

# Datalog vs. First-Order Logic\*

Miklos Ajtai<sup>†</sup>

Yuri Gurevich<sup>‡</sup>

August 27, 2001

## Abstract

Our main result is that every datalog query expressible in first-order logic is bounded; in terms of classical model theory it is a kind of compactness theorem for finite structures. In addition, we give some counter-examples delimiting the main result.

## 1 Introduction

First-order logic and datalog are two important paradigms in the theory of relational database query languages. How different are they from the point of view of expressive power? What can be expressed both in first-order logic and datalog? Let us make this latter question precise.

An  $r$ -ary *global relation*  $\mathcal{R}$  of signature  $\sigma$  is a function that, given a  $\sigma$ -structure  $A$ , produces an  $r$ -ary relation  $\mathcal{R}_A$  on  $A$  [Gu].  $\mathcal{R}$  is *abstract* if, for every isomorphism  $f$  from a  $\sigma$ -structure  $A$  to a  $\sigma$ -structure  $B$  and all elements  $a_1, \dots, a_r$  in  $A$ ,  $\mathcal{R}_A(a_1, \dots, a_r)$  holds in  $A$  if and only if  $\mathcal{R}_B(fa_1, \dots, fa_r)$  holds in  $B$ . Abstract global relations are often called queries. We reserve the term “query” to denote datalog queries as syntactical objects. By the way, we do not presuppose any familiarity with datalog. The necessary definitions are given in Section 2.

Here and everywhere else in this paper, a signature is a finite collection of predicates (i.e. relation symbols) and individual constants; no function symbols of positive arity are allowed. The term “formula” is restricted to denote first-order formulas with equality. As usual, the equality sign is a logical constant; it does not appear in signatures and is interpreted as the identity in every structure.

A formula  $\varphi(v_1, \dots, v_r)$  of signature  $\sigma$  with free individual variables  $v_1, \dots, v_r$  in the lexicographical order expresses and means the  $r$ -ary global relation of signature  $\sigma$  that, given a  $\sigma$ -structure  $A$ , produces the relation  $\{(a_1, \dots, a_r) : A \models \varphi(a_1, \dots, a_r)\}$  on  $A$ . If  $\Pi$  is a datalog program and  $Q$  is an  $r$ -ary intentional predicate of  $\Pi$  then the  $r$ -ary (datalog) query  $(\Pi, Q)$  expresses and means the intended value of  $Q$  on databases for  $\Pi$ .

By default our structures are finite. Respectively, a global relation  $\mathcal{R}$  is considered to be first-order expressible (resp. datalog expressible) if there exists a formula (resp. a query) that expresses  $\mathcal{R}$  on finite structures.

**Question:** Which global relations are expressible both in first-order logic and datalog?

---

\*J of Computer and System Sciences, Vol 49, no 3, Dec 1994, 562–588.

<sup>†</sup>IBM Almaden Research Center, San Jose, CA 95120.

<sup>‡</sup>Supported by NSF grants CCR 89-04728, CCR 92-04742 and by ONR grant N00014-91-J-11861. Electrical Engineering and Computer Science Department, University of Michigan, Ann Arbor, MI 48109-2122. The main part of this work was done during 1988-89 academic year when this author was with Stanford University and IBM Almaden Research Center (on a sabbatical leave from the University of Michigan).

It is not difficult to check that bounded queries (see the definition in Section 2) are first-order expressible. Cosmadakis conjectured [Co] that every first-order expressible query is bounded and confirmed the conjecture in a number of important special cases. We prove the conjecture of Cosmadakis; this is our main result. Thus, a query is first-order expressible if and only if it is bounded. It is easy to transform every bounded query to a query with no intentional predicates in the body of any rule (a non-recursive query) in such a way that the two queries are equivalent, that is, express the same global relation on finite structures. It is easy to check that each non-recursive query is equivalent to a positive existential formula (the definitions of positivity and existentiality are recalled in Section 3), and the other way round.

**Theorem 1.1** *If a query  $Q$  and a formula  $\varphi$  express the same global relation on finite structures then  $Q$  is bounded and  $\varphi$  is equivalent to a positive existential formula.*

Since recursion is the strength of datalog, bounded queries are often viewed to be trivial. In that sense, first-order logic and datalog are almost disjoint.

If infinite structures are allowed, Theorem 1.1 can be established by a straightforward compactness argument; see Section 3. This should not be surprising. In the presence of infinite structures, the expressibility condition is stronger whereas every query bounded on finite structures is bounded on infinite ones as well [Section 2]. Of course the proof using a compactness argument does not survive the restriction to finite structures. As a rule, theorems whose proofs rely heavily on a compactness argument do not survive the restriction to finite structures [Gu]. Theorem 1.1 seems to be the first non-trivial exception. In Section 4 it is reformulated in terms of classical model theory.

The proof of Theorem 1.1 occupies Sections 3–9. Recall that a sentence is a formula without free individual variables and that a boolean query is a query of arity zero. In Section 3 we verify that the following four assertions are equivalent:

**B** Every first-order expressible query is bounded.

**B0** Every first-order expressible boolean query is bounded.

**E** Every datalog expressible formula is equivalent to a positive existential one.

**E0** Every datalog expressible sentence is equivalent to a positive existential one.

In Section 4 we formulate a kind of compactness assertion **C** (for finite structures) and prove that **C** is equivalent to **E0**. In Section 5 we prove that **B0**, **E0** and **C** are equivalent to their versions in the case when there are no individual constants.

In Section 6 we define, for each natural number  $s$ , the notion of  $s$ -wide class of structures. In Section 7, we prove that, for no sentence  $\varphi$  and no  $s$ , the class of models of  $\varphi$  is  $s$ -wide. Then, in Section 9, we prove that, for each unbounded boolean query  $Q$ , the class of models of  $Q$  is  $s$ -wide for some  $s$ . This completes the proof of Theorem 1.1. The Section 8 is auxiliary (and probably the most interesting mathematically speaking).

In Section 10 we construct counter-examples to the generalizations of Theorem 1.1 in the case when negations or only inequalities are allowed in the bodies of datalog rules. In particular, there exists an unbounded boolean query with inequalities but without individual constants that is first-order expressible. In Section 11, we construct a counter-example to the generalization of Theorem 1.1 in the case when the notion of first-order expressibility is relaxed to implicit first-order expressibility.

The reference [AG] is an extended abstract of this paper.

**Acknowledgment** Phokion Kolaitis and Moshe Vardi drew our attention to the problem of first-order expressibility of datalog queries and shared with us their knowledge of the subject, Paris Kanellakis prompted us to clarify the case of datalog with inequalities but without individual constants, and Frank Messerle found two real mistakes in a previous version. We are thankful to all these people.

## 2 Datalog

In this section, we explain what datalog is and establish terminology. An *atomic formula* is an equality  $e_1 = e_2$  or a *proper atomic formula*  $P(e_1, \dots, e_r)$  where  $P$  is an  $r$ -ary predicate different from equality; here each  $e_p$  is an individual variable or individual constant. A *datalog rule* is an expression of the form

$$\alpha \leftarrow \beta_1, \dots, \beta_k$$

where  $k$  is a natural number (possibly zero),  $\alpha$  is a proper atomic formula and each other  $\beta_i$  is an atomic formula. The atomic formula  $\alpha$  is the *head* of the rule, and the sequence  $\beta_1, \dots, \beta_k$  is the *body*. A *datalog program* is a finite set of datalog rules. In the rest of the paper, the terms “rule” and “program” refer to datalog rules and datalog programs respectively.

Here is an example program  $\Pi_0$ :

$$\begin{aligned} xTy &\leftarrow xEy \\ xTy &\leftarrow xTz, zEy \\ Q &\leftarrow c_1Tc_2 \end{aligned}$$

The head predicates of a program  $\Pi$  are *intentional*; the other predicates are *extensional*. The extensional predicates and the individual constants form the *extensional signature*  $\text{Sig}_e(\Pi)$  of  $\Pi$ . Any structure  $D$  of signature  $\text{Sig}_e(\Pi)$  is a *database* for  $\Pi$ . In the case of the example program  $\Pi_0$ , the extensional signature comprises the binary predicate  $E$  and the individual constants  $c_1, c_2$ . A database for  $\Pi_0$  is a directed graph with two distinguished nodes.  $\Pi_0$  has two intentional predicates, namely, the binary predicate  $T$  and the zero-ary predicate  $Q$ .

By analogy with directed graphs, an *edge* of an arbitrary structure  $D$  is a true statement of the form  $a_1 = a_2$  or  $P(a_1, \dots, a_r)$  where  $P$  is an  $r$ -ary predicate in the signature of  $D$  and each  $a_i$  is an element of  $D$ .

Given a database  $D$ , a program  $\Pi$  computes the intended values of its intentional predicates. To define the intended values, interpret the pair  $(\Pi, D)$  as a logical calculus with edges of  $D$  as axioms and rules of  $\Pi$  as inference rules. Objects derivable in the calculus  $(\Pi, D)$  have the form  $Q(\bar{a})$  where  $Q$  is an intentional predicate and  $\bar{a}$  is a tuple of elements in  $D$  of the appropriate length. If  $(\Pi, D)$  derives  $Q(\bar{a})$ , then  $Q(\bar{a})$  is a *link* of  $D$  (with respect to  $\Pi$ ). The intended value  $Q_D^*$  of an intentional predicate  $Q$  on  $D$  is the set of all  $Q$ -links of  $D$ . (In order for this to make sense in the case of zero-ary predicates, we suppose that the empty set represents falsity and the singleton set whose only element is the empty tuple represents truth.)

It is convenient to view the intended values of intentional predicates as the result of some evolution. For each intentional predicate  $Q$  and every natural number  $t$ , let  $Q_D^t$  be the set of  $Q$ -links derivable in  $\leq t$  steps. Thus,  $Q_D^0$  is empty and  $Q_D^* = \bigcup_t Q_D^t$ . The least  $t$  such that  $Q_D^t = Q_D^*$  is the *evolution time* of  $Q$  over  $D$ . Let  $D^t$  be the enrichment of  $D$  with all the relations  $Q_D^t$  and  $D^*$  be the enrichment of  $D$  with all the relations  $Q_D^*$ . Structures  $D^0, D^1, \dots, D^*$  are stages of the evolution in question.

Consider the example program  $\Pi_0$  and a digraph  $G$  with two distinguished vertices. It is easy to see that, for each  $k > 0$ ,  $T^k$  comprises pairs  $(a, b)$  of vertices of  $G$  such that  $G$  has a path of length at most  $k$  from  $a$  to  $b$ . The intended value of  $T$  is the transitive closure of the edge relation  $E$ , the evolution time of  $T$  is the diameter of  $G$ , the intended value of  $Q$  is the truth value of the statement “There is a path from  $c_1$  to  $c_2$ ”, and the evolution time of  $Q$  is bounded by 1 plus the diameter of  $G$ . Notice that replacing  $zEy$  with  $zTy$  in the body of the second rule does not change the intended values of intentional predicates but speeds up the evolution exponentially.

