INFORMATION AND COMPUTATION **129**, 1–19 (1996)
ARTICLE NO. 0070

Logical Hierarchies in PTIME*

Lauri Hella

Department of Mathematics, P.O. Box 4 (Hallituskatu 15), 00014 University of Helsinki, Finland
E-mail: hella@cc.helsinki.fi

We consider the problem of finding a characterization for polynomial time computable queries on finite structures in terms of logical definability. It is well known that fixpoint logic provides such a characterization in the presence of a built-in linear order, but without linear order even very simple polynomial time queries involving counting are not expressible in fixpoint logic. Our approach to the problem is based on generalized quantifiers. A generalized quantifier is n-ary if it binds any number of formulas, but at most n variables in each formula. We prove that, for each natural number n, there is a query on finite structures which is expressible in fixpoint logic, but not in the extension of first-order logic by any set of n-ary quantifiers. It follows that the expressive power of fixpoint logic cannot be captured by adding finitely many quantifiers to first-order logic. Furthermore, we prove that, for each natural number n, there is a polynomial time computable query which is not definable in any extension of fixpoint logic by n-ary quantifiers. In particular, this rules out the possibility of characterizing PTIME in terms of definability in fixpoint logic extended by a finite set of generalized quantifiers.

1. INTRODUCTION

It is well known that the expressive power of first-order logic, FO, on finite structures is rather limited. This is due to the fact that FO is unable to express any queries requiring non-trivial recursion. For example, there is no first-order formula defining the transitive closure of a binary relation uniformly on all finite structures. Thus, it is natural that researchers of finite model theory and database theory have studied extensions of first-order logic which are strong enough to capture such recursive queries, but, unlike second-order logic, still remain inside the computationally feasible region of PTIME.

There are several alternative ways of enhancing the expressive power of first-order logic. *Least fixpoint logic*, LFP, is obtained by incorporating a recursion mechanism into FO via least fixpoints of positive formulas. This logic was first studied by Moschovakis (1974) under the name *inductive definability* in the context of fixed infinite structures. Later Chandra and Harel (1982) brought LFP into the attention of computer scientists as a language for finite structures. In the context of finite model theory, LFP is perhaps the most successful extension of FO discovered so far. Immerman (1986) and Vardi (1982) proved that, in the presence of a built-in linear order, LFP captures PTIME, i.e., any query on finite structures which contain a linear order as one of their basic relations, is definable by a formula of LFP if and only if it is computable in polynomial time.

The theorem of Immerman and Vardi cited above is one of the cornerstones of *descriptive complexity theory*, an area of research that studies the connections between computational complexity and logical definability. The study of descriptive complexity theory was initiated by Fagin (1974), who proved that NP consists of exactly those problems that are definable by existential second-order sentences. After this pioneering result of Fagin, similar characterizations in terms of definability in various logics have been proved for all the basic complexity classes, including LOGSPACE (Immerman, 1987), PTIME (Immerman, 1986, and Vardi, 1982), and PSPACE (Vardi, 1982).

One of the most intriguing open problems in descriptive complexity theory is whether there exists a reasonable logic that would capture exactly the PTIME computable queries on finite structures. As noted above, this problem was already solved by Immerman and Vardi on the class of finite structures with built-in linear order. However, the presence of linear order is necessary for their theorem to hold, since on the class of all finite (unordered) structures even very simple PTIME computable queries involving counting are not expressible in LFP. The general problem of finding a logical characterization for PTIME was first posed by Chandra and Harel (1982) in a slightly different form, and it has been considered by Immerman (1986) and Gurevich (1988), among others. Gurevich (1988) also gave a precise definition for the phrase "$\mathscr{L}$ is a logic which captures PTIME" and conjectured that no such logic exists.

The simplest way of extending first-order logic, while maintaining its usual closure properties, is to add *generalized quantifiers* corresponding to undefinable properties of structures. The notion of generalized quantifier was introduced by Mostowski (1957), who considered extensions of first-order logic by cardinality quantifiers like

* A preliminary version of this paper appeared in "Proceedings of the 7th IEEE Symposium on Logic in Computer Science," 1992.

0890-5401/96 $18.00

"there exist infinitely many" and "there exist uncountably many." Lindström (1966) gave a more general definition for generalized quantifiers. According to Lindström, any property of structures over some fixed finite vocabulary can be taken as the interpretation of a quantifier. For example, the Ramsey quantifier $Q^n x_1 \cdots x_n \varphi(x_1, ..., x_n)$ has the interpretation "there is an infinite set P such that $\varphi(a_1, ..., a_n)$ holds for all distinct elements $a_1, ..., a_n$ of P", and the Härtig quantifier $Ix, y(\varphi(x), \psi(y))$ is interpreted as "the number of elements satisfying φ is equal to the number of elements satisfying ψ." Thus, a quantifier can bind several variables in one formula and also variables in several formulas simultaneously. We say that a quantifier Q is n-ary if it binds at most n variables in each formula it binds.

Generalized quantifiers offer one possible approach to the problem of logical characterization of PTIME: one can try to find some effectively described set of polynomial time computable quantifiers such that the corresponding extension of first-order logic or fixpoint logic captures PTIME. With this purpose in mind, we study in this paper logics of the form $\mathrm{FO}(\mathbf{Q})$ and $\mathrm{IFP}(\mathbf{Q})$, where $\mathbf{Q}$ is a set of quantifiers expressing properties of finite structures. Here we use *inflationary fixpoint logic*, IFP, as a base logic rather than LFP, since least fixpoint logic can be extended by generalized quantifiers only when all quantifiers considered are monotone. We will in particular consider the possibility of characterizing PTIME by $\mathrm{FO}(\mathbf{Q})$ or $\mathrm{IFP}(\mathbf{Q})$ for some finite set $\mathbf{Q}$ of quantifiers.

Unary generalized quantifiers have already been considered in connection with the problem of capturing PTIME by a logic. Immerman (1986) suggested adding *counting quantifiers* "there are at least n" to fixpoint logic, and extending structures with an extra sort consisting of an initial segment of natural numbers, as a possible solution to the problem. However, Cai *et al.* (1992) proved that this *fixpoint logic with counting* does not even capture all LOGSPACE computable queries on finite graphs. Kolaitis and Väänänen (1995) give other examples of natural PTIME computable queries which are not definable in the extension of fixpoint logic by all unary quantifiers binding only one formula.

As the main result of this paper we will prove two hierarchy theorems for PTIME in terms of arity of generalized quantifiers. First, we show that, for each natural number n, there exists a query which is definable in LFP, but not in $\mathrm{FO}(\mathbf{Q})$ for any set $\mathbf{Q}$ of n-ary quantifiers. Consequently, there cannot exist a finite set $\mathbf{Q}$ of quantifiers such that $\mathrm{FO}(\mathbf{Q})$ has the same expressive power as LFP on finite structures. In fact, the queries which we use in our counterexamples are computable already in the database query language DATALOG, whose programs consist of function-free Horn clauses. Thus, no extension of first-order logic by a finite number of generalized quantifiers captures all DATALOG computable queries.

In our second main result we prove that, for each natural number n, there is a PTIME computable query which is not definable in $\mathrm{IFP}(\mathbf{Q})$ for any set $\mathbf{Q}$ of n-ary quantifiers. In particular, this rules out the possibility of capturing PTIME by a logic obtained by adding a finite number of quantifiers to fixpoint logic. Since the counting quantifiers are unary, our theorem generalizes the above mentioned result of Cai *et al.* (1992).

Both of these hierarchy theorems are based on Ehrenfeucht–Fraïssé games characterizing equivalence of structures with respect to the class $\mathbf{Q}_n$ of all n-ary quantifiers. In Hella (1989), we defined Ehrenfeucht–Fraïssé games which provide necessary and sufficient conditions for equivalence in the infinitary logic $\mathscr{L}_{\infty\omega}(\mathbf{Q}_n)$ with all n-ary quantifiers; in the present paper we use these games of Hella (1989) for proving that two finite structures are $\mathrm{FO}(\mathbf{Q}_n)$-equivalent up to a given quantifier rank. Furthermore, we introduce here new Ehrenfeucht–Fraïssé games that characterize equivalence with respect to $\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)$, where $\mathscr{L}^k_{\infty\omega}$ is the k-variable logic, i.e., the sublogic of the usual infinitary logic $\mathscr{L}_{\infty\omega}$ consisting of formulas which contain at most k distinct variables. It is well known that all queries definable in fixpoint logic (or even in partial fixpoint logic) are expressible in the finite variable logic $\mathscr{L}^\omega_{\infty\omega} = \bigcup_{k\in\omega} \mathscr{L}^k_{\infty\omega}$ (see Kolaitis and Vardi, 1992a). This inclusion remains valid in the presence of additional quantifiers. Hence, in order to prove that a query q is not definable in $\mathrm{IFP}(\mathbf{Q}_n)$, it suffices to show that q is not definable in $\mathscr{L}^\omega_{\infty\omega}(\mathbf{Q}_n)$.

The paper is organized as follows. We start in Section 2 by recalling some basic notions concerning logics and queries, and discussing the problem of capturing PTIME by a logic. In Sections 3 and 4, we give precise definitions for generalized quantifiers and logics of the form $\mathrm{FO}(\mathbf{Q})$, $\mathrm{IFP}(\mathbf{Q})$ and $\mathscr{L}^k_{\infty\omega}(\mathbf{Q})$. Section 5 is devoted to the Ehrenfeucht–Fraïssé games characterizing equivalence with respect to n-ary quantifiers. In Section 6, we introduce certain "building block" structures which we then use in the next two sections for constructing non-isomorphic structures which cannot be distinguished by n-ary quantifiers. Sections 7 and 8 contain our hierarchy theorems for LFP and PTIME, respectively. Finally, we conclude the paper in Section 9 by considering some problems that are left open in this paper.

Our notation is fairly standard. For example, if A is a set, its cardinality is denoted by $|A|$. The domain of a function f is denoted by $\mathrm{dom}(f)$, and the range of f is denoted by $\mathrm{rng}(f)$. Furthermore, we write $f \upharpoonright A$ for the restriction of f to A. The set of natural numbers is denoted by ω.

We will occasionally identify tuples with the sets of their components. For example, if f is a function and $\mathbf{a} = (a_1, ..., a_n)$, we can write $f \upharpoonright \mathbf{a}$ instead of the correct notation $f \upharpoonright \{a_1, ..., a_n\}$. Also, we use the shorthand notation $f\mathbf{a}$ for the tuple $(f(a_1), ..., f(a_n))$. The concatenation of tuples $\mathbf{a}$ and $\mathbf{b}$ is denoted by $\mathbf{a}^\frown \mathbf{b}$.

2. BACKGROUND

In this paper, all vocabularies are assumed to be relational and finite. Thus, a vocabulary τ is a finite sequence $\langle R_1, ..., R_k \rangle$ of relation symbols, where each R_i has a fixed arity denoted by $\mathrm{ar}(R_i)$. A τ-structure $\mathbf{A} = \langle A, R_1^{\mathbf{A}}, ..., R_k^{\mathbf{A}} \rangle$ consists of a non-empty set A, the *universe* of $\mathbf{A}$, and relations $R_i^{\mathbf{A}} \subseteq A^{n_i}$, where $n_i = \mathrm{ar}(R_i)$ for $1 \leqslant i \leqslant k$. Since we are mainly concerned with finite model theory, structures are assumed to be finite, unless otherwise stated.

A *logic* (*on finite structures*) $\mathscr{L}$ consists of a mapping that assigns a set $\mathscr{L}[\tau]$ of formulas to each vocabulary τ, and *satisfaction relation* $\models_{\mathscr{L}}$ between (finite) structures (with interpretations of possible free variables) and formulas. For a formal definition of the notion of logic we refer to the article of Ebbinghaus (1985); the notion of logic on finite structures was introduced by Kolaitis and Väänänen (1995). We say that a logic is concrete if it is defined via explicit rules for formula formation and matching rules for semantics. Most of the logics we are going to study in this paper are concrete.

Let $\mathbf{A}$ be a τ-structure and $\varphi(\mathbf{x})$, $\mathbf{x} = (x_1, ..., x_n)$, a formula in $\mathscr{L}[\tau]$. We write $\langle \mathbf{A}, \mathbf{a} \rangle \models_{\mathscr{L}} \varphi(\mathbf{x})$ if $\mathbf{a} = (a_1, ..., a_n)$ is a tuple of elements of A such that $\mathbf{A}$ satisfies $\varphi(\mathbf{x})$ for the interpretation assigning the element a_i for the variable x_i for each $1 \leqslant i \leqslant n$. The subscript $\mathscr{L}$ is usually clear from the context and hence omitted. As usually, displaying a formula in the form $\varphi = \varphi(\mathbf{x})$ means that all variables having free occurrences in φ are in the tuple $\mathbf{x}$, and the variables in $\mathbf{x}$ are distinct. Formulas with no free variables are called *sentences*.

If $\varphi(\mathbf{x}, \mathbf{y})$ is a formula in $\mathscr{L}[\tau]$ with $\mathbf{x} = (x_1, ..., x_n)$ and $\mathbf{y} = (y_1, ..., y_m)$, $\mathbf{A}$ is a τ-structure, and $\mathbf{a} = (a_1, ..., a_n)$ is a tuple of elements of A, then we denote by $\varphi^{\mathbf{A}}(\mathbf{a}, \cdot)$ the *relation defined by* $\varphi(\mathbf{x}, \mathbf{y})$ *with parameters* $\mathbf{a}$ *in* $\mathbf{A}$, i.e.,

$$\varphi^{\mathbf{A}}(\mathbf{a}, \cdot) = \{ \mathbf{b} \in A^m \mid \langle \mathbf{A}, \mathbf{a}, \mathbf{b} \rangle \models \varphi(\mathbf{x}, \mathbf{y}) \}.$$

If there are no parameters (i.e., $n = 0$) we write just $\varphi^{\mathbf{A}}$ instead of $\varphi^{\mathbf{A}}(\cdot)$.

Let k be a positive integer and τ a vocabulary. Any isomorphism preserving mapping q that associates a k-ary relation $q(\mathbf{A}) \subseteq A^k$ for each τ-structure $\mathbf{A}$ is called a *k-ary query* on τ-structures. Here q is isomorphism preserving if it satisfies the condition:

- if $f: \mathbf{A} \to \mathbf{B}$ is an isomorphism, then f is also an isomorphism $\langle A, q(\mathbf{A}) \rangle \to \langle B, q(\mathbf{B}) \rangle$.

A *Boolean query* on τ-structures is a mapping q from τ-structures to the set $\{0, 1\}$ such that $q(\mathbf{A}) = q(\mathbf{B})$ whenever $\mathbf{A}$ and $\mathbf{B}$ are isomorphic; we shall often identify q with the class $\{\mathbf{A} \mid q(\mathbf{A}) = 1\}$ of τ-structures.

We say that a k-ary query q on τ-structures is definable in a logic $\mathscr{L}$ if there is a formula $\varphi(x_1, ..., x_k) \in \mathscr{L}[\tau]$ such that $q(\mathbf{A}) = \varphi^{\mathbf{A}}$ for every τ-structure $\mathbf{A}$. Similarly, a Boolean query q on τ-structures is definable in $\mathscr{L}$ if there is a sentence $\varphi \in \mathscr{L}[\tau]$ such that $q(\mathbf{A}) = 1$ if and only if $\mathbf{A} \models \varphi$. Note that the satisfaction relation of any logic $\mathscr{L}$ is invariant under isomorphisms, whence the mapping $q(\mathbf{A}) = \varphi^{\mathbf{A}}$ is a query for every formula φ.

