletin of the AMS* 11 (1984), 159–162.

[32] A. Tarski. *A Decision Method for Elementary Algebra and Geometry.* 2nd ed., Univ. California Press, 1951.

[33] L. van den Dries. *Tame Topology and o-minimal Structures.* Cambridge Univ. Press, 1998.

[34] A.J. Wilkie. Model completeness results for expansions of the ordered field of real numbers by restricted Pfaffian functions and the exponential function. *J. Amer. Math. Soc.* 9 (1996), 1051–1094.