A *(datalog) query*  $Q$  is a pair  $(\Pi, Q)$  where  $\Pi$  is a program and  $Q$  is an intentional predicate of  $\Pi$ . The arity of predicate  $Q$  is the *arity* of the query  $Q$ ;  $Q$  is *boolean* if the arity is zero. The meaning of  $Q$  is the global relation of signature  $\text{Sig}_e(\Pi)$  that, given a database  $D$  for  $\Pi$ , produces the intended value  $Q_D^*$  of  $Q$  on  $D$  with respect to  $\Pi$ .

For brevity only, we define the *evaluation time* of a query  $(\Pi, Q)$  over a database  $D$  to be the evolution time of  $Q$  over  $D$ .

A query is *non-recursive* if no intentional predicate appears in the body of any rule. A query is *bounded* if there is a number  $b$  such that the evaluation time of the query over any database is bounded by  $b$ . Every non-recursive query is bounded (with  $b = 1$ ).

A structure  $B$  is a *substructure* of a structure  $A$  of the same signature if every element of  $B$  is an element of  $A$  and every edge of  $B$  is an edge of  $A$ . A substructure  $B$  is *induced* if every edge of  $A$  on elements of  $B$  is an edge of  $B$ .

**Lemma 2.1** *A query  $(\Pi, Q)$  is bounded if and only if there exists a positive integer  $n$  such that every  $Q$ -link of an arbitrary database  $A$  for  $\Pi$  is also a link of a substructure  $B$  of  $A$  with at most  $n$  elements.*

**Proof** To establish the only-if implication, notice that derivations of depth bounded by a fixed number involve only so many axioms and therefore only so many elements. To establish the if implication, notice that databases of size bounded by a fixed number  $n$  have only so many links; this gives a bound on evolution time.  $\square$

The lemma and the proof remain valid in the case when infinite structures are allowed. It follows that every query bounded on finite structures is bounded on infinite structures as well.

### 3 Reduction to boolean queries

In this section we prove that the assertions **B**, **B0**, **E**, **E0** defined in the Introduction are indeed equivalent.

The definition of positive formulas can be found in logic textbooks. For our purposes, the following simplified definition will do. *Positive formulas* are built from atomic formulas and propositional constants *true*, *false* by means of conjunctions, disjunctions, existential quantifiers and universal quantifiers. A formula is *existential* if it has the form  $(\exists u_1 \dots \exists u_k)\Phi$  where  $\Phi$  is quantifier-free. Universal formulas are defined similarly.

**Lemma 3.1** *Every positive existential formula is equivalent to a non-recursive query.*

**Proof** Without loss of generality, the quantifier-free part of the given positive existential formula  $\varphi(v_1, \dots, v_r)$  is a disjunction where the  $i$ -th disjunct is the conjunction of some list  $L_i$  of atomic formulas. It is easy to see that  $\varphi$  is equivalent to the non-recursive query that consists of a new  $r$ -ary predicate  $Q$  and the program with rules  $Q(\vec{v}) \leftarrow L_i$ .  $\square$

**Lemma 3.2** *Every bounded query is equivalent to a positive existential formula.*

**Proof** The desired formula uses  $\leq n$  existential quantifiers where  $n$  is the smallest number such that the bounded-ness criterion of Lemma 2.1 is satisfied.  $\square$

By the way, it may be necessary to use the equality in the formula even if the given query  $(\Pi, Q)$  does not use it [Me]. For example,  $\Pi$  may have a single rule  $Q(v_1, v_1) \leftarrow \cdot$ . Then the desired formula is  $v_1 = v_2$ .

Notice that the bounded-ness criterion is purely semantical. It follows that every query equivalent to a bounded one is bounded itself. This is an interesting peculiarity of datalog. There is no semantical characterization of positive existential formulas because any such formula is equivalent to a formula that is neither positive nor existential.

**Theorem 3.1** *The assertions **B**, **B0**, **E** and **E0** are equivalent.*

**Proof** By Lemma 3.2, **B** implies **E**, and **B0** implies **E0**.

To prove that **E** implies **B**, assume **E** and let  $Q$  be a query that is first-order expressible. By **E**,  $Q$  is equivalent to a positive existential formula. By Lemma 3.1,  $Q$  is equivalent to a non-recursive query. By Lemma 2.1,  $Q$  is bounded. The same proof establishes that **E0** implies **B0**.

Obviously, **B** implies **B0**, and **E** implies **E0**.

To prove that **B0** implies **B**, assume **B0** and let  $(\Pi, Q)$  be a query expressible by a formula  $\varphi(v_1, \dots, v_r)$ . Define  $\Pi'$  to be the extension of  $\Pi$  by an additional rule  $R \leftarrow Q(c_1, \dots, c_r)$  where  $R$  is a new zero-ary predicate and  $c_1, \dots, c_r$  are new individual constants. Clearly, the sentence  $\varphi(c_1, \dots, c_r)$  is equivalent to the boolean query  $(\Pi', R)$ . By **B0**, there exists a bound  $k + 1$  on the evaluation time of  $(\Pi', R)$  over any database. We claim that  $k$  is a bound on the evaluation time of  $(\Pi, Q)$  over any database. By contradiction, suppose that the evaluation time of  $(\Pi, Q)$  on some database  $D$  exceeds  $k$ . Pick a link  $Q(a_1, \dots, a_r)$  in  $Q_D^* - Q_D^k$  and interpret individual constants  $c_1, \dots, c_r$  as elements  $a_1, \dots, a_r$  respectively. The evaluation time of  $R$  over the resulting extension of  $D$  exceeds  $k + 1$ , which gives the desired contradiction.  $\square$

For future use, we notice the following corollary of the proof.

**Corollary 3.1** *In the case of logic without individual symbols, **B** is equivalent to **E**, and **B0** is equivalent to **E0**.*

The following claim is a side remark; it is not needed for the proof of Theorem 1.1.

**Claim 3.1** ***E0** holds if infinite structures are allowed.*

**Proof** Suppose that a sentence  $\varphi$  is equivalent to a boolean query  $(\Pi, Q)$ . For each  $k$ , there exists a positive existential sentence  $\psi_k$  that is true on a database  $D$  if and only if  $Q$  is derivable in  $(\Pi, D)$  it at most  $k$  steps. Obviously, every  $\psi_k$  implies  $\varphi$ . It suffices to prove that  $\varphi$  implies some  $\psi_k$ .

By contradiction, suppose that  $\varphi$  does not imply any  $\psi_k$ . Then every pair  $\{\varphi, \neg\psi_k\}$  is satisfiable. Clearly, every finite subset of the infinite collection  $\mathcal{C} = \{\varphi, \neg\psi_1, \neg\psi_2, \dots\}$  is satisfiable. By the compactness theorem,  $\mathcal{C}$  itself is satisfiable. Let  $A$  be a model of  $\mathcal{C}$ . Since  $A \models \varphi$ ,  $Q$  is a link of  $A$  and therefore some  $\psi_k$  holds in  $A$  which is impossible.  $\square$

## 4 Finite Compactness

In this section, we show that boolean queries are equivalent to special second-order sentences. Then we reformulate our main result in terms of traditional model theory as Finite Compactness Theorem.

We recall (variations of) some well known definitions. A *Horn clause* is a formula of the form  $\beta \rightarrow \alpha$  where the antecedent  $\beta$  is a conjunction of some number  $j$  of atomic formulas and the succedent  $\alpha$  is either a proper atomic formula or the logical constant *false*. If the succedent is atomic, we call the clause *imperative*; otherwise we call it *declarative*. The number  $j$  can be zero in which case  $\beta$  is the logical constant *true*. A *Horn formula* is a conjunction of clauses. It is common to represent Horn formulas as sets of clauses. The predicates that appear in the succedents of imperative clauses are *intentional*; the other predicates are *extensional*. The extensional predicates and the individual constants form the *extensional signature*  $\text{Sig}_e(\eta)$  of a Horn formula  $\eta$ .

If  $\eta$  is a Horn formula then  $\text{Pr}(\eta)$  is the class of all structures  $A$  of signature  $\text{Sig}_e(\eta)$  such that  $A$  together with some values of the intentional predicates universally satisfies  $\eta$  (that is,  $\eta$  is satisfied for all values of individual variables). A class  $K$  of structures is *projective* if there exists a Horn formula  $\eta$  such that  $K = \text{Pr}(\eta)$ . For example, the class of acyclic digraphs is projective; the projectivity witness is

$$xEy \rightarrow xTy, \quad (xTy \wedge yTz) \rightarrow xTz, \quad xTx \rightarrow \text{false}.$$

Call a boolean query  $(\Pi, Q)$  *proper* if  $Q$  does not occur in the body of any rule of  $\Pi$ . Every boolean query  $(\Pi, Q)$  is equivalent to a proper one. Just remove all rules where  $Q$  occurs in the body; it is clear that this does not change the meaning of the query.

There is a close connection between proper boolean queries and Horn formulas. To transform a given proper boolean query  $\mathcal{Q} = (\Pi, Q)$  into a Horn formula,

- replace every rule  $Q \leftarrow \alpha_1, \dots, \alpha_j$  with the declarative clause

$$(\alpha_1 \wedge \dots \wedge \alpha_j) \longrightarrow \text{false}$$

- and replace every other rule  $\alpha_0 \leftarrow \alpha_1, \dots, \alpha_j$  with the imperative clause

$$(\alpha_1 \wedge \dots \wedge \alpha_j) \longrightarrow \alpha_0.$$

Call the resulting formula  $H(\mathcal{Q})$ .

A database  $D$  for a boolean query  $\mathcal{Q} = (\Pi, Q)$  is a *model* for  $\mathcal{Q}$  if  $D$  generates  $Q$ .  $\text{Mod}(\mathcal{Q})$  is the class of models of  $\mathcal{Q}$ .

**Lemma 4.1** *Let  $\eta = H(\mathcal{Q})$ . Then  $\text{Pr}(\eta)$  is the complement of  $\text{Mod}(\mathcal{Q})$  in the class of databases for  $\mathcal{Q}$ . In other words, for arbitrary database  $D$  for  $\mathcal{Q}$ , the following statements are equivalent:*

1.  $D \in \text{Mod}(\mathcal{Q})$ .
2. For all interpretations of the intentional predicates of  $\eta$  in  $D$  such that all imperative clauses of  $\eta$  are universally true, some declarative clause fails, that is the body of that declarative clause is satisfied.