Let $\mathbf{A}$ and $\mathbf{B}$ be structures of a common vocabulary τ and $\mathscr{L}$ a logic. We say that $\mathbf{A}$ and $\mathbf{B}$ are *$\mathscr{L}$-equivalent*, in symbols $\mathbf{A} \equiv \mathbf{B}(\mathscr{L})$, if they satisfy exactly the same sentences of $\mathscr{L}[\tau]$. In other words, $\mathbf{A} \equiv \mathbf{B}(\mathscr{L})$ if and only if $q(\mathbf{A}) = q(\mathbf{B})$ for every Boolean query q on τ-structures that is definable in $\mathscr{L}$. Furthermore, we write $\langle \mathbf{A}, \mathbf{a} \rangle \equiv \langle \mathbf{B}, \mathbf{b} \rangle (\mathscr{L})$ if $\mathbf{a} \in A^k$, $\mathbf{b} \in B^k$ and $\mathbf{a} \in \varphi^{\mathbf{A}} \Leftrightarrow \mathbf{b} \in \varphi^{\mathbf{B}}$ for every formula $\varphi \in \mathscr{L}[\tau]$.

If $\mathscr{L}$ and $\mathscr{L}'$ are logics, their expressive powers on the class $\mathscr{F}$ of all finite structures can be compared in the following standard way: $\mathscr{L}'$ is (*semantically*) *at least as strong as* $\mathscr{L}$, $\mathscr{L} \leqslant_{\mathscr{F}} \mathscr{L}'$, if every query definable in $\mathscr{L}$ is also definable in $\mathscr{L}'$. The logics $\mathscr{L}$ and $\mathscr{L}'$ are (*semantically*) *equivalent*, $\mathscr{L} \equiv_{\mathscr{F}} \mathscr{L}'$, if both $\mathscr{L} \leqslant_{\mathscr{F}} \mathscr{L}'$ and $\mathscr{L}' \leqslant_{\mathscr{F}} \mathscr{L}$.

More generally, we can compare the expressive powers of logics on restricted classes of finite structures. If $\mathscr{K} \subseteq \mathscr{F}$ and q is a query on τ-structures, we denote by $q \upharpoonright \mathscr{K}$ the restriction of q to the class of those τ-structures which are in $\mathscr{K}$. We write $\mathscr{L} \leqslant_{\mathscr{K}} \mathscr{L}'$ if for every query q definable in $\mathscr{L}$ there is a query q' definable in $\mathscr{L}'$ such that $q \upharpoonright \mathscr{K} = q' \upharpoonright \mathscr{K}$, and $\mathscr{L} \equiv_{\mathscr{K}} \mathscr{L}'$ if both $\mathscr{L} \leqslant_{\mathscr{K}} \mathscr{L}'$ and $\mathscr{L}' \leqslant_{\mathscr{K}} \mathscr{L}$. The case in which $\mathscr{K}$ is the class $\mathscr{O}$ of ordered finite structures is of particular interest. Here we say that a finite structure $\mathbf{A}$ of vocabulary $\langle R_1, ..., R_k \rangle$ is *ordered* if $R_1^{\mathbf{A}}$ is a linear order of the universe A of $\mathbf{A}$.

Thus, from semantical point of view, a logic $\mathscr{L}$ on finite structures can be identified with the class of queries definable in $\mathscr{L}$. Similarly, we will also identify complexity classes with corresponding query languages. This makes it is possible to compare the expressive power of a logic and of a complexity class to each other. For example, the statement $\mathscr{L} \leqslant_{\mathscr{F}}$ PTIME means that all queries definable in $\mathscr{L}$ are computable by polynomial time Turing machines.

Let COMP be a complexity class, $\mathscr{L}$ a logic and $\mathscr{K}$ a class of finite structures. If $\mathscr{L} \equiv_{\mathscr{K}}$ COMP, we say that $\mathscr{L}$ *captures* COMP on $\mathscr{K}$; in the case $\mathscr{K} = \mathscr{F}$, we just say that $\mathscr{L}$ captures COMP.

Capturing complexity classes by various logics is the central theme of descriptive complexity theory. A canonical, and historically the first, example of such a result is Fagin's theorem, according to which NP is captured by existential second order logic:

2.1. THEOREM (Fagin, 1974). $\Sigma_1^1 \equiv_{\mathscr{F}}$ NP.

After this pioneering result of Fagin, practically all major complexity classes have been given characterizations in terms of expressibility in more or less natural logics. However, all known logical characterizations of complexity classes below NP require the presence of linear order; i.e., they hold on the class $\mathscr{O}$ of ordered finite structures, but fail on the class $\mathscr{F}$ of all finite structures. In particular, this is true for the theorem of Immerman (1986) and Vardi (1982) stating that PTIME is captured by fixpoint logic.

One of the most intriguing open problems in finite model theory is whether there exists a reasonable logic capturing PTIME on the class of all finite structures. This question was raised by Chandra and Harel (1982) in the form "Does there exist an effective enumeration of the PTIME computable queries?," and it has been widely discussed in Gurevich (1988) and Immerman (1986), among others.

It should be remarked that the problem of existence of a logic characterizing PTIME, as stated above, is vague, unless the word "reasonable" is given some precise meaning. If any abstract logic on finite structures is considered to be reasonable, then it is trivial to give a positive solution to the problem. Indeed, any collection C of queries gives rise to a logic $\mathscr{L}$ on finite structures in the following way: For each vocabulary τ, let $\mathscr{L}[\tau]$ be the set of all expressions $q(\mathbf{x})$ (q, respectively), where $q \in \mathrm{C}$ is a k-ary query (Boolean query, respectively) on τ-structures and $\mathbf{x}$ is a k-tuple of distinct variables. The satisfaction relation is then given by the rules: $\langle \mathbf{A}, \mathbf{a} \rangle \models q(\mathbf{x}) \Leftrightarrow \mathbf{a} \in q(\mathbf{A})$ for each k-ary $q \in \mathrm{C}$, and $\mathbf{A} \models q \Leftrightarrow q(\mathbf{A}) = 1$ for Boolean $q \in \mathrm{C}$. It is now obvious that $\mathscr{L} \equiv_{\mathscr{F}} \mathrm{C}$. Since PTIME is closed under Boolean operations, first-order quantification and compositions, in the case $\mathrm{C} = \mathrm{PTIME}$ we could even close $\mathscr{L}$ with respect to the usual formula formation rules of first-order logic in order to make it look more like a real logic.

However, it is clear that the logic $\mathscr{L}$ constructed as above for $\mathrm{C} = \mathrm{PTIME}$ is not reasonable in the intuitive meaning of the word, because we do not know any effective description for its syntax and semantics. Indeed, any reasonable logic capturing a complexity class should be effective in the sense that it satisfies the following two requirements:

(1) For each vocabulary τ, the set $\mathscr{L}[\tau]$ of formulas is recursive.

(2) For each vocabulary τ, there is a Turing machine M which, given any $\mathscr{L}[\tau]$-formula φ and any τ-structure $\mathbf{A}$ as input, computes the relation $\varphi^{\mathbf{A}}$.

We say that $\mathscr{L}$ is *computable*, if it satisfies these effectivity conditions.

It can still be argued that being computable is not enough for a logic $\mathscr{L}$ capturing some complexity class COMP to be reasonable. While condition (2) implies that, for any formula φ of $\mathscr{L}$, we can effectively find a Turing machine which computes the query defined by φ, it does not necessarily give us Turing machines witnessing that all queries definable in $\mathscr{L}$ are computable in COMP. In fact, Gurevich (1988) proposed a stronger interpretation for the phrase "$\mathscr{L}$ is a reasonable logic capturing PTIME" which amounts to requiring, in addition to computability, the following strengthening of condition (2):

(3) For each vocabulary τ, there is a Turing machine M which, given any $\mathscr{L}[\tau]$-formula φ as input, outputs another Turing machine M_φ and a polynomial P such that M_φ computes the query $q(\mathbf{A}) = \varphi^{\mathbf{A}}$ on τ-structures in time bounded by $P(|A|)$.

Accordingly, we say that $\mathscr{L}$ is a *Gurevich logic* if it is computable and satisfies the condition (3) above.

The problem of existence of a reasonable logic capturing PTIME can now be restated as a precise mathematical question:

2.2. QUESTION. *Does there exist a Gurevich logic capturing* PTIME?

Gurevich (1988) conjectured that the answer to this question is "no."

In the present paper we study the possibility of solving Question 2.2 in terms of generalized quantifiers. As a partial result towards Gurevich's conjecture we prove that there does not exist a finite set $\mathbf{Q}$ of generalized quantifiers such that first-order logic, or even fixpoint logic, extended by the quantifiers in $\mathbf{Q}$ would capture PTIME.

It should be remarked that proving Gurevich's conjecture would imply a major breakthrough in complexity theory: If there is a polynomial time algorithm for canonizing finite structures (i.e., an algorithm that given a structure $\mathbf{A}$ returns a linear order $<$ of A such that the isomorphism type of $\langle \mathbf{A}, < \rangle$ depends only on the isomorphism type of $\mathbf{A}$), then combining this algorithm with least fixpoint logic yields a Gurevich logic for PTIME. Hence, a negative answer to Question 2.2 would imply that PTIME is not equal to NP.

3. GENERALIZED QUANTIFIERS

In this section, we will review the notion of generalized quantifier in the sense of Lindström (1966). Since our results concern descriptive complexity theory, we will pay particular attention to quantifiers expressing computable properties of finite structures.

Generalized quantifiers provide a minimal way of extending the expressive power of logics. For example, if q is a Boolean query which is not definable in first-order logic, FO, then the easiest way of making q definable is to add the associated generalized quantifier Q_q to FO. The logic $\mathrm{FO}(Q_q)$ obtained this way is the least extension of FO in which q is definable and which is closed under the usual first-order operations and under substituting formulas for relation symbols.

3.1. DEFINITION. *The syntax of* $\mathrm{FO}(Q_q)$. Assume that $\tau = \langle R_1, ..., R_k \rangle$ is a vocabulary where $\mathrm{ar}(R_i) = n_i$ for $1 \leqslant i \leqslant k$, and q is a Boolean query on τ-structures. For each vocabulary σ, $\mathrm{FO}(Q_q)[\sigma]$ is the smallest set that contains all atomic σ-formulas and is closed under negations $\neg\varphi$, disjunctions $\varphi \vee \psi$, conjunctions $\varphi \wedge \psi$, existential quantification $\exists x\varphi$, universal quantification $\forall x\varphi$, and the additional rule

- if $\psi_1, ..., \psi_k \in \mathrm{FO}(Q_q)[\sigma]$ and $\mathbf{x}_i$ is an n_i-tuple of distinct variables for each $1 \leqslant i \leqslant k$, then $Q_q \mathbf{x}_1, ..., \mathbf{x}_k(\psi_1, ..., \psi_k) \in \mathrm{FO}(Q_q)[\sigma]$.

Thus, from the syntactical point of view the quantifier Q_q is an operator that binds k formulas together, and n_i variables in the ith formula. The free and bound variables of $\varphi = Q_q \mathbf{x}_1, ..., \mathbf{x}_k(\psi_1, ..., \psi_k)$ are defined in the natural way: if x is in the tuple $\mathbf{x}_i$, then all free occurrences of it in ψ_i are bound by $Q_q \mathbf{x}_1, ..., \mathbf{x}_k$. Note that x can nevertheless remain a free variable in φ, since the other formulas $\psi_j, j \neq i$, are not in the scope of the quantification over $\mathbf{x}_i$. For example, if $k = 2$ and $n_1 = n_2 = 1$, then both x and y are free variables of the formula $Q_q x, y(R(x, y), S(x, y))$: the occurrence of x in $R(x, y)$ is bound, but the occurrence in $S(x, y)$ is free (and similarly for y).

3.2. DEFINITION. *The semantics of* $\mathrm{FO}(Q_q)$. The satisfaction relation between σ-structures and $\mathrm{FO}(Q_q)[\sigma]$-formulas is defined in the usual way with the following special clause for the quantifier Q_q

- Let $\varphi(\mathbf{x}) = Q_q \mathbf{y}_1, ..., \mathbf{y}_k(\psi_1(\mathbf{x}_1, \mathbf{y}_1), ..., \psi_k(\mathbf{x}_k, \mathbf{y}_k))$. Then $\langle \mathbf{A}, \mathbf{a} \rangle \models \varphi(\mathbf{x})$ if and only if $q(\mathbf{B}) = 1$, where $\mathbf{B} = \langle A, \psi_1^{\mathbf{A}}(\mathbf{a}_1, \cdot), ..., \psi_k^{\mathbf{A}}(\mathbf{a}_k, \cdot) \rangle$.

Here $\mathbf{x}$ is a (non-repeating) list of all variables in the tuples $\mathbf{x}_1, ..., \mathbf{x}_k$ and $\mathbf{a}$ is the corresponding list of the parameters in $\mathbf{a}_1, ..., \mathbf{a}_k$.

Thinking in computational terms, generalized quantifiers are used like oracles: to evaluate the formula $Q_q \mathbf{y}_1, ..., \mathbf{y}_k(\psi_1, ..., \psi_k)$ one first computes the relations defined by the formulas ψ_i, and then asks the oracle whether the structure formed from these relations is in the query q. The analogy between generalized quantifiers and oracles was first studied by Grädel (1990). Makowsky and Pnueli (1994, 1995) proved that, in the presence of linear order and with a suitable choice of the model of oracle computation, this analogy can be turned to an exact correspondence between logics with generalized quantifiers and complexity classes with oracles. Quite recently Gottlob (1995) characterized those complexity classes COMP for which $\mathrm{FO}(\mathbf{Q})$ captures LOGSPACE with oracles in COMP (under the usual Ladner–Lynch model of oracle computation) whenever all problems in COMP are reducible to quantifiers in $\mathbf{Q}$ via first-order formulas.

We can also define the extension of FO by a set $\mathbf{Q}$ of generalized quantifiers, $\mathrm{FO}(\mathbf{Q})$, just by adding the formula formation rules and semantic rules for each quantifier $Q \in \mathbf{Q}$ simultaneously to the rules of first-order logic. More generally, if $\mathscr{L}$ is a concrete logic such that the rules for formula formation do not have any restrictions, then logics of the form $\mathscr{L}(Q)$ and $\mathscr{L}(\mathbf{Q})$ can usually be defined in the same way without any problems. (Restrictions in syntax may cause problems; see the discussion on $\mathrm{LFP}(Q)$ in the next section.)

If $Q = Q_q$ and $Q' = Q_{q'}$ are generalized quantifiers, it is natural to compare their expressive powers. We say that Q is *definable* by Q' if the defining class q of Q is definable in the logic $\mathrm{FO}(Q')$, i.e., if there is an $\mathrm{FO}(Q')[\tau]$-sentence φ such that $q = \{\mathbf{A} \mid \mathbf{A} \models \varphi\}$. If this is the case, then actually $\mathrm{FO}(Q) \leqslant_{\mathscr{F}} \mathrm{FO}(Q')$. Indeed, if $\theta = Q_q \mathbf{y}_1, ..., \mathbf{y}_k(\psi_1, ..., \psi_k)$ is a formula of $\mathrm{FO}(Q)$, then the formula obtained from φ by substituting the formulas ψ_i for the corresponding relation symbols in the vocabulary of q is equivalent to θ and contains one occurrence of Q_q less. Hence, by simple induction we obtain a formula of $\mathrm{FO}(Q')$ defining the same query as θ.

Quantifiers can also be classified purely syntactically according to their variable binding pattern. Let Q_q be a generalized quantifier, where q is a Boolean query on τ-structures. We say that Q_q is of type $\langle n_1, ..., n_k \rangle$ if $\tau = \langle R_1, ..., R_k \rangle$ and $\mathrm{ar}(R_i) = n_i$ for $1 \leqslant i \leqslant k$. The arity of Q_q is $\mathrm{ar}(Q_q) = \max\{n_1, ..., n_k\}$, and we say that Q_q is n-ary if $\mathrm{ar}(Q_q) \leqslant n$. Furthermore, Q_q is simple if $k = 1$, i.e., if Q_q binds only one formula. For each natural number n, we reserve the special notation $\mathbf{Q}_n$ for the set of all n-ary quantifiers on finite structures.

We give below some examples of quantifiers expressing natural computable properties of finite structures. Many of these quantifiers have already been studied in the literature.