**Proof** To simplify the exposition, we assume that  $P$  is the only intentional predicate of  $\eta$ . Let  $D$  be a database for  $\mathcal{Q} = (\Pi, Q)$  and  $P^*$  be the intended value of  $P$  with respect to the program of  $\mathcal{Q}$ .

(1) $\rightarrow$ (2). Assume (1) and let  $P'$  be an arbitrary interpretation of  $P$  such that all imperative clauses of  $\eta$  are satisfied. Clearly,  $P'$  includes  $P^*$ . By (1), some  $Q$ -rule fires in  $D$ . Hence the corresponding declarative clause fails in  $(D, P^*)$  and therefore in  $(D, P')$ .

(2) $\rightarrow$ (1). Assume (2) and choose  $P' = P^*$ . Some declarative clause of  $\eta$  fails in  $(D, P^*)$  and therefore the corresponding  $Q$ -rule fires in  $D$ .  $\square$

We say that a class  $K$  of structures of some signature  $\sigma$  is *compact* if there exists  $n$  such that an arbitrary  $\sigma$ -structure  $A$  belongs to  $K$  if and only if all substructures of  $A$  of cardinality at most  $n$  belong to  $K$ .

**Lemma 4.2**  *$K$  is compact if and only if its complement is axiomatizable by means of a positive existential sentence.*

**Proof** Clear.  $\square$

Restrict the term “axiomatizable” to mean finitely axiomatizable.

**Theorem 4.1** *The following two assertions are equivalent:*

**E0** *Every datalog expressible sentence is equivalent to a positive existential one.*

**C** *Every axiomatizable projective class is compact.*

In particular, **C** implies that acyclicity is not expressible in the first-order language of digraphs (a known fact). To prove this, it suffices to check that the class of acyclic digraphs is not compact. To show that a given  $n$  is not a compactness witness, consider a cycle of length  $n + 1$ .

**Proof** First we assume **E0** and prove **C**. Let  $K$  be an axiomatizable, projective class of  $\sigma$ -structures. There exist a sentence  $\varphi$  and a Horn formula  $\eta$  such that  $K = \text{Mod}(\varphi) = \text{Pr}(\eta)$ . Construct a boolean query  $\mathcal{Q}$  such that  $\eta = H(\mathcal{Q})$ . By Lemma 4.1,  $\text{Mod}(\mathcal{Q})$  is the complement of  $\text{Pr}(\eta)$  and therefore  $\text{Mod}(\mathcal{Q}) = \text{Mod}(\neg\varphi)$ . By **E0**,  $\neg\varphi$  is equivalent to a positive existential sentence. Now use Lemma 4.2.

Next we assume **C** and prove **E0**. Let a sentence  $\varphi$  be equivalent to a boolean query  $\mathcal{Q} = (\Pi, Q)$ . This means that the formula and the query denote the same global relation  $\mathcal{R}$  of some signature  $\sigma$  where  $\sigma$  is the signature of  $\varphi$  and the extensional signature of  $\mathcal{Q}$ . Let  $K$  be the class of  $\sigma$ -structures  $D$  where  $\mathcal{R}_D$  is false. By Lemma 4.1,  $K$  is projective. By **C**,  $K$  is compact. Now use Lemma 4.2.  $\square$

For future use, we notice the following corollary of the proof.

**Corollary 4.1** *In the logic without individual constants, **E0** is equivalent to **C**.*

## 5 Removing individual constant

Call a formula or query or signature *plebeian* if it does not have individual constants. In this section, we reduce assertion **B0** to its restriction **B0P** to plebeian formulas.

**Theorem 5.1** *The following assertions are equivalent:*

**B0** *Every first-order expressible boolean query is bounded.*

**B0P** Every first-order expressible plebeian boolean query is bounded.

**E0** Every datalog expressible sentence is equivalent to a positive existential sentence.

**E0P** Every datalog expressible plebeian sentence is equivalent to a plebeian positive existential sentence.

**C** Every axiomatizable projective class is compact.

**CP** Every axiomatizable projective class of plebeian signature is compact.

**Proof** Obviously, **B0** implies **B0P**, and **E0** implies **E0P**, and **C** implies **CP**. By Theorem 3.1, **B0** is equivalent to **E0**; by Corollary 3.1, **B0P** is equivalent to **E0P**. By Theorem 4.1, **E0** is equivalent to **C**; by Corollary 4.1, **E0P** is equivalent to **CP**. It suffices to prove that **CP** implies **C**.

Let  $K$  be an axiomatizable projective class of structures of some signature  $\sigma$ . We need to prove that, for some  $n$ , an arbitrary  $\sigma$ -structure  $A$  belongs to  $K$  if and only all substructures of  $A$  of cardinality  $\leq n$  belong to  $K$ . To simplify the exposition, we suppose that  $\sigma = \{P, c, d\}$  where  $P$  is a binary predicate and  $c, d$  are individual constants. Let  $K_1$  (resp.  $K_2$ ) be the collection of structures from  $K$  where  $c = d$  (resp.  $c \neq d$ ). Obviously,  $K_2$  is axiomatizable and projective. View  $K_1$  as a class of structures of signature  $\{P, c\}$ ; it is also axiomatizable and projective. It suffices to prove that  $K_1$  and  $K_2$  are compact. We restrict attention to  $K_2$ . In the rest of the proof,  $K = K_2$ .

Call an element of a given  $\sigma$  structure  $A$  *plebeian* if it is not distinguished, i.e., if it isn't the interpretation of some individual constant. Call  $A$  *trivial* if it has no plebeian elements. If  $A$  is not trivial, let  $A^p$  be the induced substructure of  $A$  that contains all and only plebeian elements (the *plebeian substructure*).

Remove individual constants from  $\sigma$  and then add unary predicates  $P_{c*}, P_{d*}, P_{*c}, P_{*d}$  and zero-ary predicates  $P_{cc}, P_{cd}, P_{dc}, P_{dd}$ ; call the resulting signature  $\sigma'$ . For each non-trivial  $\sigma$ -structure  $A$ , enrich  $A^p$  with values of the new predicates in such a way that the following axioms are satisfied when  $u$  ranges over  $A^p$ .

$$P_{c*}(u) \leftrightarrow P(c, u) \tag{1}$$

$$P_{*c}(u) \leftrightarrow P(u, c) \tag{2}$$

$$P_{d*}(u) \leftrightarrow P(d, u) \tag{3}$$

$$P_{*d}(u) \leftrightarrow P(u, d) \tag{4}$$

$$P_{cc} \leftrightarrow P(c, c) \tag{5}$$

$$P_{cd} \leftrightarrow P(c, d) \tag{6}$$

$$P_{dc} \leftrightarrow P(d, c) \tag{7}$$

$$P_{dd} \leftrightarrow P(d, d) \tag{8}$$

This turns  $A^p$  into a  $\sigma'$  structure which will be called  $A'$ . Let  $K' = \{A' : A \in K\}$ .

**Lemma 5.1**  $K'$  is projective.

**Proof** Let a Horn formula  $\eta$  witness that  $K$  is projective. To simplify the exposition, we suppose that  $\eta$  is

$$Q(u, v) \leftarrow [P(c, u) \wedge Q(v, c)].$$



The desired projectivity witness  $\eta'$  for  $K'$  is the conjunction of the following 9 clauses where  $Q_{c*}, Q_{d*}, Q_{*c}, Q_{*d}$  are new unary predicates and  $Q_{cc}, Q_{cd}, Q_{dc}, Q_{dd}$  are new zero-ary predicates. The idea is to restrict variables to plebeian elements.

$$Q_{cc} \leftarrow [P_{cc} \wedge Q_{cc}] \quad (9)$$

$$Q_{cd} \leftarrow [P_{cc} \wedge Q_{dc}] \quad (10)$$

$$Q_{dc} \leftarrow [P_{cd} \wedge Q_{cc}] \quad (11)$$

$$Q_{dd} \leftarrow [P_{cd} \wedge Q_{dc}] \quad (12)$$

$$Q_{c*}(v) \leftarrow [P_{cc} \wedge Q_{*c}(v)] \quad (13)$$

$$Q_{d*}(v) \leftarrow [P_{cd} \wedge Q_{*c}(v)] \quad (14)$$

$$Q_{*c}(u) \leftarrow [P_{c*}(u) \wedge Q_{cc}] \quad (15)$$

$$Q_{*d}(u) \leftarrow [P_{c*}(u) \wedge Q_{dc}] \quad (16)$$

$$Q(u, v) \leftarrow [P_{c*}(u) \wedge Q_{*c}(v)] \quad (17)$$

First we check that every structure in  $K'$  with some values of  $Q$  and its relatives satisfies  $\eta'$ . Let  $A \in K$  and fix an interpretation of  $Q$  such that the corresponding enrichment of  $A$  universally satisfies  $\eta$ . Interpret  $Q_{c*}$  and other new predicates on the plebeian elements  $x$  of  $A$  in the obvious way, e.g., interpret  $Q_{c*}$  as  $\{x : A \models Q(c, x)\}$ . It is easy to see that the corresponding enrichment of  $A'$  universally satisfies  $\eta'$ .

Now suppose that  $B$  is a  $\sigma'$ -structure and some enrichment  $B^*$  of  $B$  with interpretations of  $Q$  and its relatives universally satisfies  $\eta'$ . There exists a  $\sigma$ -structure  $A$  such that  $B = A'$ . To obtain the desired  $A$ , add to  $B$  two new elements interpreting  $c$  and  $d$  respectively, then extend  $P$  with respect to equalities 1–8 and forget the other predicates. Extend the interpretation of  $Q$  to the new universe such that it equals the union of the following sets:

$$\begin{aligned} &\{(x, y) : B^* \models Q(x, y)\}, \\ &\{(c, x) : B^* \models Q_{c*}(x)\}, \\ &\{(d, x) : B^* \models Q_{d*}(x)\}, \\ &\{(x, c) : B^* \models Q_{*c}(x)\}, \\ &\{(x, d) : B^* \models Q_{*d}(x)\}, \\ &\{(c, d) : B^* \models Q_{cd}\} \end{aligned}$$

The respective enrichment of  $A$  satisfies  $\eta$ , so that  $A \in K$  and therefore  $B = A' \in K'$ .  $\square$

**Lemma 5.2**  $K'$  is axiomatizable.

**Proof** By induction on a  $\sigma$ -formula  $\psi$ , we define a  $\sigma'$ -formula  $\psi'$  with the same individual variables, the *plebeian companion* of  $\psi$ . The intention is that  $\psi'$  translates  $\psi$  but speaks about plebeian elements only.