3.3. EXAMPLES. (a) *Counting quantifiers.* For each natural number m, let q_m be the class of all finite structures $\langle A, P \rangle$ such that $P \subseteq A$ has at least m elements. The counting quantifier Q_{q_m} is usually written more intuitively as $\exists^{\geqslant m}$. These quantifiers may appear trivial since they are easily definable in first-order logic. However, the first-order definition of q_m requires quantification over at least m distinct variables, whereas $\exists^{\geqslant m}$ binds only one variable. This makes a dramatic difference in contexts where the number of distinct variables is restricted.

(b) *Härtig and Rescher quantifiers.* Let I be the quantifier which is determined by the class q of all structures $\langle A, P, S \rangle$ such that $P, S \subseteq A$ and $|P| = |S|$. Then I is a quantifier of type $\langle 1, 1 \rangle$, and it is called the Härtig quantifier. The Rescher quantifier R is defined similarly, except that the condition $|P| = |S|$ is replaced with $|P| \leqslant |S|$.

(c) *Graph quantifiers.* Any property of finite graphs gives rise to a simple binary quantifier. For example, the PTIME computable property of being connected and the NP-complete property of being 3-colorable are captured by the quantifiers Q_q and $Q_{q'}$, where q is simply the class of all finite connected graphs and q' is the class of all finite 3-colorable graphs, respectively.

(d) *Ramsey quantifiers*. For each function $f\colon \omega \to \omega$, let Q_f be the quantifier defined by the class of all finite structures $\langle A, P\rangle$ with $P \subseteq A$ and $|P| \geqslant f(|A|)$. The nth Ramseyfication of Q_f, denoted by Q_f^n, is obtained by changing the arity of P from 1 to n, and requiring that there is a homogeneous set $S \subseteq A$ for P with at least $f(|A|)$ elements; i.e., $\mathbf{A} \models Q_f^n x_1, ..., x_n R(x_1, ..., x_n)$ if and only if there exists $S \subseteq A$ such that $|S| \geqslant f(|A|)$ and $(a_1, ..., a_n) \in R^{\mathbf{A}}$ for any distinct elements $a_1, ..., a_n \in S$.

(e) *Henkin quantifier*. The Henkin quantifier H corresponds to the simplest non-trivial partially ordered prefix

$$\begin{Bmatrix} \forall x & \exists y \\ \forall u & \exists v \end{Bmatrix}$$

of first-order quantifiers. Thus, the interpretation of H is the class q of all structures $\langle A, R\rangle$ such that $R \subseteq A^4$ and there exist functions f, $g\colon A \to A$ such that $(a, f(a), b, g(b)) \in R$ for all $a, b \in A$. More generally, for all natural numbers $n \geqslant 1$ and $k \geqslant 2$, we denote by H_k^n the quantifier arising from the partially ordered prefix with k rows and n universal quantifiers preceding one existential quantifier on each row. Thus, H_k^n is the simple $(n+1)\,k$-ary quantifier defined by the class of all structures $\langle A, R\rangle$ with $R \subseteq A^{(n+1)k}$ and satisfying the condition: there are functions $f_1, ..., f_k\colon A^n \to A$ such that $(\mathbf{a}_1, f_1(\mathbf{a}_1), ..., \mathbf{a}_k, f_k(\mathbf{a}_k)) \in R$ for all $\mathbf{a}_1, ..., \mathbf{a}_k \in A^n$. We use the notation $\mathbf{H}$ for the set of all Henkin quantifiers.

Partially ordered prefixes were introduced by Henkin (1961), and later Walkoe (1970) proved that they are all definable in the logic $\mathrm{FO}(\mathbf{H})$. Blass and Gurevich (1986) studied computational aspects of Henkin quantifiers and so-called *narrow Henkin quantifiers* which are obtained by replacing (some of) the existential quantifications over the universe with existential quantification over a Boolean variable. They proved that all these quantifiers, except the narrow Henkin quantifiers with only two rows, are capable of expressing NP-complete problems. Moreover, as Blass and Gurevich pointed out, the fragment of $\mathrm{FO}(\mathbf{H})$ consisting of sentences of the form $H_k^n \mathbf{x}\, \varphi(\mathbf{x})$, where $\varphi(\mathbf{x})$ is quantifier-free, captures exactly the class of NP-computable (Boolean) queries. This is a direct consequence of Fagin's Theorem 2.1 and the result of Walkoe (1970) that the fragment defined above has the same expressive power as Σ_1^1 on the class of all (finite and infinite) structures.

Generalized quantifiers give us a direct way of making Boolean queries definable in a small extension of a given logic, but what about k-ary queries for $k \geqslant 1$? In principle it is possible to extend the notion of quantifier to the case of non-Boolean queries. In fact, all the transitive closure operators DTC (deterministic transitive closure), TC (transitive closure), and ATC (alternating transitive closure) introduced by Immerman (1987) can be seen as instances of such an extended definition.

However, restricting to Boolean queries in the definition of quantifier does not cause any loss of generality. If q is a k-ary query on τ-structures, then q is definable in the logic $\mathrm{FO}(Q_{q'})$, where q' is the Boolean query on $\tau^\frown\langle X_1, ..., X_k\rangle$-structures, $\mathrm{ar}(X_1) = \cdots = \mathrm{ar}(X_k) = 1$, such that $q'(\langle \mathbf{A}, S_1, ..., S_k\rangle) = 1$ if and only if $S_1 \times \cdots \times S_k \subseteq q(\mathbf{A})$.[1] On the other hand, q' is clearly definable in any logic which is able to define q and is closed under first-order operations. In particular, the extension of first-order logic by the various transitive closure operators can also be defined in terms of generalized quantifiers.

Note also that the arity of the quantifier $Q_{q'}$ above does not depend on k; it is always equal to the maximum arity of relation symbols in τ. Hence, if all relation symbols in τ are at most n-ary, then all queries on τ-structures are definable in $\mathrm{FO}(\mathbf{Q}_n)$.

4. ADDING QUANTIFIERS TO FIXPOINT LOGICS

Let $\tau = \langle R_1, ..., R_k\rangle$ and $\sigma = \langle R_1, ..., R_k, X\rangle$ be vocabularies, and let $\varphi = \varphi(X, \mathbf{x}, \mathbf{y})$ be a formula of a vocabulary σ, where $\mathbf{x} = (x_1, ..., x_m)$, $\mathbf{y} = (y_1, ..., y_n)$ and X is n-ary. For each τ-structure $\mathbf{A}$ and each tuple $\mathbf{a} \in A^m$ of parameters, φ defines an operation $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}$ in the set of n-ary relations of A:

- $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}(S) = \{\mathbf{b} \in A^n \mid \langle \mathbf{A}, S, \mathbf{a}, \mathbf{b}\rangle \models \varphi(X, \mathbf{x}, \mathbf{y})\}$.

A relation $S \subseteq A^n$ is a *fixpoint* of the operation $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}$ if $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}(S) = S$. If in addition $S \subseteq T$ for all other fixpoints of $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}$, then we say that S is the *least fixpoint* of $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}$.

A canonical way of looking for possible fixpoints of $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}$ is to iterate it starting from the empty relation. Thus, we define the *stages* Γ^i, $i \in \omega$, of $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}$ inductively as follows:

- $\Gamma^0 = \varnothing$,
- $\Gamma^{i+1} = \Gamma_\varphi^{\mathbf{A}, \mathbf{a}}(\Gamma^i)$.

Since, by our general assumption, the structure $\mathbf{A}$ is finite, either there is some j for which Γ^j is a fixpoint of $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}$, or else the sequence $\Gamma^0, \Gamma^1, ...$ of stages is periodic: there exist j and $p > 1$ such that $\Gamma^i = \Gamma^{i+p}$ for all $i \geqslant j$. In the former case we say that Γ^j is the *inductive fixpoint* of $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}$, and we denote it by Γ^∞.

The operation $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}$ is *monotone* if $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}(S) \subseteq \Gamma_\varphi^{\mathbf{A}, \mathbf{a}}(T)$ whenever $S \subseteq T \subseteq A^n$, and it is *inflationary* if $S \subseteq \Gamma_\varphi^{\mathbf{A}, \mathbf{a}}(S)$ for all $S \subseteq A^n$. It is easy to see that if $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}$ is monotone or inflationary, then the sequence of stages is increasing: $\Gamma^i \subseteq \Gamma^{i+1}$ for all $i \in \omega$. Consequently, the inductive fixpoint Γ^∞ exists for all monotone and inflationary operations $\Gamma_\varphi^{\mathbf{A}, \mathbf{a}}$, and in fact, $\Gamma^\infty = \Gamma^j$ for some $j \leqslant |A|^n$. Moreover, if

[1] This trick is due to Kerkko Luosto.

$\Gamma_{\varphi}^{\mathbf{A},\mathbf{a}}$ is monotone, then Γ^{∞} is the least fixpoint of $\Gamma_{\varphi}^{\mathbf{A},\mathbf{a}}$; this is not true in general for inflationary operations $\Gamma_{\varphi}^{\mathbf{A},\mathbf{a}}$.

It is an undecidable problem to tell if a first-order formula φ has the property that $\Gamma_{\varphi}^{\mathbf{A},\mathbf{a}}$ is monotone for every $\mathbf{A}$ and $\mathbf{a}$. But fortunately there is a simple syntactic condition which guarantees monotonicity. An occurrence of the relation symbol X in the formula φ is *positive* if it is in the scope of an even number of negations, and φ is *positive in* X if every occurrence of it in φ is positive. *Least fixpoint logic*, LFP, is the extension of first-order logic obtained by closing under least fixpoints of positive formulas.

4.1. Definition. *The syntax of* LFP. The sets LFP$[\tau]$ of formulas are defined by simultaneous induction for all vocabularies τ. First of all, each LFP$[\tau]$ contains all atomic τ-formulas. Secondly, each LFP$[\tau]$ is closed under negations $\neg\varphi$, disjunctions $\varphi \vee \psi$, conjunctions $\varphi \wedge \psi$, existential quantification $\exists x\varphi$, and universal quantification $\forall x\varphi$. Finally, for $\tau = \langle R_1, ..., R_k \rangle$ and $\sigma = \langle R_1, ..., R_k, X \rangle$ with $\mathrm{ar}(X) = n$, we have the additional rule

- if $\varphi \in \mathrm{LFP}[\sigma]$ is positive in X, and $\mathbf{y}$ and $\mathbf{z}$ are n-tuples of variables, then $(\mathsf{LFP}_{X,\mathbf{y}}\,\varphi)[\mathbf{z}] \in \mathrm{LFP}[\tau]$.

Here the fixpoint operator $\mathsf{LFP}_{X,\mathbf{y}}$ binds the variables in $\mathbf{y}$ (and the relation symbol X), whereas the occurrences of variables in the tuple $\mathbf{z}$ are free.

4.2. Definition. The semantics of LFP. The satisfaction relation between τ-structures and LFP$[\tau]$-formulas is defined as usually with the following special rule for the fixpoint operator LFP

- $\langle \mathbf{A}, \mathbf{a}, \mathbf{b} \rangle \models (\mathsf{LFP}_{X,\mathbf{y}}\,\varphi(X, \mathbf{x}, \mathbf{y}))[\mathbf{z}]$ if and only if the tuple $\mathbf{b} \in A^n$ is in the least fixpoint Γ^{∞} of $\Gamma_{\varphi}^{\mathbf{A},\mathbf{a}}$.

From the point of view of descriptive complexity theory, least fixpoint logic is one of the most successful extensions of first-order logic introduced so far. Indeed, by the well-known result due to Immerman and Vardi, LFP captures PTIME on ordered finite structures.

4.3. Theorem (Immerman, 1986; Vardi, 1982). LFP $\equiv_{\mathscr{O}}$ PTIME.

On the other hand, it is well known that LFP falls badly short of capturing PTIME on the class of all finite structures. For example, the query q_{e} defined by

$$q_{\mathrm{e}}(\mathbf{A}) = 1 \Leftrightarrow \mathbf{A} \text{ has an even number of elements},$$

is not definable in LFP, as observed by Chandra and Harel (1982). A simple solution to this problem would be to add the generalized quantifier Q_{q_e} to fixpoint logic. More generally, we would like to consider extensions of fixpoint logic by other PTIME computable quantifiers in search for a solution to Question 2.2.

However, the requirement of positivity in the crucial formula formation rule causes a problem for the definition of LFP($\mathbf{Q}$). While the notion of positive occurrences of relation symbols extends without changes to formulas containing quantifiers $Q_q \in \mathbf{Q}$, positivity does not anymore imply monotonicity, unless all the quantifiers $Q_q \in \mathbf{Q}$ are monotone. Here we say that a quantifier Q_q of type $\langle n_1, ..., n_k \rangle$ is *monotone* if $\langle A, R_1, ..., R_k \rangle \in q$ and $R_i \subseteq S_i \subseteq A^{n_i}$, for $1 \leqslant i \leqslant k$, implies $\langle A, S_1, ..., S_k \rangle \in q$.

In the case of monotone quantifiers there are no difficulties in defining LFP($\mathbf{Q}$) (see Kolaitis and Väänänen, 1995, Sect. 2.4). However, in order to avoid a loss of generality, we do not want to restrict our attention to monotone quantifiers. For our purposes, a better solution is to replace LFP with another variant of fixpoint logic as a base logic. *Inflationary fixpoint logic*, IFP, is defined in the same way as LFP, except that the syntactical and semantical rules concerning the fixpoint operator LFP are replaced with the following rules:

- if $\varphi \in \mathrm{IFP}[\tau^\frown\langle X \rangle]$, where $\mathrm{ar}(X) = n$, and $\mathbf{y}$ and $\mathbf{z}$ are n-tuples of variables, then $(\mathsf{IFP}_{X,\mathbf{y}}\,\varphi)[\mathbf{z}] \in \mathrm{IFP}[\tau]$; and
- $\langle \mathbf{A}, \mathbf{a}, \mathbf{b} \rangle \models (\mathsf{IFP}_{X,\mathbf{y}}\,\varphi(X, \mathbf{x}, \mathbf{y}))[\mathbf{z}]$ if and only if the tuple $\mathbf{b} \in A^n$ is in the inductive fixpoint Γ^{∞} of $\Gamma_{\psi}^{\mathbf{A},\mathbf{a}}$, where $\psi = \varphi(X, \mathbf{x}, \mathbf{y}) \vee X(\mathbf{y})$.

Note that the operation $\Gamma_{\psi}^{\mathbf{A},\mathbf{a}}$ above is always inflationary, and hence the inductive fixpoint Γ^{∞} is guaranteed to exist. In fact, this is true even if the formula φ contains generalized quantifiers. Thus, we can define the logic IFP($\mathbf{Q}$) for any set $\mathbf{Q}$ of generalized quantifiers just by adding the formula formation rules and semantic rules of FO($\mathbf{Q}$) and IFP together.

Gurevich and Shelah (1986) proved that least fixpoint logic and inflationary fixpoint logic have exactly the same expressive power: a query is definable in LFP if and only if it is definable in IFP. In particular, all IFP-definable queries are polynomial time computable. This remains true even if we add polynomial time computable quantifiers to IFP.

4.4. Proposition. *If* $\mathbf{Q}$ *is a set of quantifiers such that the defining class* q *is in* PTIME *for every* $Q_q \in \mathbf{Q}$, *then* IFP($\mathbf{Q}$) $\leqslant_{\mathscr{F}}$ PTIME.

Proof. Note that if $\varphi = Q_q\mathbf{x}_1, ..., \mathbf{x}_k(\psi_1, ..., \psi_k)$, where $q(\mathbf{A})$ is computable in time bounded by $p(|A|)$ and, for each $1 \leqslant i \leqslant k$, $\psi_i^{\mathbf{A}}$ is computable in time bounded by $p_i(|A|)$, then $\varphi^{\mathbf{A}}$ is computable in time bounded by $p_1(|A|) + \cdots + p_k(|A|) + p(|A|)$. Thus, the claim follows by a straightforward induction just like in the case of IFP without extra quantifiers. ∎

Note that if the set $\mathbf{Q}$ in Proposition 4.4 is finite, then it is not difficult to see that IFP($\mathbf{Q}$) is in fact a Gurevich logic; the same is true also for FO($\mathbf{Q}$), of course.