- In the case when  $\psi$  is atomic and contains an individual constant, we are guided by the equivalences 1–8. For example, if  $\psi = P(c, v)$  then  $\psi' = P_{c*}(v)$ .
- In the case when  $\psi$  is an atomic formula without individual constants,  $\psi' = \psi$ .
- $(\psi_1 \wedge \psi_2)' = (\psi_1' \wedge \psi_2')$ ,  $(\psi_1 \vee \psi_2)' = (\psi_1' \vee \psi_2')$ ,  $(\neg\psi)' = \neg(\psi')$ .

•

$$\begin{aligned} ((\forall v)(\psi(v)))' &= [\psi'(c) \wedge \psi'(d) \wedge (\forall v)(\psi(v)')] \\ ((\exists v)(\psi(v)))' &= [\psi'(c) \vee \psi'(d) \vee (\exists v)(\psi(v)')] \end{aligned}$$

It is easy to see that, for every  $\sigma$ -formula  $\psi(v_1, \dots, v_k)$ , every non-trivial  $\sigma$ -structure  $A$  and every tuple  $a_1, \dots, a_k$  of plebeian elements of  $A$ ,

$$A \models \varphi(a_1, \dots, a_k) \iff A' \models \varphi'(a_1, \dots, a_k).$$

It follows that if  $\varphi$  axiomatizes  $K$  then  $\varphi'$  axiomatizes  $K'$ .  $\square$

We are ready now to finish the proof of Theorem 5.1. By **CP** and Lemmas 5.1–5.2, there exists  $n$  such that an arbitrary  $\sigma'$ -structure belongs to  $K'$  if and only if every substructure of it of cardinality  $\leq n$  belong to  $K'$ . We check that  $n+2$  is a compactness witness for  $K$ .

Since  $K$  is projective, it is closed under substructures. For, let  $\eta$  be a projectivity witness,  $A \in K$  and  $B$  is a substructure of  $A$ . There are values of intentional predicates such that the enrichment of  $A$  universally satisfies  $\eta$ . Restrict those values to  $B$ . It is easy to see that this enrichment of  $B$  universally satisfies every clause of  $\eta$ .

Now, suppose that all substructures of cardinality  $\leq n+2$  of a non-trivial  $\sigma$ -structure  $A$  belong to  $K$ . (The case of trivial  $A$  is obvious.) It follows that all substructures of cardinality  $\leq n$  of  $A'$  belong to  $K'$ . Hence  $A' \in K'$  and therefore  $A \in K$ .  $\square$

## 6 The proof of the main theorem

Theorem 3.1 reduces Theorem 1.1 to assertion **B0**. Theorem 5.1 reduces **B0** to assertion **B0P**. Thus it suffices to prove **B0P**. In this section we reduce **B0P** to two theorems. One of them will be proved in Section 7 and the other in Section 9.

A mapping  $h$  from a structure  $A$  into a structure  $B$  of the same signature is a *homomorphism* if the  $h$ -image of every edge of  $A$  is an edge of  $B$ , i.e.,  $B \models P(h(a_1), \dots, h(a_r))$  whenever  $A \models P(a_1, \dots, a_r)$ . (Notice that we do not require, as it is often done, that  $h$  is onto.) A sentence  $\varphi$  is *preserved under homomorphisms* if, for structures  $A, B$  of the signature of  $\varphi$  and for every homomorphism  $h$  from  $A$  to  $B$ ,  $A \models \varphi$  implies  $B \models \varphi$ .

**Lemma 6.1** *For every boolean query  $\mathcal{Q}$ , the class  $\text{Mod}(\mathcal{Q})$  is closed under homomorphisms.*

**Proof** Suppose that  $A$  satisfies  $\mathcal{Q}$ ,  $h$  is a homomorphism from  $A$  into  $B$ , and  $\eta = H(\mathcal{Q})$ . We check that  $B$  satisfies condition (2) of Lemma 4.1. To simplify the exposition, we suppose that a binary predicate  $P$  is the only intentional predicate in  $\eta$ . Given an arbitrary value  $P'$  of  $P$  which makes all imperative clauses of  $\eta$  universally true in  $B$ , define relation

$$P'' = \{(a_1, a_2) : (ha_1, ha_2) \in P'\}$$

on  $A$ . It is easy to see that  $P''$  makes all imperative clauses of  $\eta$  universally true in  $A$ . By Lemma 4.1, some declarative clause of  $\eta$  fails in  $(A, P'')$ . It is easy to see that the same clause fails in  $(B, P')$ .  $\square$

A member  $A$  of a class  $K$  of structures is *minimal* if no proper substructure of  $A$  satisfies  $\varphi$ . (A substructure of  $A$  is *proper* if it differs from  $A$ .)

The *Gaifman graph*  $G(A)$  for a structure  $A$  is the graph  $(|A|, E)$  where  $|A|$  is the universe of  $A$  and  $E$  comprises pairs  $\{a_1, a_2\}$  such that some edge of  $A$  involves both  $a_1$  and  $a_2$ . The *distance* between elements  $a_1$  and  $a_2$  in  $A$  is the distance between  $a_1$  and  $a_2$  in  $G(A)$ , i.e., the number of edges in the shortest path connecting the two vertices; the distance is  $\infty$  if there is no path connecting the two vertices. A subset  $S$  of a structure  $A$  is *d-scattered* if the distance between any two elements of  $S$  exceeds  $d$ .

**Definition 6.1** *A class  $K$  of structures is  $s$ -wide if, for all positive integers  $m, d$ , there exist a minimal  $A \in K$  and an induced subgraph  $H$  of  $G(A)$  such that  $\|A\| - \|H\| \leq s$  and  $H$  has a  $d$ -scattered subset of cardinality  $m$ .  $K$  is  $s$ -narrow if it is not  $s$ -wide.*

The condition that  $H$  has a  $d$ -scattered subset of cardinality  $m$  is not necessarily equivalent to the condition that the corresponding induced substructure  $B$  of  $A$  has a  $d$ -scattered subset of cardinality  $m$ . To show this, suppose that  $A$  has an edge  $P(a, b_1, b_2)$  where  $a \in A - B$  and  $b_1, b_2 \in B$ . Then  $b_1, b_2$  are adjacent in  $G(A)$  and therefore in  $H$ , but they are not necessarily adjacent in  $B$ .

**Theorem 6.1** *If a plebeian first-order sentence  $\varphi$  is preserved under homomorphisms then  $\text{Mod}(\varphi)$  is  $s$ -narrow for all  $s$ .*

Theorem 6.1 will be proved in Section 7.

**Theorem 6.2** *If a plebeian boolean query  $\mathcal{Q}$  is unbounded then  $\text{Mod}(\mathcal{Q})$  is  $s$ -wide for some  $s$ .*

Theorem 6.2 will be proved in Sections 9.

**Corollary 6.1** *The assertion **B0P** is true. In other words, if a boolean query without constants is first-order expressible then it is bounded.*

## 7 Local Properties

We prove Theorem 6.1. It can be generalized to the case with equality but we do not need the generalization.

The *vicinity*  $V_A^r(a)$  (or simply  $V^r(a)$ ) of radius  $r$  of an element  $a$  in a structure  $A$  is the induced substructure of  $A$  containing elements  $b$  with the distance  $\delta(a, b) \leq r$ . Given positive integers  $r, n$  and a formula  $\psi(v)$  in the language of  $A$ , it is easy to write a sentence  $\varphi$  in the same language asserting that there is a  $2r$ -scattered subset  $S$  of cardinality  $n$  such that if  $v \in S$  then  $V^r(v) \models \psi(v)$ . Such sentence  $\varphi$  will be called *local*.

**Proposition 7.1** ([Ga]) *Every first-order sentence without individual constants is logically equivalent to a boolean combination of local sentences.*

First we prove that, for every plebeian sentence preserved under homomorphisms,  $\text{Mod}(\varphi)$  is 0-narrow.

**Lemma 7.1** *Suppose that a plebeian sentence  $\varphi$  is preserved under homomorphisms. There exist  $d$  and  $m$  such that no minimal model of  $\varphi$  has a  $d$ -scattered set of cardinality  $m$ .*

**Proof** By virtue of Proposition 7.1, we may suppose that  $\varphi$  is a boolean combination of local sentences  $\varphi_i$ ,  $1 \leq i \leq j$ . Each  $\varphi_i$  asserts that there is a  $2r_i$ -scattered subset of cardinality  $n_i$  such that if  $v$  belongs to the subset then  $V^{r_1}(v) \models \psi_i(v)$ .

Let  $d = 4 \max_i r_i$ . For each  $i$ , write down a formula  $\Psi_i(u)$  asserting the existence of  $v$  such that  $\delta(u, v) \leq r_i$  and  $V^{r_1}(v) \models \psi_i(v)$ . In any structure whose signature is that of  $\varphi$ , define two elements  $v_1, v_2$  to be *equivalent* if

$$[V^{2r_i}(v_1) \models \Psi_i(v_1)] \longleftrightarrow [V^{2r_i}(v_2) \models \Psi_i(v_2)]$$

for all  $i$  in  $[1..j]$ . Set  $m = 2^j + 1$ .

By contradiction, suppose that there is a minimal model  $A$  for  $\varphi$  with a  $d$ -scattered subset  $S$  of cardinality  $m$ . Since  $m > 2^j$ ,  $S$  contains equivalent elements  $a \neq a'$ . It is impossible that both  $a$  and  $a'$  are isolated (i.e. incident to no edge) because if they are then the identification of  $a'$  with  $a$  is a homomorphism of  $A$  onto a proper substructure of  $A$ . Without loss of generality, there exists an edge  $e$  that involves  $a$ . Let  $B$  be the result of removing  $e$  from  $A$ . By the minimality of  $A$ ,  $B$  fails to satisfy  $\varphi$ . Let  $n = \max_i n_i$ ,  $B_n$  be the disjoint sum of  $n$  copies of  $B$ , and  $A_n = A + B_n$ . There exists an (injective) homomorphism from  $A$  into  $A_n$  and therefore  $A_n$  satisfies  $\varphi$ . There is a (surjective) homomorphism from  $B_n$  onto  $B$  and therefore  $B_n$  does not satisfy  $\varphi$ .

To get a contradiction, we show that no  $\varphi_i$  distinguishes between  $A_n$  and  $B_n$ . By the symmetry, we may restrict attention to the case  $i = 1$ . Since  $B_n$  is isomorphic to a induced substructure of  $A_n$ ,  $A_n$  satisfies  $\varphi_1$  if  $B_n$  does. We suppose that  $A_n$  has a  $2r_1$ -scattered subset  $X$  of cardinality  $n_1$  such that  $V_A^{r_1}(x) \models \psi_1(x)$  for all  $x \in X$  and prove that  $B_n$  has such a subset as well. The case  $n_1 > 1$  is easy. In this case, the  $r_1$ -vicinity of some  $x \in X$  does not contain  $e$ . Then  $V_B^{r_1}(x) = V_A^{r_1}(x)$  and therefore each of the  $n$  summands of  $B_n$  has an  $r_1$  vicinity isomorphic to  $V_A^{r_1}(x)$ .