A third variant of fixpoint logic that has been extensively studied in the literature is *partial fixpoint logic*, PFP. The syntax of PFP is defined as that of IFP, except that the inflationary fixpoint operator IFP is replaced with the partial fixpoint operator PFP. The corresponding semantic rule is

- $\langle \mathbf{A}, \mathbf{a}, \mathbf{b} \rangle \models (\mathsf{PFP}_{X, \mathbf{y}}\, \varphi(X, \mathbf{x}, \mathbf{y}))[\mathbf{z}]$ if and only if the inductive fixpoint Γ^∞ of $\Gamma^{\mathbf{A}, \mathbf{a}}_\varphi$ exists and $\mathbf{b} \in \Gamma^\infty$.

The sequence of stages of the operation $\Gamma^{\mathbf{A}, \mathbf{a}}_\varphi$ is not increasing in general, and so the (straightforward) computation of Γ^∞ may take exponential time. However, Γ^∞ can always be computed in polynomial space, whence all PFP-definable queries are in PSPACE. Moreover, in the presence of linear order PFP captures PSPACE.

4.5. Theorem (Abiteboul and Vianu, 1991; Vardi, 1982). $\mathrm{PFP} \equiv_{\mathscr{O}}$ PSPACE.

As in the case of IFP, there are no problems for defining the extension of PFP with a set $\mathbf{Q}$ of generalized quantifiers in the natural way. Furthermore, we have the following analogue of Proposition 4.4.

4.6. Proposition. *If* $\mathbf{Q}$ *is a set of quantifiers such that the defining class* q *is in* PSPACE *for every* $Q_q \in \mathbf{Q}$, *then* $\mathrm{PFP}(\mathbf{Q}) \leqslant_{\mathscr{F}}$ PSPACE.

Let $\mathscr{L}_{\infty\omega}$ be the usual infinitary logic which is obtained from first-order logic by allowing conjunctions and disjunctions over arbitrary sets of formulas. For each natural number k, the *k-variable logic*, $\mathscr{L}^k_{\infty\omega}$, consists of those formulas of $\mathscr{L}_{\infty\omega}$ which contain at most k distinct variables (free or bound). Without loss of expressive power we can assume that all variables occurring in the formulas of $\mathscr{L}^k_{\infty\omega}$ are among $x_1, ..., x_k$. We denote by FO^k the intersection of $\mathscr{L}^k_{\infty\omega}$ and FO; i.e., FO^k is the set of first-order formulas with variables among $x_1, ..., x_k$. The *finite variable logic*, $\mathscr{L}^\omega_{\infty\omega}$, is the union of the logics $\mathscr{L}^k_{\infty\omega}$ over k. In other words, $\mathscr{L}^\omega_{\infty\omega}$ is the restriction of $\mathscr{L}_{\infty\omega}$ to formulas with only finitely many distinct variables occurring in them. We refer to Kolaitis and Vardi (1992a) for a more detailed exposition of the logics $\mathscr{L}^\omega_{\infty\omega}$ and $\mathscr{L}^k_{\infty\omega}$.

Logics of the form $\mathrm{FO}^k(\mathbf{Q})$, $\mathscr{L}^k_{\infty\omega}(\mathbf{Q})$ and $\mathscr{L}^\omega_{\infty\omega}(\mathbf{Q})$ can now be defined without difficulties in the natural way. However, in two first cases we require that all quantifiers in $\mathbf{Q}$ are k-ary, since quantification over a tuple of more than k distinct variables does not make sense in the context of k-variable logic.

The finite variable logic $\mathscr{L}^\omega_{\infty\omega}$ has turned out to be a useful tool in analyzing the expressive power of fixpoint logics (see Kolaitis and Vardi, 1992a; Kolaitis and Vardi, 1992b; and Dawar *et al.*, 1995). This is due to two facts: First, all variants of fixpoint logic are subsumed by $\mathscr{L}^\omega_{\infty\omega}$, so that upper bounds for definability in $\mathscr{L}^\omega_{\infty\omega}$ are automatically upper bounds for definability in LFP, IFP, and PFP, too. Secondly, the expressive power of $\mathscr{L}^\omega_{\infty\omega}$ is completely characterized by so-called *pebble games* (see Kolaitis and Vardi, 1992a, Sect. 2.3). Our next aim is to show that the finite variable logic with all n-ary quantifiers, $\mathscr{L}^\omega_{\infty\omega}(\mathbf{Q}_n)$, has a similar role as a tool for analyzing the logics $\mathrm{IFP}(\mathbf{Q}_n)$ and $\mathrm{PFP}(\mathbf{Q}_n)$.

We start by noting that, for any set $\mathbf{Q}$ of quantifiers, $\mathrm{IFP}(\mathbf{Q})$ and $\mathrm{PFP}(\mathbf{Q})$ are subsumed by $\mathscr{L}^\omega_{\infty\omega}(\mathbf{Q})$. In the next section we will then give a pebble game characterization for the expressive power of the logic $\mathscr{L}^\omega_{\infty\omega}(\mathbf{Q}_n)$.

4.7. Lemma. *For any set* $\mathbf{Q}$ *of quantifiers*, $\mathrm{IFP}(\mathbf{Q}) \leqslant_{\mathscr{F}} \mathrm{PFP}(\mathbf{Q}) \leqslant_{\mathscr{F}} \mathscr{L}^\omega_{\infty\omega}(\mathbf{Q})$.

Proof. The claim $\mathrm{IFP}(\mathbf{Q}) \leqslant_{\mathscr{F}} \mathrm{PFP}(\mathbf{Q})$ is obvious by the definitions. The second claim is proved in the same way as the analogous result without extra quantifiers (see Kolaitis and Vardi, 1992a). ∎

5. EHRENFEUCHT–FRAÏSSÉ GAMES FOR *n*-ARY QUANTIFIERS

The classification of generalized quantifiers according to arity raises the question whether for each n there are $(n+1)$-ary quantifiers which are not definable in terms of n-ary quantifiers. Väänänen (1986) proved that this is indeed the case. Using the proof of Väänänen as a starting point we developed in Hella (1989) a fairly general method for proving similar results. However, all examples obtained so far are based on constructions of infinite non-isomorphic structures which cannot be distinguished with n-ary quantifiers; the infiniteness of the structures is an unavoidable feature of these constructions (see, e.g., Hella, 1989, Model Construction 3.1). In the present paper, we will show how to obtain such structures of finite cardinality.

The main tool in Hella (1989) for proving that a class of structures is not definable by n-ary quantifiers is a back-and-forth characterization of equivalence with respect to the infinitary logic $\mathscr{L}_{\infty\omega}$ extended by all n-ary quantifiers (Hella, 1989, Theorems 2.5 and 2.8). This back-and-forth characterization is valid for structures of any cardinality, whence we can use it freely in the context of finite model theory. We will state the characterization here in terms of a corresponding Ehrenfeucht–Fraïssé game (Theorem 5.3).

Recall from Section 3 that the set of all n-ary quantifiers on finite structures is denoted by $\mathbf{Q}_n$. We fix for this section a vocabulary τ and two finite τ-structures $\mathbf{A}$ and $\mathbf{B}$. Since already $\mathbf{A} \equiv \mathbf{B}(\mathrm{FO})$ implies that $\mathbf{A}$ and $\mathbf{B}$ are isomorphic, we have to consider proper fragments of $\mathscr{L}_{\infty\omega}(\mathbf{Q}_n)$ in order to get useful back-and-forth characterizations. The classical solution to this problem is to restrict the quantifier rank.

5.1. Definition. Let $\mathbf{Q}$ be a set of quantifiers. The *quantifier rank*, $qr(\varphi)$, of a formula φ of $\mathrm{FO}(\mathbf{Q})$ is defined by induction in the usual way:

- $qr(\varphi)=0$ if φ is atomic,
- $qr(\neg\varphi)=qr(\varphi)$,
- $qr(\varphi\wedge\psi)=qr(\varphi\vee\psi)=\max\{qr(\varphi), qr(\psi)\}$,
- $qr(\forall x\varphi)=qr(\exists x\varphi)=qr(\varphi)+1$,
- $qr(\varphi)=\max\{qr(\psi_1), ..., qr(\psi_k)\}+1$ if $\varphi=Q_q\mathbf{x}_1, ..., \mathbf{x}_k(\psi_1, ..., \psi_k)$ for some $Q_q\in\mathbf{Q}$.

We use the notation $\langle\mathbf{A}, \mathbf{a}\rangle\equiv_n^m\langle\mathbf{B}, \mathbf{b}\rangle$ if $\mathbf{a}\in A^k$ and $\mathbf{b}\in B^k$ are tuples such that the equivalence $\langle\mathbf{A}, \mathbf{a}\rangle\models\varphi(\mathbf{x})\Leftrightarrow\langle\mathbf{B}, \mathbf{b}\rangle\models\varphi(\mathbf{x})$ holds for every formula $\varphi(\mathbf{x})\in\mathrm{FO}(\mathbf{Q}_n)[\tau]$ with $qr(\varphi)\leqslant m$. In the case $k=0$ we write $\mathbf{A}\equiv_n^m\mathbf{B}$ rather than $\langle\mathbf{A}, \varnothing\rangle\equiv_n^m\langle\mathbf{B}, \varnothing\rangle$.

5.2. Definition. Let p be an injective function such that $\mathrm{dom}(p)\subseteq A$ and $\mathrm{rng}(p)\subseteq B$. We say that p preserves the truth of a τ-formula $\varphi(\mathbf{x})$ if, for any tuple $\mathbf{a}$ of elements of $\mathrm{dom}(p)$, $\langle\mathbf{A}, \mathbf{a}\rangle\models\varphi(\mathbf{x})$ if and only if $\langle\mathbf{B}, p\mathbf{a}\rangle\models\varphi(\mathbf{x})$. The function p is a partial isomorphism $\mathbf{A}\to\mathbf{B}$ if it preserves the truth of all atomic τ-formulas.

Note that the empty function $\varnothing$ is always a partial isomorphism, since by our assumption on vocabularies there are no atomic sentences.

Let $\mathbf{a}=(a_1, ..., a_k)\in A^k$ and $\mathbf{b}=(b_1, ..., b_k)\in B^k$. The *n-bijective Ehrenfeucht–Fraïssé game of length m* for $\langle\mathbf{A}, \mathbf{a}\rangle$ and $\langle\mathbf{B}, \mathbf{b}\rangle$ is played between two players, I and II. The game has m rounds, and in each round i player II chooses a bijection $f_i\colon A\to B$ and player I answers by choosing a subset $D_i\subseteq A$ with $|D_i|\leqslant n$. The outcome of the game is the relation

$$p=p_0\cup(f_1\restriction D_1)\cup\cdots\cup(f_m\restriction D_m),$$

where $p_0=\{(a_1, b_1), ..., (a_k, b_k)\}$. Player II wins the game if p is a partial isomorphism $\mathbf{A}\to\mathbf{B}$; player I wins if this is not the case.

We denote this game by $BEF_n^m(\mathbf{A}, \mathbf{a}, \mathbf{B}, \mathbf{b})$. If $k=0$, we write $BEF_n^m(\mathbf{A}, \mathbf{B})$ instead of $BEF_n^m(\mathbf{A}, \varnothing, \mathbf{B}, \varnothing)$.

The notion of *winning strategy* is defined in the usual way: player I (player II) has a winning strategy in the game $BEF_n^m(\mathbf{A}, \mathbf{a}, \mathbf{B}, \mathbf{b})$ if he has a systematic way of choosing the sets D_i (the bijections f_i) such that using it he is guaranteed to win. It is clear that $BEF_n^m(\mathbf{A}, \mathbf{a}, \mathbf{B}, \mathbf{b})$ is a determined game; i.e., exactly one of the players has a winning strategy.

In Hella (1989) we proved that the game BEF_n^m characterizes the equivalence of (arbitrary) structures up to quantifier rank m in the logic $\mathscr{L}_{\infty\omega}(\mathbf{Q}_n^*)$, where $\mathbf{Q}_n^*$ is the family of all n-ary quantifiers Q with no restriction on the cardinality of the type of Q. In the case of finite structures infinitary connectives and quantifers of infinite types are not actually needed (this can be seen from the proof of Theorem 2.5 in Hella, 1989). Thus, we have

5.3. Theorem. *$\langle\mathbf{A}, \mathbf{a}\rangle\equiv_n^m\langle\mathbf{B}, \mathbf{b}\rangle$ if and only if player II has a winning strategy in the game $BEF_n^m(\mathbf{A}, \mathbf{a}, \mathbf{B}, \mathbf{b})$.*

Theorem 5.3 provides us with a method for proving that a given query is not definable in the logic $\mathrm{FO}(\mathbf{Q}_n)$.

5.4. Corollary. *Let q be a Boolean query (k-ary query, respectively) on σ-structures. If for each natural number m there are σ-structures $\mathbf{C}$ and $\mathbf{D}$ (and k-tuples $\mathbf{c}$ and $\mathbf{d}$, resp.) such that*

(i) *player II has a winning strategy in the game $BEF_n^m(\mathbf{C}, \mathbf{D})$ ($BEF_n^m(\mathbf{C}, \mathbf{c}, \mathbf{D}, \mathbf{d})$, resp.),*

(ii) *$q(\mathbf{C})=1$ ($\mathbf{c}\in q(\mathbf{C})$, resp.), and*

(iii) *$q(\mathbf{D})=0$ ($\mathbf{d}\notin q(\mathbf{D})$, resp.),*

then q is not definable in $\mathrm{FO}(\mathbf{Q}_n)$.

Proof. If q is defined by a sentence $\varphi\in\mathrm{FO}(\mathbf{Q}_n)[\sigma]$ with $qr(\varphi)=m$, then $\mathbf{C}\equiv_n^m\mathbf{D}$ implies that $q(\mathbf{C})=q(\mathbf{D})$. Hence, there cannot exist structures $\mathbf{C}$ and $\mathbf{D}$ satisfying conditions (i)–(iii) above. The case of k-ary q is proved similarly. ▮

We are also going to need Ehrenfeucht–Fraïssé game characterizations for the logics $\mathscr{L}_{\infty\omega}^k(\mathbf{Q}_n)$. For this purpose we will next elaborate the back-and-forth systems of Hella (1989) by taking the bound k for the number of distinct variables into account.

We use the notation $\mathrm{Part}^k(\mathbf{A}, \mathbf{B})$ for the set of all partial isomorphisms $p\colon\mathbf{A}\to\mathbf{B}$ such that $|p|\leqslant k$. Partial isomorphisms preserving the truth of $\mathscr{L}_{\infty\omega}^k(\mathbf{Q}_n)$-formulas are of specific interest to us.

5.5. Definition. For each natural number k, let $\mathbf{J}^k(\mathbf{A}, \mathbf{B})$ be the set of all $p\in\mathrm{Part}^k(\mathbf{A}, \mathbf{B})$ which preserve the truth of all $\mathscr{L}_{\infty\omega}^k(\mathbf{Q}_n)[\tau]$-formulas. Furthermore, let $\mathbf{K}^k(\mathbf{A}, \mathbf{B})$ be the set of all $p\in\mathrm{Part}^k(\mathbf{A}, \mathbf{B})$ preserving the truth of all $\mathrm{FO}^k(\mathbf{Q}_n)[\tau]$-formulas.

It is clear that $\mathbf{J}^k(\mathbf{A}, \mathbf{B})$ is a subset of $\mathbf{K}^k(\mathbf{A}, \mathbf{B})$; we will soon show that these sets are in fact identical.

5.6. Definition. The sequence $(I_m^k(\mathbf{A}, \mathbf{B}))_{m\in\omega}$ of *canonical k-variable n-bijective back-and-forth sets* for $\mathbf{A}$ and $\mathbf{B}$ is defined recursively as follows:

(i) $I_0^k(\mathbf{A}, \mathbf{B})=\mathrm{Part}^k(\mathbf{A}, \mathbf{B})$,

(ii) $I_{m+1}^k(\mathbf{A}, \mathbf{B})$ is the set of all $p\in I_m^k(\mathbf{A}, \mathbf{B})$ for which there exists a bijection $f\colon A\to B$ such that $(p\restriction C)\cup(f\restriction D)\in I_m^k(\mathbf{A}, \mathbf{B})$ whenever $C\subseteq\mathrm{dom}(p)$, $D\subseteq A$, $|D|\leqslant n$, and $|C\cup D|\leqslant k$.

The *canonical k-variable n-bijective back-and-forth system* for $\mathbf{A}$ and $\mathbf{B}$ is $\mathbf{I}^k(\mathbf{A}, \mathbf{B})=\bigcap_{m\in\omega}I_m^k(\mathbf{A}, \mathbf{B})$.

Since the structures $\mathbf{A}$ and $\mathbf{B}$ are finite, and $I_0^k(\mathbf{A}, \mathbf{B})\supseteq I_1^k(\mathbf{A}, \mathbf{B})\supseteq\cdots$ by definition, there is an integer m such that $I_m^k(\mathbf{A}, \mathbf{B})=I_{m+1}^k(\mathbf{A}, \mathbf{B})$. Hence, actually $\mathbf{I}^k(\mathbf{A}, \mathbf{B})=I_m^k(\mathbf{A}, \mathbf{B})$ for this m, and so $\mathbf{I}^k(\mathbf{A}, \mathbf{B})$ satisfies the *bijective extension condition*:

(BE) for each $p \in \mathbf{I}^k(\mathbf{A}, \mathbf{B})$ there is a bijection $f: A \to B$ such that $(p \upharpoonright C) \cup (f \upharpoonright D) \in \mathbf{I}^k(\mathbf{A}, \mathbf{B})$ whenever $C \subseteq \operatorname{dom}(p)$, $D \subseteq A$, $|D| \leqslant n$, and $|C \cup D| \leqslant k$.

It is not difficult to see that $\mathbf{I}^k(\mathbf{A}, \mathbf{B})$ is in fact the largest subset of $\mathrm{Part}^k(\mathbf{A}, \mathbf{B})$ satisfying (BE).

5.7. Lemma. $\mathbf{I}^k(\mathbf{A}, \mathbf{B}) = \mathbf{J}^k(\mathbf{A}, \mathbf{B}) = \mathbf{K}^k(\mathbf{A}, \mathbf{B})$.

Proof. We have already observed that $\mathbf{J}^k(\mathbf{A}, \mathbf{B}) \subseteq \mathbf{K}^k(\mathbf{A}, \mathbf{B})$. Hence, it suffices to prove here the inclusions $\mathbf{I}^k(\mathbf{A}, \mathbf{B}) \subseteq \mathbf{J}^k(\mathbf{A}, \mathbf{B})$ and $\mathbf{K}^k(\mathbf{A}, \mathbf{B}) \subseteq \mathbf{I}^k(\mathbf{A}, \mathbf{B})$.

Let $\varphi(\mathbf{x})$ be an $\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)$-formula. We prove by induction on φ that every $p \in \mathbf{I}^k(\mathbf{A}, \mathbf{B})$ preserves the truth of φ. If φ is an atomic formula, the claim is true since $\mathbf{I}^k(\mathbf{A}, \mathbf{B}) \subseteq \mathrm{Part}^k(\mathbf{A}, \mathbf{B})$. The induction steps for the connectives $\neg$, $\bigwedge$ and $\bigvee$ are trivial. Consider then the case

$$\varphi(\mathbf{x}) = Q_q \mathbf{y}_1, ..., \mathbf{y}_l(\psi_1(\mathbf{x}_1, \mathbf{y}_1), ..., \psi_l(\mathbf{x}_l, \mathbf{y}_l)),$$

for some $Q_q \in \mathbf{Q}_n$ of type $\langle n_1, ..., n_l \rangle$. (The first-order quantifiers $\forall$ and $\exists$ are contained in $\mathbf{Q}_n$, whence there is no need to consider them separately.) To prove the claim for φ, assume that $p \in \mathbf{I}^k(\mathbf{A}, \mathbf{B})$ and $\mathbf{a}$ is a tuple of elements of $\operatorname{dom}(p)$. For $1 \leqslant i \leqslant l$, denote by $\mathbf{a}_i$ the subtuple of $\mathbf{a}$ corresponding to the variables in $\mathbf{x}_i$. Let $f: A \to B$ be a bijection satisfying the condition (BE) for p. Now for each i and $\mathbf{b} \in A^{n_i}$, $|\mathbf{b}| \leqslant n_i \leqslant n$ and $|\mathbf{a}_i \cup \mathbf{b}| \leqslant |\mathbf{x}_i \cup \mathbf{y}_i| \leqslant k$, and so $(p \upharpoonright \mathbf{a}_i) \cup (f \upharpoonright \mathbf{b}) \in \mathbf{I}^k(\mathbf{A}, \mathbf{B})$. By induction hypothesis, $(p \upharpoonright \mathbf{a}_i) \cup (f \upharpoonright \mathbf{b})$ preserves the truth of ψ_i:

$$\langle \mathbf{A}, \mathbf{a}_i, \mathbf{b} \rangle \models \psi_i(\mathbf{x}_i, \mathbf{y}_i) \Leftrightarrow \langle \mathbf{B}, p\mathbf{a}_i, f\mathbf{b} \rangle \models \psi_i(\mathbf{x}_i, \mathbf{y}_i).$$

However, this means that f is an isomorphism

$$\langle A, \psi_1^{\mathbf{A}}(\mathbf{a}_1, \cdot), ..., \psi_l^{\mathbf{A}}(\mathbf{a}_l, \cdot) \rangle \cong \langle B, \psi_1^{\mathbf{B}}(p\mathbf{a}_1, \cdot), ..., \psi_l^{\mathbf{B}}(p\mathbf{a}_l, \cdot) \rangle.$$

Since q is invariant under isomorphisms, we conclude that

$$\langle \mathbf{A}, \mathbf{a} \rangle \models \varphi(\mathbf{x}) \Leftrightarrow \langle \mathbf{B}, p\mathbf{a} \rangle \models \varphi(\mathbf{x});$$

i.e., p preserves the truth of φ. This completes the induction, and we have thus established the first inclusion $\mathbf{I}^k(\mathbf{A}, \mathbf{B}) \subseteq \mathbf{J}^k(\mathbf{A}, \mathbf{B})$.

Next we prove by induction on m that $\mathbf{K}^k(\mathbf{A}, \mathbf{B}) \subseteq I^k_m(\mathbf{A}, \mathbf{B})$ for all $m \in \omega$. Since $\mathbf{I}^k(\mathbf{A}, \mathbf{B}) = \bigcap_{m \in \omega} I^k_m(\mathbf{A}, \mathbf{B})$, this will complete the proof of the lemma. The case $m = 0$ is clear, since all $p \in \mathbf{K}^k(\mathbf{A}, \mathbf{B})$ are partial isomorphisms. Assume then that $m > 0$ and the claim holds for $m - 1$. Suppose that contrary to the claim there exists $p \in \mathbf{K}^k(\mathbf{A}, \mathbf{B}) \backslash I^k_m(\mathbf{A}, \mathbf{B})$. Let $\mathbf{a}$ be a tuple enumerating the elements of $\operatorname{dom}(p)$ (i.e., each element of $\operatorname{dom}(p)$ occurs in $\mathbf{a}$ exactly once), and let $\mathbf{x}$ be a tuple of variables of the same length ($\leqslant k$). Thus, for every bijection $f: A \to B$ there exist a subtuple $\mathbf{a}_f$ of $\mathbf{a}$ and a tuple $\mathbf{b}_f \in A^{\leqslant n}$ such that $|\mathbf{a}_f \cup \mathbf{b}_f| \leqslant k$ and $(p \upharpoonright \mathbf{a}_f) \cup (f \upharpoonright \mathbf{b}_f) \notin I^k_{m-1}(\mathbf{A}, \mathbf{B})$, and so, by induction hypothesis, $(p \upharpoonright \mathbf{a}_f) \cup (f \upharpoonright \mathbf{b}_f) \notin \mathbf{K}^k(\mathbf{A}, \mathbf{B})$. Hence, for each bijection $f: A \to B$ we can choose an $\mathrm{FO}^k(\mathbf{Q}_n)$-formula $\psi_f(\mathbf{x}_f, \mathbf{y}_f)$ such that

$$\langle \mathbf{A}, \mathbf{a}_f, \mathbf{b}_f \rangle \models \psi_f(\mathbf{x}_f, \mathbf{y}_f) \quad \text{and} \quad \langle \mathbf{B}, p\mathbf{a}_f, f\mathbf{b}_f \rangle \not\models \psi_f(\mathbf{x}_f, \mathbf{y}_f).$$

Here $\mathbf{x}_f$ is the subtuple of $\mathbf{x}$ corresponding to the subtuple $\mathbf{a}_f$ of $\mathbf{a}$. It follows that the structures

$$\mathbf{A}_p = \langle A, \psi^{\mathbf{A}}_{f_1}(\mathbf{a}_{f_1}, \cdot), ..., \psi^{\mathbf{A}}_{f_s}(\mathbf{a}_{f_s}, \cdot) \rangle$$
$$\text{and} \qquad \mathbf{B}_p = \langle B, \psi^{\mathbf{B}}_{f_1}(p\mathbf{a}_{f_1}, \cdot), ..., \psi^{\mathbf{B}}_{f_s}(p\mathbf{a}_{f_s}, \cdot) \rangle$$

are non-isomorphic, where $f_1, ..., f_s$ is a list of all bijections $A \to B$. We let now q be the query consisting of all structures isomorphic with $\mathbf{A}_p$. Since $|\mathbf{y}_{f_i}| = |\mathbf{b}_{f_i}| \leqslant n$ for $1 \leqslant i \leqslant s$, Q_q is an n-ary quantifier. Thus,

$$\varphi(\mathbf{x}) = Q_q \mathbf{y}_{f_1}, ..., \mathbf{y}_{f_s}(\psi_{f_1}(\mathbf{x}_{f_1}, \mathbf{y}_{f_1}), ..., \psi_{f_s}(\mathbf{x}_{f_s}, \mathbf{y}_{f_s}))$$

is an $\mathrm{FO}^k(\mathbf{Q}_n)[\tau]$-formula such that $\langle \mathbf{A}, \mathbf{a} \rangle \models \varphi(\mathbf{a})$ but $\langle \mathbf{B}, p\mathbf{a} \rangle \not\models \varphi(\mathbf{a})$. This is in contradiction with our assumption that $p \in \mathbf{K}^k(\mathbf{A}, \mathbf{B})$. ∎

As an immediate corollary we get a back-and-forth characterization for equivalence with respect to $\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)$ and $\mathrm{FO}^k(\mathbf{Q}_n)$.

5.8. Theorem. *Let* $\mathbf{a} \in A^l$, $l \leqslant k$, *and let* p *be a function such that* $\operatorname{dom}(p) = \mathbf{a}$ *and* $\operatorname{rng}(p) \subseteq B$. *The following conditions are equivalent*:

(1) $p \in \mathbf{I}^k(\mathbf{A}, \mathbf{B})$,

(2) $\langle \mathbf{A}, \mathbf{a} \rangle \equiv \langle \mathbf{B}, p\mathbf{a} \rangle (\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n))$,

(3) $\langle \mathbf{A}, \mathbf{a} \rangle \equiv \langle \mathbf{B}, p\mathbf{a} \rangle (\mathrm{FO}^k(\mathbf{Q}_n))$.

Proof. Obviously $\langle \mathbf{A}, \mathbf{a} \rangle \equiv \langle \mathbf{B}, p\mathbf{a} \rangle (\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n))$ if and only if $p \in \mathbf{J}^k(\mathbf{A}, \mathbf{B})$, and similarly, $\langle \mathbf{A}, \mathbf{a} \rangle \equiv \langle \mathbf{B}, p\mathbf{a} \rangle (\mathrm{FO}^k(\mathbf{Q}_n))$ if and only if $p \in \mathbf{K}^k(\mathbf{A}, \mathbf{B})$. Hence, the claim follows from Lemma 5.7. ∎

Note that, as a special case ($l = 0$) of the characterization above, we get the equivalence

$$\mathbf{A} \equiv \mathbf{B}(\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)) \Leftrightarrow \varnothing \in \mathbf{I}^k(\mathbf{A}, \mathbf{B}).$$

The promised pebble game characterization for $\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)$ can now be derived from Theorem 5.8 as follows. Let $\mathbf{a} \in A^l$ and $\mathbf{b} \in B^l$, $l \leqslant k$, be tuples of distinct elements. Imagine two players, I and II, arguing about the possible $\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)$-equivalence of $\langle \mathbf{A}, \mathbf{a} \rangle$ and $\langle \mathbf{B}, \mathbf{b} \rangle$: player II claims that $\langle \mathbf{A}, \mathbf{a} \rangle \equiv \langle \mathbf{B}, \mathbf{b} \rangle (\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n))$ and player I tries to refute this. Thus, player II has to show that the function $p = \{(a_1, b_1), ..., (a_l, b_l)\}$ is in $\mathbf{I}^k(\mathbf{A}, \mathbf{B})$, so he starts by giving a bijection $f_1 \colon A \to B$ and claiming that it satisfies the bijective extension condition (BE) for p. Player I challenges this move of II by choosing sets $C_1 \subseteq \mathrm{dom}(p)$ and $D_1 \subseteq A$ such that $|D_1| \leqslant n$ and $|C_1 \cup D_1| \leqslant k$. If the function $p_1 = (p \restriction C_1) \cup (f_1 \restriction D_1)$ is not a partial isomorphism $\mathbf{A} \to \mathbf{B}$, then player II was wrong in his claim about f_1, and hence player I wins the game. If this is not the case, player II gives a new bijection $f_2 \colon A \to B$ as evidence that $p_1 \in \mathbf{I}^k(\mathbf{A}, \mathbf{B})$, and player I answers by choosing sets $C_2 \subseteq \mathrm{dom}(p_1)$ and $D_2 \subseteq A$ such that $|D_2| \leqslant n$ and $|C_2 \cup D_2| \leqslant k$. Player I wins the game at this point if the new function $p_2 = (p_1 \restriction C_2) \cup (f_2 \restriction D_2)$ is not a partial isomorphism. Otherwise player II has to show that $p_2 \in \mathbf{I}^k(\mathbf{A}, \mathbf{B})$, and the game goes on.

Thus, after each round $i \in \omega$ of the game the moves of the players determine a function p_i with $\mathrm{dom}(p_i) \subseteq A$ and $\mathrm{rng}(p_i) \subseteq B$. Player I wins the game if after some round this function is not a partial isomorphism, while player II wins if I does not win in any finite number of rounds.

We call this game the *n-bijective k-pebble game*[2] for $\langle \mathbf{A}, \mathbf{a} \rangle$ and $\langle \mathbf{B}, \mathbf{b} \rangle$, and denote it by $BP^k_n(\mathbf{A}, \mathbf{a}, \mathbf{B}, \mathbf{b})$. (In the case $l = 0$ we write just $BP^k_n(\mathbf{A}, \mathbf{B})$ instead of $BP^k_n(\mathbf{A}, \varnothing, \mathbf{B}, \varnothing)$.)

It is easy to see that a winning strategy of player II in $BP^k_n(\mathbf{A}, \mathbf{a}, \mathbf{B}, \mathbf{b})$ gives rise to a set of partial isomorphisms containing the function $p = \{(a_1, b_1), ..., (a_l, b_l)\}$ and satisfying the bijective extension condition (BE), and vice versa. Hence, we can now state our pebble game characterization of $\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)$-equivalence.