Suppose  $n_1 = 1$  and let  $x$  be the only element of  $X$ . It suffices to prove that  $A$  contains an element  $y$  such that  $V_A^{r_1}(y)$  does not contain  $e$  and  $V_A^{r_1}(y) \models \psi_1(y)$ . If  $V_A^{r_1}(x)$  does not contain  $e$ , we have finished; so suppose that  $V_A^{r_1}(x)$  contains  $e$ . Then  $\delta(a, x) \leq r_1$  and therefore  $V_A^{2r_1}(a)$  satisfies  $\Psi_1(a)$ . Recall that  $a'$  is equivalent to  $a$  and  $\delta(a, a') > d \geq 4r_1$ . Hence  $V_A^{2r_1}(a')$  satisfies  $\Psi_1(a')$  and does not contain  $e$ . Hence there exists  $y \in A$  such that  $V_A^{r_1}(y) \models \psi_1(y)$  and  $V_A^{r_1}(y)$  is included into  $V_A^{2r_1}(a')$  and therefore does not contain  $e$ .  $\square$

In the remainder of the section we suppose that  $\varphi$  is a plebeian sentence preserved under homomorphisms and  $s$  is a positive integer, and we prove that  $\text{Mod}(\varphi)$  is  $s$ -narrow.

Let  $\sigma$  be the extension of the signature of  $\varphi$  by  $s$  individual constants  $c_1, \dots, c_s$  and let  $\varphi'$  be the plebeian companion of  $\varphi$  with respect to  $\sigma$  as defined in Section 5 (at the beginning of the proof of Lemma 5.2).

The sentence  $\varphi'$  is preserved under homomorphisms. For suppose that  $h$  is a homomorphism from  $B_1$  into  $B_2$ . As we saw in Section 5 (in the proof of Lemma 5.1), each  $B_i$  is the plebeian companion of some  $A_i$ . In the obvious way, the homomorphism  $h$  extends to a homomorphism from  $A_1$  into  $A_2$ . We have,

$$[B_1 \models \varphi'] \longrightarrow [A_1 \models \varphi] \longrightarrow [A_2 \models \varphi] \longrightarrow [B_2 \models \varphi'].$$

By the previous lemma, there exist  $d$  and  $m$  such that no minimal model of  $\varphi'$  has a  $d$ -scattered subset of cardinality  $m$ . Fix appropriate  $d$  and  $m$ . We prove that, for no minimal model  $A$  of  $\varphi$ , there is an induced subgraph  $H$  of  $G(A)$  such that  $\|A\| - \|H\| \leq s$  and  $H$  has a  $d$ -scattered subset of cardinality  $m$ .

By contradiction, suppose that  $A$  and  $H$  form a counter-example. Let  $A^+$  be the enrichment of  $A$  obtained by interpreting the individual elements  $c_1, \dots, c_s$  by means of elements in  $|A| - |H|$

in such a way that all those elements become distinguished. It is easy to see that the plebeian companion  $A'$  of  $A$  is a minimal model for  $\varphi'$  and it has a  $d$ -scattered subset of cardinality  $m$ . This contradicts the previous lemma. Theorem 6.1 is proved.  $\square$

## 8 Nostrums

In order to prove Theorem 6.2, we introduce and study objects that we call nostrums.

**Definition 8.1** *A nostrum is a forest  $F$  together with a nonempty set  $V$  and a function that assigns a nonempty connected subset of  $F$  to each element of  $V$ .*

We use the following terminology and notation. Elements of  $F$  are *nodes*, and elements of  $V$  are *vertices*. The set of nodes assigned to a vertex is a *twig*. It is often convenient to view the given nostrum as the forest together with the function that assigns to each node  $X$  its *grasp*  $\{v : X \in \text{Twig}(v)\}$ . We will be interested in nostrums of bounded grasp-size. Bruno Courcelle [Cou] pointed out that the notion of nostrums of bounded grasp-size is related to the notion of the bounded-width tree decompositions of Robertson and Seymour [RS].

Further, a sequence  $(X_0, \dots, X_k)$  of nodes forms a *bridge* from  $X_0$  to  $X_k$  if, for every pair  $(X_i, X_{i+1})$  of successive nodes, some  $\text{Twig}(v)$  contains both nodes. A forest path  $P$  *embeds* a bridge  $B$  if (i)  $B$  is a subsequence, not necessarily contiguous, of  $P$  and (ii) for every two successive members  $X, Y$  of  $B$ , some twig includes the corresponding segment  $[X, Y]$  of  $P$ .

**Lemma 8.1** *For every bridge  $B$  from  $U_1$  to  $U_2$ , there exists a bridge  $B'$  from  $U_1$  to  $U_2$  of the same or smaller length that is embedded into the shortest path from  $U_1$  to  $U_2$ .*

The shortest path from  $U_1$  to  $U_2$  goes from  $U_1$  straight up to the youngest common ancestor of  $U_1$  and  $U_2$  and then straight down to  $U_2$ .

**Proof** First we notice that  $B$  is embedded into some path  $P$ . One such path can be constructed by replacing every two-element segment  $[X, Y]$  of  $B$  with the shortest path from  $X$  to  $Y$ ; the resulting path embeds  $B$  because every twig (as any other connected set) that contains  $X$  and  $Y$  includes the shortest path from  $X$  to  $Y$ .

If the sequence  $P$  has no repetitions then  $P$  is the shortest path from  $U_1$  to  $U_2$ , and we have finished. Suppose that  $P$  contains a segment  $Q$  from some  $Z$  to the same  $Z$ . It suffices to prove that the path  $P'$  obtained from  $P$  by replacing  $Q$  with one node  $Z$  embeds a bridge  $B'$  from  $U_1$  to  $U_2$  whose length is bounded by the length of  $B$ .

If no node of  $B$  is in  $Q$ , choose  $B' = B$ . Otherwise, let  $P_1$  be the initial segment of  $P$  bordering upon  $Q$ ,  $B_1$  be the initial segment of  $B$  embedded in  $P_1$ ,  $X$  be the final node of  $B_1$  and  $X'$  be the successor of  $X$  in  $B$ . Clearly  $X' \in Q$ . Since  $P$  embeds  $B$  and  $Z$  belongs to the segment  $[X, X']$  of  $P$ , some twig includes the segment  $[X, Z]$  of  $P'$ . Similarly, some twig contains  $Z$  and the initial node  $Y$  of the final segment  $B_2$  of  $B$  that is embedded into the final segment  $P_2$  of  $P$  bordering upon  $Q$ . The desired  $B'$  is composed of  $B_1$ ,  $Z$  and  $B_2$ .  $\square$

The forest of a nostrum  $N$  is denoted  $\text{Fr}(N)$ , and the vertex set is denoted  $\text{VS}(N)$ .  $N$  *dominates* a nostrum  $N'$  if  $\text{VS}(N) = \text{VS}(N')$  and, for every node  $Y$  of  $N'$ , there exists a node of  $N$  whose grasp includes that of  $Y$ . We introduce three transformations of nostrums where some nodes are discarded but the grasps of the surviving nodes are not changed.

To perform the first transformation, discard all empty-grasp nodes. Surviving nodes are ordered as before; if a surviving node  $X$  loses its parent but retains at least one proper ancestor then the

youngest surviving proper ancestor of  $X$  becomes the parent of  $X$ . The result is a nostrum that dominates the original one.

A *weak child* is a node  $X$  with a parent such that the grasp of the parent includes the grasp of any descendent of  $X$  (including  $X$ ). To perform the second transformation, discard all weak children and their descendants. The resulting nostrum dominates the original one.

A *weak dynasty* is a maximal node sequence  $(X_1, \dots, X_k)$  where every  $X_{i-1}$  is the only child of  $X_i$  and the grasp of  $X_{i-1}$  includes that of  $X_i$ . To perform the third transformation, reform every weak dynasty  $(X_1, \dots, X_k)$  by discarding nodes  $X_2, \dots, X_k$ . Again, the order of surviving nodes does not change; if  $(X_1, \dots, X_k)$  was a weak dynasty and  $X_k$  had a parent  $Y$ , then  $Y$  becomes the parent of  $X_1$ . The resulting nostrum dominates the original one.

A nostrum is *strong* if it has no empty-grasp nodes, no weak children and no weak dynasties. Performing the three transformation (in the order they were introduced) results in a strong nostrum that dominates the original one; the combined transformation will be called *simplification*.

**Lemma 8.2** *Let  $N$  be a strong nostrum,  $u$  a vertex of  $N$ ,  $X$  a node of  $N$ ,  $X'$  the parent of  $X$  and  $X_0$  a child of  $X$ . Suppose that  $X, X'$  and  $X_0$  grasp  $u$ , and  $X$  grasps at least one other vertex. Discard  $u$  and simplify the resulting nostrum. The node  $X$  survives the simplification.*

**Proof** Obviously,  $X$  survives the first simplification stage.

We show that all descendants of  $X$  in  $N$  survive the second stage. It suffices to show that an arbitrary leaf  $Y \leq X$  survives the first two stages. Since  $N$  is strong,  $Y$  isn't a weak child in  $N$ . Hence it grasps a vertex  $v$  not grasped by its parent and therefore not grasped by any other node. In particular,  $v$  is not grasped by  $X$  and thus  $v \neq u$ . Thanks to  $v$ ,  $Y$  survives the first two stages.

By contradiction, suppose that  $X$  is discarded during the third simplification stage. This means that, after the second stage,  $X_0$  is the only child of  $X$  and the grasp of  $X_0$  includes that of  $X$ . But then the same is true in  $N$  which contradicts the fact that  $N$  is strong.  $\square$

**Definition 8.2** *A graph  $G$  admits a nostrum if there is a nostrum  $N$  such that the universe of  $G$  is the vertex set of  $N$  and every edge of  $G$  is within the grasp of some node of  $N$ .*

If  $N$  satisfies the requirement in the above definition, we say that  $G$  and  $N$  are *legal* for each other.

**Lemma 8.3** *Suppose that  $N$  is a legal nostrum for a graph  $G$  and  $u \in G$  and  $|G| - \{u\} \neq \emptyset$ . Let  $H$  be the induced subgraph of  $G$  with universe  $|G| - \{u\}$ . Discard  $u$  from  $N$  and simplify the remaining nostrum  $N_0$ . The result is a strong nostrum legal for  $H$ .*

**Proof** Let  $e$  be an edge of  $H$ . It is also an edge of  $G$ . Hence there exists a node  $X$  of  $N$  that grasps both ends of  $e$ . Clearly,  $X$  grasps both ends of  $e$  in  $N_0$ . The simplification  $N_1$  of  $N_0$  dominates  $N_0$  and therefore some node of  $N_1$  grasps both ends of  $e$ .  $\square$

**Lemma 8.4** *If  $G$  is a legal graph for a nostrum  $N$ ,  $X$  grasps  $x$  and  $Y$  grasps  $y$ , then the length of the shortest bridge from  $X$  to  $Y$  is bounded by 1 plus the distance between  $x$  and  $y$  in  $G$ .*

**Proof** Let  $z_0, \dots, z_k$  be a path from  $x$  to  $y$  in  $G$ . For every positive  $i \leq k$ , let  $Z_i$  be a node that grasps both  $z_{i-1}$  and  $z_i$ . Then  $X, Z_1, \dots, Z_k, Y$  is a bridge from  $X$  to  $Y$ .  $\square$

Let  $N$  be a legal nostrum for a graph  $G$ . A set  $I$  of nodes and vertices of  $N$  is a *marsh* if:

- Any  $I$ -vertex is disconnected in  $G$  from the other  $I$ -vertices and from the vertices grasped by  $I$ -nodes.
- The  $I$ -nodes that belong to the same tree form a chain, i.e., a connected linearly ordered subset.

**Theorem 8.1** *Suppose that a graph  $G$  admits a strong nostrum that has a marsh of cardinality  $n \geq m^{s+1}(d+2)^s$  with marsh-nodes of grasp-size  $\leq s$ . Then  $G$  has an induced subgraph  $H$  such that  $\|G\| - \|H\| \leq s$  and  $H$  has a  $d$ -scattered subset of cardinality  $m$ .*

**Proof** An induction on  $s$ . Suppose that either  $s = 1$  or else  $s > 1$  and the lemma is proved for  $s - 1$ . Define  $C(t) = m^{t+1}(d+2)^t$  and let  $N$  be a legal strong nostrum for  $G$  with a marsh  $I$  of cardinality  $C(s)$  with nodes of grasp-size  $\leq s$ .

Case 1: There exists a vertex  $u$  such that  $\|I \cap \text{Twig}(u)\| \geq C(s-1) + 2$ .

By the definition of marshes,  $I \cap \text{Twig}(u)$  is a chain. Remove the end-nodes from the chain and let  $J$  be the remaining chain of cardinality  $C(s-1)$ . Let  $G'$  be the induced subgraph of  $G$  with universe  $|G| - \{u\}$ . Remove  $u$  from  $N$  and simplify the remaining nostrum  $N_0$ . By Lemma 8.3, the resulting strong nostrum  $N'$  is legal for  $G'$ .

By Lemma 8.2, every node of  $J$  that grasps a vertex different from  $u$  survives the simplification. Thus the grasp of every discarded node of  $J$  is  $\{u\}$ .

There is a function  $f$  that assigns a vertex in  $N'$  to each discarded node in  $J$  in such a way that the surviving part of  $J$  and the range of  $f$  form a marsh  $I'$  for  $N'$ . Indeed, consider a node  $X$  in  $J$  discarded during the simplification. If  $X_0$  is the child of  $X$  in  $I$ , we have  $\text{Grasp}(X) = \{u\} \subseteq \text{Grasp}(X_0)$ . Since  $N$  has no weak dynasties,  $X$  has another child  $X_1$ . Let  $X_2$  be any leaf descendant of  $X_1$  in  $N$ . Since  $N$  has no weak children, there is a vertex grasped by  $X_2$  only. Choose one such vertex to be  $f(X)$ . We check that, in  $G'$ ,  $f(X)$  is disconnected from any vertex  $v$  such that  $v = f(Y)$  for some other discarded  $I$ -node  $Y$  or else  $v$  is grasped by a surviving  $I$ -node. By contradiction, suppose that  $f(X)$  is connected to  $v$  and consider nostrum  $N_0$ . By Lemma 8.4, there is a bridge  $B$  from  $X_2$  to a node  $Z$  grasping  $v$ . By Lemma 8.1, the shortest path  $P$  from  $X_2$  to  $Z$  embeds a bridge from  $X_2$  to  $Z$ . The node  $X$  is necessarily on  $P$  and belongs to some twig which is impossible.

If  $s = 1$  then  $I'$  is composed of  $\geq m$  disconnected elements of  $G'$ . Thus  $G'$  is the desired  $H$ .

Suppose that  $s > 1$ . By the induction hypothesis, there is an induced subgraph  $H'$  of  $G'$  such that  $\|G'\| - \|H'\| \leq s - 1$  and  $H'$  has a  $d$ -scattered set  $S$  of cardinality  $m$ .  $H'$  is the desired  $H$ .

Case 2: For every vertex  $u$ ,  $\|I \cap \text{Twig}(u)\| \leq m^s(d+2)^{s-1} + 1$ .

Let  $b = n/m$ . We consider only the case when  $I$  has no vertices and all nodes of  $I$  belong to the same tree; other cases are even easier. In our case,  $I$  is a chain  $(X_1, \dots, X_{bm})$ . Set  $S = \{X_{bi} : 1 \leq i \leq m\}$ .

If  $(Z_0, \dots, Z_k)$  is a bridge from one node of  $S$  to another then  $k \geq d + 2$ . By contradiction, suppose that  $k \leq d + 1$ . By Lemma 8.1, we may suppose that the given bridge is embedded in the shortest path from  $Z_0$  to  $Z_k$ . Since the distance between  $Z_0$  and  $Z_k$  is at least  $n/m$ , the average distance between  $Z_i$  and  $Z_{i+1}$  is at least  $n/(mk)$  which exceeds  $m^s(d+2)^{s-1}$ . Hence there exists  $i$  such that the distance between  $Z_i$  and  $Z_{i+1}$  is at least  $m^s(d+2)^{s-1} + 1$ , so that the interval  $[Z_i, Z_{i+1}]$  contains at least  $m^s(d+2)^{s-1} + 2$  members. According to the definition of embedding a bridge into a path, there exists a vertex  $u$  whose twig includes  $[Z_i, Z_{i+1}]$  which contradicts Case 2.

We define a one-to-one function  $f$  from  $S$  to  $G$ .  $f$  coincides with the identity function on the vertices of  $S$ . If  $X$  is a node in  $G$ , let  $f(X)$  be an arbitrary vertex in the grasp of  $X$ . By Lemma 8.4,  $R$  is  $d$ -scattered in  $G$ . The desired  $H$  is  $G$ .  $\square$

## 9 Global properties

In this section, we prove Theorem 6.2. Suppose that a plebeian boolean query  $\mathcal{Q} = (\Pi, Q)$  is unbounded. The equality sign can be eliminated from  $\Pi$  without changing the meaning of  $\mathcal{Q}$ . Thus, we may assume that the equality sign does not appear in  $\Pi$ .

Call an individual variable *relevant* to a rule  $\rho$  if it appears in the head of the rule or in at least two atomic formulas in the body. Let  $s$  be the maximal number of variables relevant to any rule in  $\Pi$ . We will prove that  $\text{Mod}(\mathcal{Q})$  is  $s$ -wide. Pick arbitrary  $d$  and  $m$ . With respect to Theorem 8.1, it suffices to prove that there exists a minimal  $A \in \text{Mod}(\mathcal{Q})$  whose graph admits a strong nostrum with nodes of grasp-size  $\leq s$  and a marsh of cardinality  $n \geq m^{s+1}(d+2)^s$ .

Consider a calculus  $(\Pi, D)$  where  $D$  is an arbitrary database for  $\Pi$ . Statements of the form  $R(a_1, \dots, a_r)$ , where  $R$  is an extensional or intentional predicate and  $a_1, \dots, a_r \in D$ , will be called *D-claims*.

A *proof* of a  $D$ -claim  $\alpha$  is a tree labeled with  $D$ -claims in such a way that:

- Leaves are labeled with edges and the root is labeled with  $\alpha$ , and
- If a node  $X$  has  $k$  children then there exist a rule  $\alpha_0 \leftarrow \alpha_1, \dots, \alpha_k$  and an instantiation  $I$  of the variables with elements of  $D$  such that  $I(\alpha_0)$  is the label of  $X$  and  $I(\alpha_1), \dots, I(\alpha_k)$  are the labels of the children of  $X$ .

If an element  $a \in D$  appears in the label of a node  $X$  or in the labels of at least two children of  $X$ , we say that  $a$  and  $X$  are *relevant* to each other.

It is well known (and easy to check) that no sentence without equality distinguishes between a structure  $D$  and the structure  $D'$  obtained from  $D$  by replacing an element  $a$  with an arbitrary number  $k \geq 1$  of indistinguishable copies  $a_1, \dots, a_k$  of  $a$ . We explain more exactly what  $D'$  is. The universe of  $D'$  is obtained from  $D$  by removing  $a$  and adding  $a_1, \dots, a_k$  instead. A  $D'$ -edge  $\alpha$  is true if and only if the  $D$ -edge  $\alpha'$  obtained from  $\alpha$  by replacing  $a_1, \dots, a_k$  with  $a$  is true in  $D$ . (In other words,  $D'$  treats  $a_1, \dots, a_k$  as aliases for  $a$ .)

Let  $D$  be a minimal model of  $\mathcal{Q}$  satisfying a certain condition. For expository reason we delay specifying the condition. For the moment it is important only that the empty subset of  $D$  does not generate  $Q$ .

Let  $P_0$  be a shortest proof of  $Q$  in  $(\Pi, D)$ . Clearly, every element and every edge of  $D$  appears in  $P_0$ .

Rewrite  $P_0$  using fresh elements whenever possible. More exactly, for each element  $a \in D$  do the following. In each connected component  $C$  of the set of nodes relevant to  $a$  replace  $a$  with an indistinguishable copy  $a_C$ . Let  $D'$  be the structure obtained from  $D$  by replacing each  $a$  with indistinguishable copies  $a_C$ .

Let  $P$  be the result of this transformation of  $P_0$ . It is easy to check by induction that, for each node  $X$  of  $P$ , the induced substructure of  $P$  comprising the descendants of  $X$  is a proof over  $D'$ . We address only the subtlety that arises when an element  $a$  is relevant to children  $X_1, X_2$  of  $X$  but not to  $X$  itself. In  $P$ , some copy  $a_1$  (respectively  $a_2$ ) of  $a$  appears in  $X_1$  (respectively  $X_2$ ). The danger was that  $a_1$  and  $a_2$  are different and then the  $P$ -label of  $X$  may not follow from the  $P$ -labels of its children. This danger is avoided by our definition of relevance. Since  $a$  appears in the  $P_0$  labels of  $X_1, X_2$ , it is relevant to  $X$  as well as to  $X_1, X_2$  in  $P_0$ . Thus, all three nodes lie in the same connected component of the set of elements relevant to  $a$ , and therefore  $a_1, a_2$  are the same.