5.9. COROLLARY. *The following conditions are equivalent:*

(1) *player II has a winning strategy in the game* $BP^k_n(\mathbf{A}, \mathbf{a}, \mathbf{B}, \mathbf{b})$,

(2) $\langle \mathbf{A}, \mathbf{a} \rangle \equiv \langle \mathbf{B}, \mathbf{b} \rangle (\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n))$,

(3) $\langle \mathbf{A}, \mathbf{a} \rangle \equiv \langle \mathbf{B}, \mathbf{b} \rangle (\mathrm{FO}^k(\mathbf{Q}_n))$.

Note that $BP^k_n(\mathbf{A}, \mathbf{a}, \mathbf{B}, \mathbf{b})$ is a determined game in spite of its infinite length. This follows easily from the finiteness of $\mathbf{A}$ and $\mathbf{B}$, since player I can test all possible strategies during a single play of the game. (In fact, by the well-known theorem of Gale and Stewart, the game $BP^k_n(\mathbf{A}, \mathbf{a}, \mathbf{B}, \mathbf{b})$ is determined even if $\mathbf{A}$ and $\mathbf{B}$ were infinite.)

Using Corollary 5.9 we can now prove the following pebble game characterization for definability in $\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)$.

5.10. THEOREM. *An l-ary query q on σ-structures is definable in* $\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)$ *if and only if player I has a winning strategy in the game* $BP^k_n(\mathbf{C}, \mathbf{c}, \mathbf{D}, \mathbf{d})$ *whenever* $\mathbf{c} \in q(\mathbf{C})$ *and* $\mathbf{d} \notin q(\mathbf{D})$. *Similarly, a Boolean query q on σ-structures is definable in* $\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)$ *if and only if player I has a winning strategy in the game* $BP^k_n(\mathbf{C}, \mathbf{D})$ *whenever* $q(\mathbf{C}) \neq q(\mathbf{D})$.

Proof. The implication from left to right is an easy consequence of Corollary 5.9. Assume for the reverse implication that I has a winning strategy in $BP^k_n(\mathbf{C}, \mathbf{c}, \mathbf{D}, \mathbf{d})$ for all σ-structures $\mathbf{C}$ and $\mathbf{D}$ and l-tuples $\mathbf{c}$ and $\mathbf{d}$ such that $\mathbf{c} \in q(\mathbf{C})$ and $\mathbf{d} \notin q(\mathbf{D})$. By Corollary 5.9, for each such quadruple $(\mathbf{C}, \mathbf{c}, \mathbf{D}, \mathbf{d})$ there is a formula $\varphi_{\mathbf{C}, \mathbf{c}, \mathbf{D}, \mathbf{d}}(\mathbf{x}) \in \mathrm{FO}^k(\mathbf{Q}_n)[\sigma]$ such that $\langle \mathbf{C}, \mathbf{c} \rangle \models \varphi_{\mathbf{C}, \mathbf{c}, \mathbf{D}, \mathbf{d}}(\mathbf{x})$, but $\langle \mathbf{D}, \mathbf{d} \rangle \not\models \varphi_{\mathbf{C}, \mathbf{c}, \mathbf{D}, \mathbf{d}}(\mathbf{x})$. Let S be the set of all proper initial segments of ω, and let $U = \{(\mathbf{C}, \mathbf{c}) \mid \mathbf{c} \in q(\mathbf{C}),\ C \in S\}$ and $V = \{(\mathbf{D}, \mathbf{d}) \mid \mathbf{d} \notin q(\mathbf{D}),\ D \in S\}$. Note that U and V are countable sets, and their union $U \cup V$ contains up to isomorphism all pairs $(\mathbf{C}, \mathbf{c})$ such that $\mathbf{C}$ is a (finite) σ-structure and $\mathbf{c} \in C^l$. It is straightforward to verify that the formula

$$\bigvee_{(\mathbf{C}, \mathbf{c}) \in U} \left(\bigwedge_{(\mathbf{D}, \mathbf{d}) \in V} \varphi_{\mathbf{C}, \mathbf{c}, \mathbf{D}, \mathbf{d}}(\mathbf{x}) \right)$$

defines q. ∎

A pebble game characterization for equivalence with respect to $\mathscr{L}^k_{\infty\omega}$ extended by the set $\mathbf{C}$ of all counting quantifiers was introduced by Immerman and Lander (1990), and it was used in the proof of the main result of Cai *et al.* (1992). Although the game of Immerman and Lander (1990) looks different at first sight, it is equivalent with the 1-bijective k-pebble game $BP^k_1(\mathbf{A}, \mathbf{B})$. In fact, as one easily verifies, all unary quantifiers are definable in $\mathscr{L}^1_{\infty\omega}(\mathbf{C})$. Hence, we have

5.11. PROPOSITION (Kolaitis and Väänänen, 1995). *For every natural number k,* $\mathscr{L}^k_{\infty\omega}(\mathbf{C}) \equiv_{\mathscr{F}} \mathscr{L}^k_{\infty\omega}(\mathbf{Q}_1)$.

Kolaitis and Väänänen (1995) define so-called $(k, \mathbf{Q})$-pebble games which characterize $\mathscr{L}^k_{\infty\omega}(\mathbf{Q})$-equivalence for arbitrary sets $\mathbf{Q}$ of quantifiers. Thus, in principle we could use these games in the proofs of our results. However, the $(k, \mathbf{Q}_n)$-pebble game is much more complicated than $BP^k_n(\mathbf{A}, \mathbf{B})$, since it contains a different rule for making moves for each different quantifier $Q_q \in \mathbf{Q}_n$.

6. BUILDING BLOCKS

In the next section we are going to construct for each natural number m a pair of non-isomorphic finite structures which cannot be separated by sentences of $\mathrm{FO}(\mathbf{Q}_n)$ of quantifier rank $\leqslant m$. Furthermore, in Section 8 we will construct for each natural number k a pair of finite structures that

[2] It is common to make games of this type more concrete by thinking them in terms of k pairs of pebbles that the players use to mark the elements chosen from the structure $\mathbf{A}$ and the corresponding elements chosen from $\mathbf{B}$ (see, e.g., Kolaitis and Vardi, 1992a). This is why we use the term "pebble game."

cannot be separated even in the logic $\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)$. A common feature in these constructions is that the structures are formed from copies of a small "building block" consisting of an $(n+1)$-ary relation on a set of $2n+2$ elements. In both constructions the building blocks are joined together via a suitably chosen binary relation. The present section is devoted to the definition and basic properties of our building blocks.

Let $C=\{c_1, ..., c_{n+1}, d_1, ..., d_{n+1}\}$, where all the $2n+2$ elements $c_1, ..., c_{n+1}, d_1, ..., d_{n+1}$ are distinct, and let $\prec$ be the (strict) partial order of C defined by

- $x \prec y \Leftrightarrow x \in \{c_i, d_i\}$ and $y \in \{c_j, d_j\}$ for some $1 \leqslant i < j \leqslant n+1$.

Thus, $\prec$ can be viewed as a linear order between the c, d-pairs $\{c_i, d_i\}$, $1 \leqslant i \leqslant n+1$. Let P be the subset of C consisting of the elements $c_1, ..., c_{n+1}$.

6.1. Definition. We define two $(n+1)$-ary relations R^+ and R^- on C as follows:

- $(a_1, ..., a_{n+1}) \in R^+ \Leftrightarrow a_1 \prec \cdots \prec a_{n+1}$ and $|\{i \mid a_i \notin P\}|$ is even,
- $(a_1, ..., a_{n+1}) \in R^- \Leftrightarrow a_1 \prec \cdots \prec a_{n+1}$ and $|\{i \mid a_i \notin P\}|$ is odd.

We denote the models $\langle C, R^+\rangle$ and $\langle C, R^-\rangle$ by $\mathbf{C}^+$ and $\mathbf{C}^-$, respectively.

Note that $(c_1, ..., c_{n+1}) \in R^+ \backslash R^-$ irrespective of n, but $(d_1, ..., d_{n+1}) \in R^+ \backslash R^-$ if n is odd and $(d_1, ..., d_{n+1}) \in R^- \backslash R^+$ if n is even.

The automorphisms of $\mathbf{C}^+$ and $\mathbf{C}^-$ and the isomorphisms between $\mathbf{C}^+$ and $\mathbf{C}^-$ will have a crucial role in our considerations. Clearly any of them must be a bijection $f\colon C \to C$ preserving the partial order $\prec$: $a \prec b \Leftrightarrow f(a) \prec f(b)$. Hence, the following lemma describes the automorphisms and isomorphisms completely.

6.2. Lemma. *Let $f\colon C \to C$ be a bijection preserving the partial order $\prec$. Then f is an automorphism of $\mathbf{C}^+$ and $\mathbf{C}^-$ if and only if the number*

$$exc(f) = |\{i \in \{1, ..., n+1\} \mid f(c_i) = d_i\}|$$

of c, d-exchanges of f is even. Similarly, f is an isomorphism $\mathbf{C}^+ \to \mathbf{C}^-$ if and only if $exc(f)$ is odd.

Proof. If $exc(f)$ is even, then for all $\mathbf{a}=(a_1, ..., a_{n+1}) \in C^{n+1}$, the numbers $|\{i \mid a_i \notin P\}|$ and $|\{i \mid f(a_i) \notin P\}|$ are of the same parity, whence $\mathbf{a} \in R^+ \Leftrightarrow f\mathbf{a} \in R^+$ and $\mathbf{a} \in R^- \Leftrightarrow f\mathbf{a} \in R^-$. On the other hand, if $exc(f)$ is odd, then $|\{i \mid a_i \notin P\}|$ and $|\{i \mid f(a_i) \notin P\}|$ are of different parity, and so $\mathbf{a} \in R^+ \Leftrightarrow f\mathbf{a} \in R^-$. ∎

6.3. Lemma. *If $f\colon C \to C$ is a bijection preserving the partial order $\prec$, and $a_1, ..., a_n \in C$, then there is an automorphism g of $\mathbf{C}^+$ and $\mathbf{C}^-$ and an isomorphism $g'\colon \mathbf{C}^+ \to \mathbf{C}^-$ such that $g(a_i) = g'(a_i) = f(a_i)$ for all $1 \leqslant i \leqslant n$.*

Proof. Let $1 \leqslant i \leqslant n+1$ be such that $c_i, d_i \notin \{a_1, ..., a_n\}$, and let $h\colon C \to C$ be the bijection obtained from f by exchanging c_i and d_i:

$$h(a) = \begin{cases} f(a), & \text{if } \ a \notin \{c_i, d_i\} \\ c_i, & \text{if } \ f(a) = d_i \\ d_i, & \text{if } \ f(a) = c_i. \end{cases}$$

Then one of the numbers $exc(f)$ and $exc(h)$ is even and the other is odd, and so the claim follows from Lemma 6.2. ∎

Note that player II can choose any bijection $f\colon C \to C$ that preserves $\prec$ as his first move f_1 in n-bijective Ehrenfeucht–Fraïssé games for the structures $\mathbf{C}^+$ and $\mathbf{C}^-$, and then win the game by choosing f_i to be the isomorphism g' of Lemma 6.3 (determined by the first move $D_1 = \{a_1, ..., a_n\}$ of player I) for the rest of his moves. Thus, player II has plenty of different winning strategies in the games $BEF^m_n(\mathbf{A}, \mathbf{B})$ and $BP^k_n(\mathbf{A}, \mathbf{B})$ if the structures $\mathbf{A}$ and $\mathbf{B}$ consist of an equal number of disjoint copies of $\mathbf{C}^+$ and $\mathbf{C}^-$. In the following two sections we will show that it is possible to make such $\mathbf{A}$ and $\mathbf{B}$ non-isomorphic by adding suitable binary relations, and still preserve at least one of these winning strategies of player II.

7. A HIERARCHY THEOREM FOR FIXPOINT LOGIC

In this section, we prove that the expressive power of fixpoint logic cannot be captured by adding a finite number of generalized quantifiers to first-order logic. In fact, we will show that there is a whole hierarchy of LFP queries which is not definable in $\mathrm{FO}(\mathbf{Q})$ for any set $\mathbf{Q}$ of quantifiers of bounded arity.

7.1. Theorem. *For each natural number n, there exists a vocabulary τ and a (unary) LFP-definable query on τ-structures which is not expressible in the logic $\mathrm{FO}(\mathbf{Q}_n)$. Hence, $\mathrm{LFP} \leqslant_{\mathscr{F}} \mathrm{FO}(\mathbf{Q}_n)$ does not hold for any n.*

Theorem 7.1 extends the earlier result of Hella and Sandu (1995) that connectivity of finite graphs cannot be expressed in $\mathrm{FO}(\mathbf{Q}_1)$. As an immediate consequence for the theorem we have

7.2. Corollary. *There does not exist a finite set $\mathbf{Q}$ of generalized quantifiers such that $\mathrm{LFP} \equiv_{\mathscr{F}} \mathrm{FO}(\mathbf{Q})$.*

Let n be a positive integer, and let τ be the vocabulary consisting of relation symbols R, S, and T with arities $(n+1)$, 2, and 1, respectively. In order to prove Theorem 7.1 we will first give a unary query q on τ-structures and show that it is definable in LFP. Then we define for each natural number m two τ-structures $\mathbf{A}_m$ and $\mathbf{B}_m$ such that $\langle \mathbf{A}_m, a\rangle \equiv_n^m \langle \mathbf{B}_m, b\rangle$, $a\in q(\mathbf{A}_m)$, but $b\notin q(\mathbf{B}_m)$ for certain (definable) elements $a\in A_m$ and $b\in B_m$. The query q is related to a two-person game which we now describe.

Let $\mathbf{A}=\langle A, R^{\mathbf{A}}, S^{\mathbf{A}}, T^{\mathbf{A}}\rangle$ be a τ-model and $a\in A$. The game $G(\mathbf{A}, a)$ is played between two players I and II. In each round i of the game player I picks an element $a_i\in A$ and player II responds by choosing an $(n+1)$-tuple $(b_1^i, ..., b_{n+1}^i)\in A^{n+1}$. As the players make their moves they must obey the following rules:

- $a_1=a$ and $a_{i+1}\in\{b_1^i, ..., b_{n+1}^i\}$,
- $(b_1^i, ..., b_{n+1}^i)\in R^{\mathbf{A}}$ and $(a_i, b_1^i), ..., (a_i, b_{n+1}^i)\in S^{\mathbf{A}}$.

The game ends if $a_i\in T^{\mathbf{A}}$ for some i, or if player II cannot make a legal move in some round (note that player I can always move according the rules). In the first case player II wins the game. Player I wins in the second case, and also in the case that the game does not end in a finite number of rounds.

The query q is now defined in terms of the game G:

7.3. Definition. For each τ-structure $\mathbf{A}$, $q(\mathbf{A})$ is the set of those elements $a\in A$ for which player II has a winning strategy in the game $G(\mathbf{A}, a)$.

Clearly $q(\mathbf{A})$ is the smallest subset of A satisfying:

- $T^{\mathbf{A}}\subseteq q(\mathbf{A})$,
- if $a_1, ..., a_{n+1}\in q(\mathbf{A})$, $(a_1, ..., a_{n+1})\in R^{\mathbf{A}}$, and $(a, a_i)\in S^{\mathbf{A}}$ for all $1\leqslant i\leqslant n+1$, then $a\in q(\mathbf{A})$.

This inductive definition of $q(\mathbf{A})$ is easily captured by a formula of least fixpoint logic:

7.4. Lemma. *The query q is definable in* LFP.

Proof. Let $\varphi(X, x)$ be the formula

$$T(x)\vee \exists x_1\cdots\exists x_{n+1}((X(x_1)\wedge\cdots\wedge X(x_{n+1})) \wedge R(x_1, ..., x_{n+1})\wedge(S(x, x_1)\wedge\cdots\wedge S(x, x_{n+1}))).$$

Then $(\mathsf{LFP}_{X,x}\varphi)[x]$ is a formula of LFP defining the query q. ∎

In fact, the query q is simple enough to be computable by DATALOG program.[3] DATALOG is the database query language whose programs consist of Horn clauses without function symbols. It is well known that DATALOG is properly contained in LFP. We refer to Kolaitis (1991) for a definition and more information on DATALOG.