The desired structure  $A$  is obtained from  $D'$  by removing all edges that do not appear in  $P$ . Clearly,  $A$  is a minimal model of  $\mathcal{Q}$ .



$G(A)$  admits a nostrum. Indeed, let  $N'$  be the nostrum where  $P$  is the forest,  $|A|$  is the vertex set, and the twig of any  $a \in A$  comprises the nodes relevant to  $a$ . Clearly  $N'$  is legal for  $G(A)$ . Unfortunately,  $N'$  is not necessarily strong. How can we find the desired strong nostrum legal for  $G(A)$ ? One may play with  $\Pi$  (before constructing  $P$ ) to insure that  $P$  is strong. In that approach  $D$  should be chosen in such a way that it generates  $Q$  on stage  $\geq n$  of the evolution. Then a longest branch of  $P$  is the desired marsh.

We choose a quick and dirty (and wasteful) solution. Chose  $D$  in such a way that every substructure of  $D$  that generates  $Q$  is of cardinality  $\geq b^n s$  where  $b$  is the maximal number of atoms in the body of any rule of  $\Pi$  (unless this number is 1 in which case let  $b = 2$ ). Let  $N$  be the result of simplification of  $N'$ . Clearly,  $N$  is legal for  $G(A)$ .

The number of nodes in  $N$  is  $\geq b^n$ . For, the equivalence relation “twigs of  $x$  and  $y$  have the same root” partitions  $A$  into blocks of cardinality  $\leq s$ , and the number of nodes is at least as large as the number of blocks.

Let  $M$  range over nostrums of cardinality  $\geq b^n$  with nodes of grasp-size  $\leq s$ , and let  $c(M)$  be the cumulative depth of the trees in  $\text{Fr}(M)$ . It suffices to prove that each  $c(M) \geq n$ . For, then  $c(N) \geq n$  and we can construct the desired marsh as follows. Pick a longest branch in each tree of  $\text{Fr}(N)$ . The union of the branches is a marsh of cardinality  $\geq c(N) + 1 > n$ . (The branch of length  $k$  has  $k + 1$  nodes.)

It suffices to prove that  $c(M) \geq n$  in the case when  $M$  is a tree because  $c(M)$  is the smallest when  $M$  is a tree. The tree  $\text{Fr}(M)$  has  $c(M)$  levels (with the root being on level zero). By the definition of  $b$ , there are  $\leq b^i$  elements on the level  $i$ . If  $c(M) < n$  then the total number of nodes is  $< \sum_{i < n} b^i < b^n$ . Therefore  $c(M) \geq n$ .  $\square$

## 10 Two Extensions of Datalog

Call a datalog program *pure* if it contains no occurrences of the equality sign. A rule of a pure program has the form  $\alpha \leftarrow \beta_1, \dots, \beta_k$  where  $\alpha$  as well as each  $\beta_i$  is a proper atomic formula. We consider two generalizations of pure datalog. In the first generalization, called for brevity *datalog with negations*, each  $\beta_i$  is either a proper atomic formula or the negation of a proper atomic formula. In the second generalization, called for brevity *datalog with inequalities*, each  $\beta_i$  is either a proper atomic formula or an inequality  $e_1 \neq e_2$ . First we show that there exists an unbounded plebeian query with negations equivalent to a first-order formula. Then we show that there exists an unbounded plebeian query with inequalities equivalent to a first-order formula.

In model theory, a first-order sentence  $\varphi$  is said to have the *extension property* if, for every structure  $A$  of the appropriate signature and every induced substructure  $B$  of  $A$ ,  $B \models \varphi$  implies  $A \models \varphi$ . If  $\varphi$  expresses a query in datalog with negations or inequalities or both then  $\varphi$  has the extensions property; moreover,  $\varphi$  is equivalent to an existential formula if and only if the query is bounded. If infinite structures are allowed, then, by a classical theorem, every formula with the extension property is equivalent to an existential formula.

In finite model theory, the situation is different. Gurevich and Shelah [Gu] constructed a first-order sentence  $\gamma_0$  that is preserved by induced substructures but is not equivalent to any universal first-order sentence. It follows that  $\neg\gamma_0$  has the extension property but is not equivalent to any existential formula. Kolaitis and Vardi [KV] constructed an unbounded query in datalog with negations *and* inequalities. This gives a counter-example to the analog of Theorem 1.1 for datalog with negations and inequalities. We offer a little improvement of Gurevich-Shelah’s formula.

Say that a formula with one and only free variable has the *extension property* if the sentence obtained by replacing the free variable with a fresh individual constant has the extension property.

Let  $\gamma(v)$  be a first-order formula with equality saying the following:

If  $<$  is a linear order with a minimal element 0, and a binary relation  $S$  is consistent with the successor relation of  $<$ , then for every  $x < v$  there exists  $y$  with  $xSy$ .

The consistency of  $E$  with the successor relation means that, for all  $x, y$ , if  $xSy$  then  $y$  is the successor of  $x$  with respect to  $<$ .

**Lemma 10.1**  $\gamma(v)$  has the extension property and is not equivalent to any existential formula.

**Proof** To check the extension property, let  $B$  be an induced substructure of  $A$  and  $B \models \gamma(a)$ . Suppose that the premise of  $\gamma(a)$  holds in  $A$ . It is easy to see that the premise holds in  $B$ . Since  $B \models \gamma(a)$ , for every  $x < a$  in  $B$ , there exists  $y$  such that  $B \models xSy$ . Using the induction principle, check that every element  $x \in A$  such that  $A \models x < a$  belongs to  $B$ . (To establish the base of induction, use the fact that 0, being a distinguished element, belongs to  $B$ .) Now suppose that  $x < a$  in  $A$  and let  $y$  be such that  $B \models xSy$ ; then  $A \models xSy$ .

By contradiction, suppose that  $\gamma(v)$  is equivalent to an existential sentence

$$(\exists u_1, \dots, u_k) \Delta(v, u_1, \dots, u_k)$$

where  $\Delta$  is quantifier free. Consider a model  $B$  for for the premise of  $\gamma(v)$  such that  $\|B\| \geq k + 3$  and  $S$  coincides with the successor relation. Choose  $v$  to be the maximal element. Fix elements  $x_1, \dots, x_k$  such that  $B \models \Delta(v, x_1, \dots, x_k)$  and choose a non-initial and non-final element  $y$  different from all  $x_i$ . If  $y'$  is the successor of  $y$  in  $B$ , discard the edge  $ySy'$  of  $S$ . The resulting structure satisfies the existential formula but does not satisfy  $\gamma(v)$ .  $\square$

It is easy to eliminate equality and 0 from  $\gamma$ . Introduce a unary predicate  $Z$  and abbreviate  $x \not< y \wedge y \not< x$  as  $xEy$ . Let  $\gamma_1(v)$  be a formula saying:

If

- $<$  is a partial order,
- $E$  is an equivalence relation and the equivalence classes of  $E$  are linearly ordered by  $<$ ,
- $S$  respects  $E$  and is consistent with the successor relation of  $<$ , and
- if  $Z \neq \emptyset$  then  $Z$  coincides with the minimal  $E$ -class,

then  $Z \neq \emptyset$  and, for every  $x < v$ , there exists  $y$  such that  $xSy$ .

(The formula would look a little more natural if we delete the two occurrences of statement  $Z \neq \emptyset$  and then form the conjunction of statement  $Z \neq \emptyset$  and the doctored implication. The reason for the present, more awkward form is purely technical.)

The global relation of  $\gamma_1(v)$  can be expressed in datalog with negations. Let  $\Pi_1$  be the following program:

$$Qv \leftarrow x < x \quad (18) \qquad Qv \leftarrow xSy, x \not< y \quad (24)$$

$$Qv \leftarrow x < y, y < z, x \not< z \quad (19) \qquad Qv \leftarrow xSz, x < y, y < z \quad (25)$$

$$Qv \leftarrow xEy, z < x, z \not< y \quad (20) \qquad Qv \leftarrow Z(x), y < x \quad (26)$$

$$Qv \leftarrow xEy, x < z, y \not< z \quad (21) \qquad Qv \leftarrow Z(x), xEy, \neg Z(y) \quad (27)$$

$$Qv \leftarrow xEy, xSz, \neg(ySz) \quad (22) \qquad Qv \leftarrow Z(v) \quad (28)$$

$$Qv \leftarrow xEy, zSx, \neg(zSy) \quad (23) \qquad Qv \leftarrow Qu, uSv \quad (29)$$

$\Pi_1$  has only one intentional relation, the intentional relation is unary, and there is only one recursive rule.

**Theorem 10.1** *The query  $(\Pi_1, Q)$  is unbounded and equivalent to  $\gamma_1(v)$ .*

**Proof** First, we check that if the antecedent of  $\gamma_1(v)$  fails then  $Q$  is universal. To this end we suppose that  $Q$  is not universal and prove the antecedent. By rules 18 and 19,  $<$  is a partial order. By the definition,  $E$  is symmetric. By 18,  $E$  is reflexive. If  $E$  is not transitive, then there are  $x, y, z$  with  $xEy$ ,  $yEz$  and either  $x < z$  or  $z < x$  which contradicts rule 20 or rule 21. Thus,  $E$  is an equivalence relation. It is easy to check that the equivalence classes of  $E$  are linearly ordered by  $<$ . By rules 22 and 23,  $S$  respects  $E$ . By rules 24 and 25,  $S$  is consistent with the successor relation of  $<$ . By rules 26 and 27, if  $Z \neq \emptyset$  then it coincides with the  $<$ -minimal  $E$ -class.

Now we can restrict attention to the case when the antecedent of  $\gamma_1(v)$  holds. In this case, the rules 18–27 do not fire. Say that an equivalence class  $Y$  of  $E$  is the *successor* of the equivalence class  $X$  if there are  $x \in X$  and  $y \in Y$  such that  $xEy$  holds. Let  $I$  be the smallest collection of equivalence classes that contains  $Z$  and is closed under the successor function. By rules 28 and 29, the intended value  $Q^*$  of  $Q$  is the union of  $I$ . It is easy to see that the succedent of  $\gamma_1(v)$  holds if and only if  $v$  belongs to that union. Thus,  $\gamma_1(v)$  and  $(\Pi, Q)$  are equivalent.

The unboundedness of  $(\Pi, Q)$  is obvious.  $\square$

We turn attention to datalog with inequalities. Recall that a formula or program is plebeian if it has no individual constants.

**Theorem 10.2** *In the case of datalog with inequalities, there exists an unbounded plebeian query that is first-order expressible.*

**Proof** We start with constructing a first-order expressible unbounded query  $(\Pi_2, Q)$  with one individual constant 0. Here is  $\Pi_2$ :

$$\begin{array}{ll}
Qv \leftarrow x < x & 0G0 \\
Qv \leftarrow x < y, y < x & 0Gy \leftarrow xGx, xSy, 0 < y \\
Qv \leftarrow xSy, xSy', y \neq y' & x'Gy \leftarrow xGy, xSx', x' < y \\
Qv \leftarrow xSy, x'Sy, x \neq x' & yGy \leftarrow xGy, xSy \\
Qv \leftarrow xSy, y < x & xBy \leftarrow x < y, yGy \\
Qv \leftarrow xSz, x < y, y < z & xBy \leftarrow xBy', ySy', x \neq y \\
Qv \leftarrow x < 0 & Qv \leftarrow xB0 \\
Qv \leftarrow xS0 & Qv \leftarrow vGv
\end{array}$$

To check that  $Q$  is unbounded, consider an initial segment  $[0..k]$  of natural numbers with  $S$  being the successor relation. It is easy to see that the intended value of  $G$  is the relation  $\leq$  but  $k$  steps do not suffice to generate  $G$ .

Next we show that  $Q$  is first-order expressible. Let

$$\delta = \forall x \forall y \bigwedge_{i=1}^8 \neg \beta_i$$

where  $\beta_i$  is the conjunction of the members of the body of the  $i$ -th clause of  $\Pi_2$ .  $\delta$  says that  $<$  is irreflexive and anti-symmetric, that no element has more than one successor or more than

one predecessor with respect to  $S$ , that 0 is a minimal element with respect to  $<$  and it has no predecessors with respect to  $S$ , and that  $S$  is consistent with  $<$  in the following sense: if  $xSy$  then  $\not y < x$  and there is no  $z$  with  $x < z < y$ . The sentence  $\delta$  fails if and only if the first 8 rules establish the universality of  $Q$ .

We may restrict attention to the class  $K = \text{Mod}(\delta)$ . For, the formula  $\delta \rightarrow \varepsilon(v)$  expresses  $Q$  if the formula  $\varepsilon(v)$  expresses  $Q$  on  $K$ .

View  $S$  as the graph of a partial function  $s$ . Define  $\bar{0} = 0$  and  $\overline{i+1} = s(\bar{i})$ . Let  $N$  be the greatest number  $i$  such that  $\bar{i}$  exists in the given database. It is easy to check by induction on  $i$  that if  $i < j \leq N$  then  $\bar{i} \neq \bar{j}$ . (Use the fact that if  $i < j \leq N$  but  $\bar{i} = \bar{j}$  then  $\overline{j-1}$  is a predecessor of  $\bar{i}$ .)

Let  $n$  be the greatest number  $i \leq N$  such that for all  $k < j \leq i$ , the given database satisfies  $\bar{k} < \bar{j}$ . If  $x < \bar{i}$  and  $x$  is different from any  $\bar{j}$  with  $j < i$ , we say that  $x$  is a *bastard* of  $\bar{i}$ . Let  $m$  be the greatest number  $i$  such that no  $\bar{j}$  with  $j \leq i$  has any bastards.

Let  $g(v)$  be the conjunction of three formulas saying respectively:

$$\begin{aligned} 0 &\leq v \\ \forall x \forall y [x < v \rightarrow \exists y (xSy \wedge y \leq v)] \\ < &\text{ is a linear order on } \{x : x \leq v\} \end{aligned}$$

where  $\leq$  is the usual abbreviation.

**Lemma 10.2** 1.  $g(v) \leftrightarrow (\exists i \leq m)(v = \bar{i})$ .

2.  $xGx \leftrightarrow (\exists i \leq n)(x = \bar{i})$ .

**Proof** (1) Clearly, the right-hand side implies the left-hand side. To prove the other implication, assume  $g(v)$ . Then 0 is the minimal element in the set  $V = \{x : x \leq v\}$  and, for every  $\bar{i} \in V$ , if  $\bar{i} \neq v$  then  $\overline{i+1} \in V$ . It follows that  $v = \bar{i}$  where  $i$  is the largest number  $j$  with  $\bar{j} \in V$ . If some  $\bar{j} \in V$  has a bastard  $b$  then there is  $k < j$  such that  $\bar{k} < b < \bar{k} + 1$  which is impossible.

(2) It is easy to generate every  $\bar{i}G\bar{i}$  with  $i \leq n$ . Thus, the right-hand side implies the left-hand side. To prove that left-hand side implies the right-hand side, notice that every  $G$ -link has the form  $\bar{i}G\bar{j}$ , that links  $\bar{i}G\bar{j}$  are generated in the lexicographical order and that a link  $\bar{i}G\bar{j}$  is generated only if  $\bar{i} \leq \bar{j}$ .  $\square$

It follows that  $m \leq n$ . Let  $\delta'$  be the universal closure of the formula

$$[g(x) \wedge xSy \wedge (\forall z \leq x)(z < y)] \rightarrow (\forall z < y)(z \leq x).$$

**Lemma 10.3** 1.  $\delta' \leftrightarrow m = n$ .

2. If  $n > m$  then  $Q$  is universal.

**Proof** (1) First suppose  $m = n$  and check  $\delta'$ . The case  $x < m$  is obvious. If  $x = m$  then the antecedent fails and therefore the implication holds. Next suppose  $m < n$  and check that  $x = \bar{m}$  and  $y = \overline{m+1}$  give a counter-example for  $\delta'$ .

(2) Suppose that  $m < n$  and  $x$  is a bastard of  $\overline{m+1}$ . Use Lemma 10.2 to verify that  $\Pi$  generates  $xB\bar{j}$  for every  $j \leq m+1$ . Because of the penultimate rule,  $Q$  is universal.  $\square$

We may restrict attention to the class  $K' \subseteq K$  of databases satisfying  $\delta'$ . For, by Lemma 10.3, the formula  $\delta' \rightarrow \varepsilon(v)$  expresses  $Q$  on  $K$  if the formula  $\varepsilon(v)$  expresses  $Q$  on  $K'$ .

By Lemmas 10.2 and 10.3, the formula  $g(v)$  expresses  $\mathcal{Q}$  on  $K'$ . Thus query  $(\Pi_2, Q)$  is indeed first-order expressible and unbounded. (It is easy to see that  $B$  can be replaced by  $G$  in  $\Pi_2$ .)

To get rid of the individual constant 0, one may use a unary predicate  $Z$ . Alternatively check that the reduction of  $\mathbf{B}$  to  $\mathbf{BOP}$  given above generalizes to datalog with inequalities. We have refuted  $\mathbf{BOP}$ ; it follows that  $\mathbf{B}$  fails as well. Theorem 10.2 is proved.  $\square$

## 11 Implicit First-Order Definability

Let  $\mathcal{R}$  be an  $r$ -ary global relation of signature  $\sigma$  and  $P$  an  $r$ -ary predicate that does not belong to  $\sigma$ .  $\mathcal{R}$  is *implicitly definable* if there exists a first-order sentence  $\varphi(P)$  of signature  $\sigma \cup \{P\}$  such that, for every  $\sigma$ -structure  $A$  and every  $r$ -ary relation  $R$  on  $A$ ,  $(A, R)$  satisfies  $\varphi$  if and only if  $R = \mathcal{R}_A$ .

**Theorem 11.1** *There exists an unbounded datalog query  $\mathcal{Q} = (\Pi, Q)$  such that the global relation of  $\mathcal{Q}$  is implicitly definable.*

**Proof** The desired program  $\Pi$  is

$$\begin{aligned} xQy &\leftarrow xEy \\ xQy &\leftarrow xQz, zQy \\ xQy &\leftarrow xQx', x'Qx', y'Qy', y'Qy \end{aligned}$$

Thus, databases are digraphs, and the intended meaning  $Q^*$  of  $Q$  is obtained from the transitive closure of the relation  $E$  by connecting every ancestor of any circle vertex with every descendent of any circle vertex. In particular, on acyclic graphs,  $Q^*$  is the transitive closure of  $S$ . By Theorem 1.1,  $Q$  is not first-order expressible (without using additional predicate symbols).

The desired sentence  $\varphi$  that defines implicitly the global relation of  $\mathcal{Q}$  is the conjunction of the universal closures of the following formulas:

$$\begin{aligned} xEy &\rightarrow xPy \\ (xPz \wedge zPy) &\rightarrow xPy \\ (xPx' \wedge x'Px' \wedge y'Py' \wedge y'Py) &\rightarrow xPy \\ (xPy \wedge \neg xEy) &\rightarrow (\exists u, v)[(xEu \wedge uPy) \wedge (xPv \wedge vEy)] \end{aligned}$$

Let  $G$  be an arbitrary digraph. Obviously,  $(G, Q^*) \models \varphi$ . Suppose that  $R$  is an arbitrary binary relation on  $G$  such that  $(G, R) \models \varphi$ . It is easy to see that  $R$  includes  $Q^*$ . By contradiction, suppose that  $R$  properly includes  $Q^*$  and pick a pair  $(a, b) \in R - Q^*$ . Obviously,  $G$  has no path from  $a$  to  $b$ . According to the last conjunct of  $\varphi$ , there exist infinite chains  $aEa_1Ea_2 \dots$  and  $\dots b_2Eb_1Eb$ . But  $G$  is finite. Hence some  $a_i$  lies on a circle and some  $b_j$  lies on a circle. Hence  $(a, b) \in Q^*$ , which gives the desired contradiction.  $\square$

## References

**AG** Miklos Ajtai and Yuri Gurevich, *Datalog vs. First-Order Logic*, 30th Annual Symposium on Foundations of Computer Science, IEEE Computer Society, 1989, 142–147.

- Co** Stavros S. Cosmadakis, *On the First-Order Expressibility of Recursive Queries*, Principles of Database Systems, ACM, 1987, 311–323.
- Cou** Bruno Courcelle. Private communication.
- Ga** Haim Gaifman, *On Local and Non-Local Properties*, Proceedings of the Herbrand Symposium, Logic Colloquium '81 (Ed. J. Stern), North-Holland, 1982.
- Gu** Yuri Gurevich, *Toward Logic Tailored for Computational Complexity*, in “Computation and Proof Theory” (Ed. M. Richter et al.), Springer Lecture Notes in Math. 1104 (1984), 175–216
- KV** Phokion Kolaitis and Moshe Y. Vardi. Private communication.
- Me** Frank Messerle. Private communication.
- RS** Neil Robertson and Paul D. Seymour, *Graph minors III: Planar Tree-width*, J. Combinatorial Theory B, vol. 36 (1984), p. 49-64.