7.5. Lemma. *There is a* DATALOG *program which computes the query q.*

[3] I am grateful to Phokion Kolaitis for pointing this out to me.

Proof. The required program consists of the two lines

$$X(x)\leftarrow T(x)$$

$$X(x)\leftarrow X(x_1), ..., X(x_{n+1}), R(x_1, ..., x_{n+1}), S(x, x_1), ..., S(x, x_{n+1})$$

where X is the goal predicate. ∎

Our next task is to define for each natural number m the promised τ-structures $\mathbf{A}_m$ and $\mathbf{B}_m$.

7.6. Definition. Let m be a natural number. The τ-structures $\mathbf{A}_m=\langle A_m, R^{\mathbf{A}_m}, S^{\mathbf{A}_m}, T^{\mathbf{A}_m}\rangle$ and $\mathbf{B}_m=\langle B_m, R^{\mathbf{B}_m}, S^{\mathbf{B}_m}, T^{\mathbf{B}_m}\rangle$, together with auxiliary unary relations $P^{\mathbf{A}_m}\subseteq A_m$ and $P^{\mathbf{B}_m}\subseteq B_m$, are defined as follows:

- $A_m=B_m=C^{\leqslant m}=\{(a_1,...,a_i)\mid i\leqslant m,\ a_1, ..., a_i\in C\}\cup\{\varnothing\}$,
- $P^{\mathbf{A}_m}=\{(a_1, ..., a_i)\in C^{\leqslant m}\mid a_i\in P\}\cup\{\varnothing\}$ *and* $P^{\mathbf{B}_m}=P^{\mathbf{A}_m}\setminus\{\varnothing\}$,
- $S^{\mathbf{A}_m}=S^{\mathbf{B}_m}=\{(s, s^\frown(a))\mid s\in C^{\leqslant m-1}, a\in C\}$,
- for $\mathbf{D}=\mathbf{A}_m, \mathbf{B}_m$, $R^{\mathbf{D}}$ is the set of all tuples $(s^\frown(a_1), ..., s^\frown(a_{n+1}))\in(C^{\leqslant m})^{n+1}$ such that either $s\in P^{\mathbf{D}}$ and $(a_1, ..., a_{n+1})\in R^+$, or $s\notin P^{\mathbf{D}}$ and $(a_1, ..., a_{n+1})\in R^-$,
- for $\mathbf{D}=\mathbf{A}_m, \mathbf{B}_m$, $T^{\mathbf{D}}=P^{\mathbf{D}}\cap C^m$.

Thus, $\mathbf{A}_m$ and $\mathbf{B}_m$ consist of copies of the structures $\mathbf{C}^+$ and $\mathbf{C}^-$ ordered in a tree-like fashion by the relation S (see Fig. 1). In both models the empty sequence, $\varnothing$, is the unique element without S-predecessors. For $m=0$, it is also the only element, and the difference between $\mathbf{A}_0$ and $\mathbf{B}_0$ is that $\varnothing\in T^{\mathbf{A}_0}$, but $\varnothing\notin T^{\mathbf{B}_0}$. For $m>0$, the models $\mathbf{A}_m$ and $\mathbf{B}_m$ are identical except that in $\mathbf{A}_m$ the "root structure" $\langle C^1, R^{\mathbf{A}_m}\cap(C^1)^{n+1}\rangle$ is a copy of $\mathbf{C}^+$ while in $\mathbf{B}_m$ it is a copy of $\mathbf{C}^-$.

The auxiliary relation P was used in the definition of $\mathbf{A}_m$ and $\mathbf{B}_m$, but it has also another role in our considerations:

7.7. Lemma. *For all natural numbers m, $q(\mathbf{A}_m)=P^{\mathbf{A}_m}$ and $q(\mathbf{B}_m)=P^{\mathbf{B}_m}$. In particular, $\varnothing\in q(\mathbf{A}_m)$, but $\varnothing\notin q(\mathbf{B}_m)$.*

Proof. Let $\mathbf{D}$ be either $\mathbf{A}_m$ or $\mathbf{B}_m$. By Definition 7.6, we have $T^{\mathbf{D}}\subseteq P^{\mathbf{D}}$. Furthermore, if $(s^\frown(a_1), ..., s^\frown(a_{n+1}))\in R^{\mathbf{D}}$ and $(s^\frown(a_i))\in P^{\mathbf{D}}$ for each $1\leqslant i\leqslant n+1$, then necessarily $(a_1, ..., a_{n+1})=(c_1, ..., c_{n+1})$. Hence $(a_1, ..., a_{n+1})\in R^+\setminus R^-$, and so $s\in P^{\mathbf{D}}$ by the definition of $R^{\mathbf{D}}$. Thus, $P^{\mathbf{D}}$ satisfies the conditions of the inductive definition of $q(\mathbf{D})$, and so $q(\mathbf{D})\subseteq P^{\mathbf{D}}$.

On the other hand, if $s\in P^{\mathbf{D}}\setminus T^{\mathbf{D}}$ is a move of player I in the game $G(\mathbf{D}, a)$, then player II responds to it by choosing

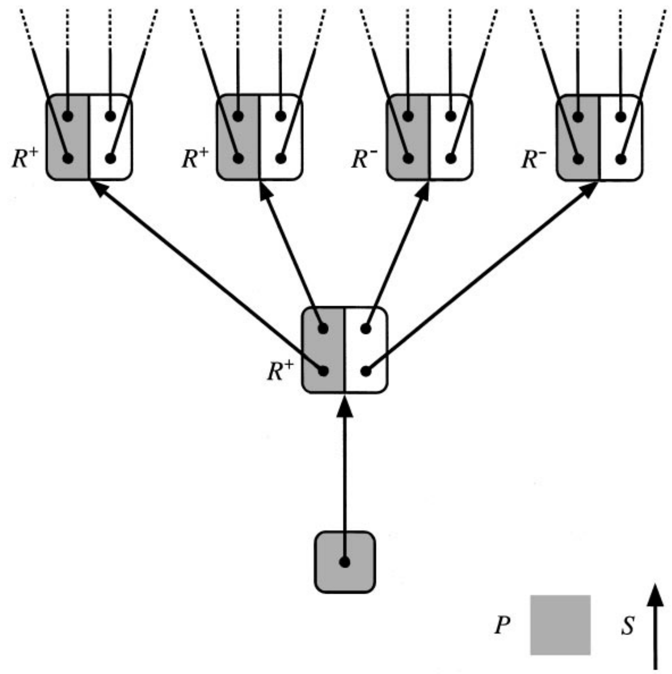

FIG. 1. A fragment of the structure $\mathbf{A}_m$ for $n = 1$.

the tuple $(s^\frown(c_1), ..., s^\frown(c_{n+1}))$, which is in $R^{\mathbf{D}}$, since $(c_1, ..., c_{n+1}) \in R^+$. The next move of player I is then necessarily in $P^{\mathbf{D}}$. Thus, player II can guarantee that the rest of the moves of player I are in $P^{\mathbf{D}}$. Since the length of the tuples from $C^{\leqslant m}$ chosen by player I increases every round, it is clear that he finally has to choose a tuple $s \in C^m \cap P^{\mathbf{D}} = T^{\mathbf{D}}$, and player II wins. Hence, also $P^{\mathbf{D}} \subseteq q(\mathbf{D})$. ▮

The final step in the proof of Theorem 7.1 is to show that the structures $\langle \mathbf{A}_m, \varnothing \rangle$ and $\langle \mathbf{B}_m, \varnothing \rangle$ are equivalent with respect to all n-ary quantifiers up to quantifier rank m (note that here and below $\varnothing$ refers to the unique S-minimal element of $\mathbf{A}_m$ and $\mathbf{B}_m$ rather than the empty sequence of elements).

7.8. Lemma. *For all natural numbers $m > 0$, $\langle \mathbf{A}_m, \varnothing \rangle \equiv_n^m \langle \mathbf{B}_m, \varnothing \rangle$.*

Proof. We will prove by induction on m that player II has a winning strategy in the bijective Ehrenfeucht–Fraïssé game $BEF_n^m(\mathbf{A}_m, \varnothing, \mathbf{B}_m, \varnothing)$. In doing this, we need the following observation: Assume that $m > 0$. For each $a \in C$, let C_a be the set $\{(a)^\frown s \mid s \in C^{\leqslant m-1}\}$, and let $\mathbf{A}_a$ and $\mathbf{B}_a$ be the relativizations of $\mathbf{A}_m$ and $\mathbf{B}_m$, respectively, to the set C_a; i.e.,

$$\mathbf{A}_a = \langle C_a, R^{\mathbf{A}_m} \cap (C_a)^{n+1}, S^{\mathbf{A}_m} \cap (C_a)^2, T^{\mathbf{A}_m} \cap C_a \rangle$$

and similarly

$$\mathbf{B}_a = \langle C_a, R^{\mathbf{B}_m} \cap (C_a)^{n+1}, S^{\mathbf{B}_m} \cap (C_a)^2, T^{\mathbf{B}_m} \cap C_a \rangle.$$

Then from Definition 7.6, we see that $\mathbf{A}_a = \mathbf{B}_a \cong \mathbf{A}_{m-1}$ if $a \in P$, and $\mathbf{A}_a = \mathbf{B}_a \cong \mathbf{B}_{m-1}$ if $a \notin P$.

Consider first the game $BEF_n^1(\mathbf{A}_1, \varnothing, \mathbf{B}_1, \varnothing)$. Player II wins this game just by giving the identity function $f_1 = id_{C^{\leqslant 1}}$ as his only move: it is obvious that, for any set $D_1 \subseteq C^{\leqslant 1}$ with $|D_1| \leqslant n$, the relation $p = \{(\varnothing, \varnothing)\} \cup (f_1 \restriction D_1)$ is an injective function preserving the relations S and T. Moreover, if $\mathbf{a} \in (\{\varnothing\} \cup D_1)^{n+1}$, then $\mathbf{a} \notin R^{\mathbf{A}_1}$ and $p\mathbf{a} \notin R^{\mathbf{B}_1}$. (Note that $\langle \mathbf{A}_0, \varnothing \rangle \not\equiv_n^0 \langle \mathbf{B}_0, \varnothing \rangle$, since, as we already remarked, $\varnothing \in T^{\mathbf{A}_0} \setminus T^{\mathbf{B}_0}$.)

Assume then that $m > 1$ and player II has a winning strategy in the game $BEF_n^{m-1}(\mathbf{A}_{m-1}, \varnothing, \mathbf{B}_{m-1}, \varnothing)$. We describe now a winning strategy for player II in the game $BEF_n^m(\mathbf{A}_m, \varnothing, \mathbf{B}_m, \varnothing)$. The first move of player II is again the identity function $f_1 = id_{C^{\leqslant m}}$. Let $D_1 = \{s_1, ..., s_n\}$ be the answer of player I to this move. There are at most n different elements $a \in C$ such that $(a)^\frown s \in D_1$ for some $s \in C^{\leqslant m-1}$; let these elements be $a_1, ..., a_k$ $(k \leqslant n)$. By Lemma 6.3, there is an isomorphism $g\colon \mathbf{C}^+ \to \mathbf{C}^-$ such that $g(a_i) = f_1(a_i)$ for $1 \leqslant i \leqslant k$. Thus, $\{(\varnothing, \varnothing)\} \cup g$ is a partial isomorphism $\mathbf{A}_m \to \mathbf{B}_m$, and for the rest of his moves f_j, $1 < j \leqslant m$, player II can fix $f_j(\varnothing) = \varnothing$ and $f_j(a) = g(a)$ for all $a \in C$. Furthermore, as we noted above, $\mathbf{A}_{a_i} = \mathbf{B}_{a_i} = \mathbf{B}_{g(a_i)}$ for $1 \leqslant i \leqslant k$, whence player II can safely use the identity function in defining $f_j \restriction C_{a_i}$ for $1 < j \leqslant m$. Finally, no elements of the models $\mathbf{A}_a$ and $\mathbf{B}_{g(a)}$ are fixed yet for $a \in C \setminus \{a_1, ..., a_k\}$ and all these structures are isomorphic with either $\mathbf{A}_{m-1}$ or $\mathbf{B}_{m-1}$. Hence, using his winning strategy for $BEF_n^{m-1}(\mathbf{A}_{m-1}, \varnothing, \mathbf{B}_{m-1}, \varnothing)$, player II can also find the restrictions $f_j \restriction C_a$ for $1 \leqslant j < m$ and $a \in C \setminus \{a_1, ..., a_k\}$ in such a way that he wins the game $BEF_n^m(\mathbf{A}_m, \varnothing, \mathbf{B}_m, \varnothing)$. ▮

We have thus completed the proof of Theorem 7.1, since by Corollary 5.4 the LFP-definable query q is not definable in $\mathrm{FO}(\mathbf{Q}_n)$. Since, by Lemma 7.5, q is a DATALOG query, we have actually proved that even DATALOG is too strong to be captured by quantifiers of bounded arity.

7.9. Theorem. *For each natural number n, there exists a* DATALOG *query which is not definable in the logic* $\mathrm{FO}(\mathbf{Q}_n)$. *In particular, the expressive power of* DATALOG *is not subsumed by* $\mathrm{FO}(\mathbf{Q})$ *for any finite set* $\mathbf{Q}$ *of quantifiers.*

The present formulation of our proof of Theorem 7.1 was inspired by Kolaitis (1991), who used so-called game trees for proving that the query language consisting of stratified DATALOG programs has strictly weaker expressive power than fixpoint logic.

8. A HIERARCHY THEOREM FOR PTIME

Cai *et al.* (1992) proved that there are polynomial time computable queries which are not definable in the finite variable logic with all counting quantifiers, $\mathscr{L}^{\omega}_{\infty\omega}(\mathbf{C})$. We generalize here this result by showing that PTIME is not subsumed by the logic $\mathscr{L}^{\omega}_{\infty\omega}(\mathbf{Q}_n)$ for any natural number n. The proof of Cai *et al.* is based on an elegant construction of pairs $(\mathbf{G}_k, \mathbf{H}_k)$ of graphs which are $\mathscr{L}^{k}_{\infty\omega}(\mathbf{C})$-equivalent, but still separated by a PTIME computable query. We use an essentially similar construction of structures in our proof; the main difference is that we use the structures $\mathbf{C}^+$ and $\mathbf{C}^-$ as building blocks instead of a gadget graph, denoted by X_3 in Cai *et al.* (1992).

One remarkable feature of the graphs $\mathbf{G}_k$ and $\mathbf{H}_k$ constructed by Cai *et al.* (1992) is that they are of *color class size* 4, i.e., the vertices of these graphs can be colored in such a way that for each color there are at most four vertices having that color, and the graphs equipped with these colors are still $\mathscr{L}^{k}_{\infty\omega}(\mathbf{C})$-equivalent. Another way of putting this is that there are partial orders $<_1$ and $<_2$ of width[4] 4 such that $\langle \mathbf{G}_k, <_1\rangle \equiv \langle \mathbf{H}_k, <_2\rangle (\mathscr{L}^{k}_{\infty\omega}(\mathbf{C}))$. The structures we will construct below are of color class size 2; in fact, they contain a partial order of width 2 as one of their basic relations. This decrease in the color class size is possible only because our structures contain a relation of arity greater than 2. Indeed, a result of Immerman and Lander (1990) implies that every query on graphs of color class size 3 is definable in $\mathscr{L}^{\omega}_{\infty\omega}(\mathbf{C})$, whence color class size 4 is best possible in the case of graphs.

Assume that $n \geqslant 2$ and $\mathbf{G} = \langle G, E^{\mathbf{G}}\rangle$ is a finite connected (undirected) graph of degree $n+1$, i.e., every vertex of $\mathbf{G}$ is adjacent to exactly $n+1$ other vertices. Thus, we can fix for each $u \in G$ a function $h_u\colon \{v \mid (u, v) \in E^{\mathbf{G}}\} \to \{1, ..., n+1\}$ which enumerates the vertices adjacent to u, i.e., $h_u(v) \neq h_u(w)$ whenever $(u, v), (u, w) \in E^{\mathbf{G}}$ and $v \neq w$. (Note that we may have $h_u(v) \neq h_v(u)$.)

8.1. Definition. Let C, R^+ and R^- be as defined in Section 6. We define for each subset $S \subseteq G$ a structure $\mathbf{D}(\mathbf{G}, S) = \langle D_G, R^{\mathbf{D}(\mathbf{G}, S)}, E^{\mathbf{D}(\mathbf{G}, S)}\rangle$, where R is $(n+1)$-ary and E is binary:

- $D_G = G \times C$,
- $R^{\mathbf{D}(\mathbf{G}, S)}$ is the set of all tuples $((u, a_1), ..., (u, a_{n+1}))$ in $(D_G)^{n+1}$ such that either $u \notin S$ and $(a_1, ..., a_{n+1}) \in R^+$, or $u \in S$ and $(a_1, ..., a_{n+1}) \in R^-$,
- $E^{\mathbf{D}(\mathbf{G}, S)}$ is the set of all pairs $((u, c_i), (v, c_j))$ and $((u, d_i), (v, d_j))$ in $(D_G)^2$ such that $(u, v) \in E^{\mathbf{G}}$, $i = h_u(v)$, and $j = h_v(u)$.

Thus, $\mathbf{D}(\mathbf{G}, S)$ is obtained from the graph $\mathbf{G}$ by replacing each vertex u with a copy of either $\mathbf{C}^-$ or $\mathbf{C}^+$, depending on whether u is in S or not, and each edge with a double edge connecting the c-components and d-components of a pair of c, d-pairs in the corresponding copies of C. (See Fig. 2.)

[4] The width of a partial order is the maximum number of pairwise incomparable elements.

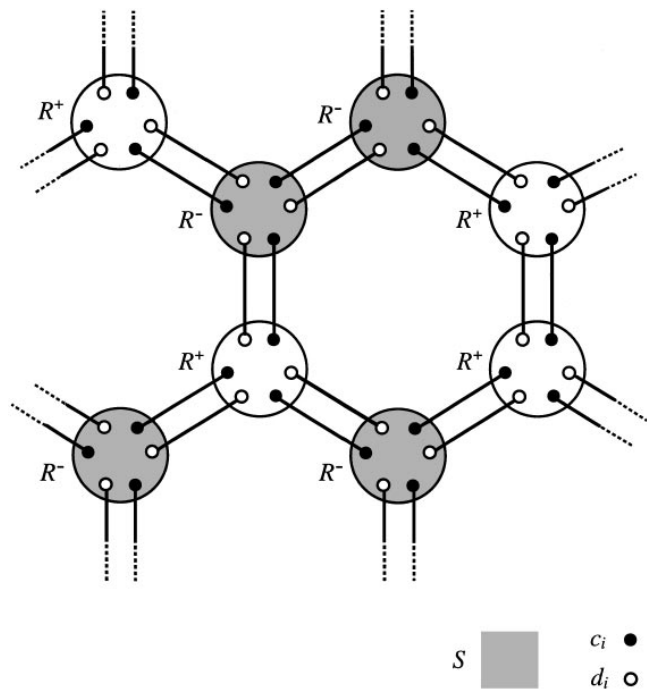

FIG. 2. A fragment of the structure $\mathbf{D}(\mathbf{G}, S)$ for $n = 2$.

Let $S, T \subseteq G$ be such that their symmetric difference consists of two points: $(T \backslash S) \cup (S \backslash T) = \{u, v\}$, $u \neq v$. Since $\mathbf{G}$ is connected, there is a path $v_0, ..., v_m$ from u to v, i.e., $v_0 = u$, $v_m = v$, and $(v_i, v_{i+1}) \in E^{\mathbf{G}}$ for all $i < m$. Let $f\colon D_G \to D_G$ be the bijection which exchanges the c-components and d-components of the c, d-pairs corresponding to the edges (v_i, v_{i+1}) of the path, and is identity elsewhere:

$$f((w, a)) = \begin{cases} (w, d_j), & \text{if} \quad w = v_i,\ a = c_j, \text{ and} \\ & j = h_{v_i}(v_{i+1}) \text{ or } j = h_{v_i}(v_{i-1}) \\ (w, c_j), & \text{if} \quad w = v_i,\ a = d_j, \text{ and} \\ & j = h_{v_i}(v_{i+1}) \text{ or } j = h_{v_i}(v_{i-1}) \\ (w, a), & \text{otherwise.} \end{cases}$$

For each $w \in G$ we denote by f_w the bijection $C \to C$ such that $f_w(a) = b$ if and only if $f((w, a)) = (w, b)$. By Lemma 6.2, we see now that f_{v_i} is an automorphism of $\mathbf{C}^+$ and $\mathbf{C}^-$ for $0 < i < m$, while f_u and f_v are isomorphisms between $\mathbf{C}^+$ and $\mathbf{C}^-$. Thus, f preserves the relation R, i.e., $\mathbf{a} \in R^{\mathbf{D}(\mathbf{G}, S)} \Leftrightarrow f\mathbf{a} \in R^{\mathbf{D}(\mathbf{G}, T)}$ for every $\mathbf{a} \in (D_G)^{n+1}$. Since f clearly preserves also the edge relation E, we conclude that f is an isomorphism $\mathbf{D}(\mathbf{G}, S) \to \mathbf{D}(\mathbf{G}, T)$.

8.2. Lemma. *Let $S, T \subseteq G$. The structures $\mathbf{D}(\mathbf{G}, S)$ and $\mathbf{D}(\mathbf{G}, T)$ are isomorphic if and only if S and T are of the same parity.*

Proof. By the argument above we can reduce the number of elements of S by two at one step preserving the isomorphism type of $\mathbf{D}(\mathbf{G}, S)$ until we reach either $\mathbf{D}(\mathbf{G}, \varnothing)$ or $\mathbf{D}(\mathbf{G}, \{u\})$ for some $u \in S$. Furthermore, by the same argument the models $\mathbf{D}(\mathbf{G}, \{u\})$ are isomorphic for all $u \in G$. Hence, the second condition implies the first one.

To prove the reverse implication it is now enough to show that $\mathbf{D}(\mathbf{G}, \varnothing)$ and $\mathbf{D}(\mathbf{G}, \{u\})$ are not isomorphic. Assume that $f: \mathbf{D}(\mathbf{G}, \{u\}) \to \mathbf{D}(\mathbf{G}, \varnothing)$ is an isomorphism with respect to the relation R. Then there are bijections $g: G \to G$ and $f_v: C \to C$, $v \in G$, such that $f((v, a)) = (g(v), f_v(a))$ for all $(v, a) \in D_G$. Clearly f_u is then an isomorphism $\mathbf{C}^- \to \mathbf{C}^+$, while for all other $v \in G$, f_v is an automorphism of $\mathbf{C}^+$. From Lemma 6.2, it follows that the total number of c, d-exchanges of f, $exc(f) = \Sigma_{v \in G}\, exc(f_v)$, is odd. However, this clearly means that f cannot preserve the edge relation E. ∎

Fix then some linear order $<^{\mathbf{G}}$ of the set G. We will include a partial order extending $<^{\mathbf{G}}$ to the structures we are going to use in the proof of our non-definability result.

8.3. Definition. Let u be the least element of G with respect to $<^{\mathbf{G}}$. We define now

- $\mathbf{A}(\mathbf{G}) = \langle A(\mathbf{G}), R^{\mathbf{A}(\mathbf{G})}, E^{\mathbf{A}(\mathbf{G})}, <^{\mathbf{A}(\mathbf{G})} \rangle$, and
- $\mathbf{B}(\mathbf{G}) = \langle B(\mathbf{G}), R^{\mathbf{B}(\mathbf{G})}, E^{\mathbf{B}(\mathbf{G})}, <^{\mathbf{B}(\mathbf{G})} \rangle$,

where $\langle A(\mathbf{G}), R^{\mathbf{A}(\mathbf{G})}, E^{\mathbf{A}(\mathbf{G})} \rangle = \mathbf{D}(\mathbf{G}, \varnothing)$, $\langle B(\mathbf{G}), R^{\mathbf{B}(\mathbf{G})}, E^{\mathbf{B}(\mathbf{G})} \rangle = \mathbf{D}(\mathbf{G}, \{u\})$, and $<^{\mathbf{A}(\mathbf{G})} = <^{\mathbf{B}(\mathbf{G})}$ is the relation

- $(v, a) <^{\mathbf{A}(\mathbf{G})} (w, b) \Leftrightarrow v <^{\mathbf{G}} w$ or $(v = w$ and $a \prec b)$

on the set $A(\mathbf{G}) = B(\mathbf{G}) = G \times C$.

Thus, $<^{\mathbf{A}(\mathbf{G})}$ is a partial order of width 2, and it can be thought of as a linear order of the set of all c, d-pairs $\{(v, c_i), (v, d_i)\}$, $v \in G$, $1 \leqslant i \leqslant n+1$.

By Lemma 8.2, the structures $\mathbf{A}(\mathbf{G})$ and $\mathbf{B}(\mathbf{G})$ are non-isomorphic, and their difference can be detected just by counting parities.

8.4. Lemma. *There is a* PTIME *computable Boolean query q such that $q(\mathbf{A}(\mathbf{G})) \neq q(\mathbf{B}(\mathbf{G}))$ for any finite connected graph $\mathbf{G}$ of degree $n+1$.*

Proof. Let $q(\mathbf{M}) = 1$ if and only if $\mathbf{M} \cong \mathbf{A}(\mathbf{H})$ for some finite connected graph $\mathbf{H}$ of degree $n+1$ and some choice of a linear order $<^{\mathbf{H}}$ (and the functions h_u, $u \in H$). Then, by definition, $q(\mathbf{A}(\mathbf{G})) = 1$ and $q(\mathbf{B}(\mathbf{G})) = 0$, whence it suffices to show that q is computable in PTIME.

Note first that given a finite structure $\mathbf{M}$, it is easy to check in polynomial time with respect to $|M|$ whether $\mathbf{M}$ is isomorphic to $\mathbf{A}(\mathbf{H})$ or $\mathbf{B}(\mathbf{H})$ for some connected graph $\mathbf{H}$ of degree $n+1$. If the answer is negative, we put $q(\mathbf{M}) = 0$. Otherwise we may assume without loss of generality that $\mathbf{M} = \mathbf{A}(\mathbf{H})$ or $\mathbf{M} = \mathbf{B}(\mathbf{H})$. Let U be an arbitrary subset of $M = H \times C$ such that U contains exactly one element of each c, d-pair (v, c_i), $(v, d_i) \in M$, and U is preserved by the edge relation E: $(a, b) \in E^{\mathbf{M}}$ implies $(a \in U \Leftrightarrow b \in U)$. Clearly we can produce such a set U in polynomial time. For each $v \in H$, the set U induces a unique tuple $\mathbf{a}_v = (a_1, ..., a_{n+1}) \in C^{n+1}$ such that $a_1 \prec \cdots \prec a_{n+1}$ and $(v, a_1), ..., (v, a_{n+1}) \in U$. Let S be the set of those $v \in H$ for which $\mathbf{a}_v \notin R^{\mathbf{M}}$. Now it is clear that $\langle M, R^{\mathbf{M}}, E^{\mathbf{M}} \rangle$ is isomorphic with $\mathbf{D}(\mathbf{H}, S)$, whence $q(\mathbf{M}) = 1$ if and only if $|S|$ is even. This completes the proof, since S is computable from U in polynomial time. ∎

Remark. As a matter of fact, the query q of the preceding lemma is already in LOGSPACE. Note however, that the argument above does not prove this, since the auxiliary relation U is not necessarily computable in logarithmic space, due to the requirement that U has to be preserved by the relation E. However, dropping this requirement and letting $q(\mathbf{M}) = 1$ if and only if the sum of $|S|$ and $1/2 \cdot |\{(a, b) \in E^{\mathbf{M}} \mid a \in U, b \notin U\}|$ is even, resolves this problem.

Our next aim is to show that if the graph $\mathbf{G}$ satisfies a suitable "largeness condition" with respect to $k > n$, then $\mathbf{A}(\mathbf{G})$ and $\mathbf{B}(\mathbf{G})$ are $\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)$-equivalent. Thus, we consider the n-bijective k-pebble game $BP^k_n(\mathbf{A}(\mathbf{G}), \mathbf{B}(\mathbf{G}))$. Let u be the least element of G with respect to the order $<^{\mathbf{G}}$. Assume that $f_i: A(\mathbf{G}) \to B(\mathbf{G})$ is a bijection played by II in this game. Then f_i must be an isomorphism with respect to the edge relation E and the partial order $<$, or else player I wins the game by choosing $D_i = \{a, b\}$ such that $f_i \upharpoonright \{a, b\}$ does not preserve E or $<$ (recall that $n \geqslant 2$). Especially, f_i must satisfy the condition

$$\{f_i((v, c_i)), f_i((v, d_i))\} = \{(v, c_i), (v, d_i)\}$$

for all $v \in G$ and $1 \leqslant i \leqslant n+1$, and thus, f_i induces for each $v \in G$ a bijection $f_{i,v}: C \to C$ such that $f_i((v, a)) = (v, f_{i,v}(a))$ for all $a \in C$. Moreover, if $(u, c_j) \in \operatorname{dom}(p_{i-1})$ (or $(u, d_j) \in \operatorname{dom}(p_{i-1})$) for some $1 \leqslant j \leqslant n+1$, then the bijection $f_{i,u}$ must be an isomorphism $\mathbf{C}^+ \to \mathbf{C}^-$, since otherwise player I wins the game by choosing $D_i = \{(u, c_l) \mid 1 \leqslant l \leqslant n+1, l \neq j\}$. In the same way we see that if $(v, a) \in \operatorname{dom}(p_{i-1})$, $v \neq u$, then $f_{i,v}$ must be an automorphism of $\mathbf{C}^+$. On the other hand, if f_i satisfies all these conditions, then it is clear that player I cannot win the game by his next move D_i. We say that f_i is a *good bijection* if this is the case.

A natural strategy for player II is thus to start with $f_0 = id_{G \times C}$, and then continue choosing good bijections $f_i: A(\mathbf{G}) \to B(\mathbf{G})$ in such a way that there is exactly one $v \in G$ for which $f_{i,v}$ is not an automorphism of $\mathbf{C}^+$ (or not an isomorphism $\mathbf{C}^+ \to \mathbf{C}^-$ if $v = u$). In the next round he can then move this bad part of f_i from v to some other vertex v' along a path $v = v_0, ..., v_m = v'$ of $\mathbf{G}$ by exchanging the c- and d-components of the elements corresponding to the edges $(v_l, v_{l+1}) \in E^{\mathbf{G}}$ on this path:

$$f_{i+1,w}(a)=\begin{cases} d_j, & \text{if } \quad w=v_l, f_{i,w}(a)=c_j, \\ & \text{and } j=h_{v_l}(v_{l+1}) \text{ or } j=h_{v_l}(v_{l-1}) \\ c_j, & \text{if } \quad w=v_l, f_{i,w}(a)=d_j, \\ & \text{and } j=h_{v_l}(v_{l+1}) \text{ or } j=h_{v_l}(v_{l-1}) \\ f_{i,w}(a), & \text{otherwise.} \end{cases}$$

It is easy to see that this new bijection f_{i+1} is good if f_i is, provided that $\mathrm{dom}(p_i)$ does not contain any of the exchanged elements or elements of the form (v', a).

This leads us to consider the following *cops&robber game* $CR_n^k(\mathbf{G})$ using the graph $\mathbf{G}$ as a board. Player I has k pebbles (cops) which he moves on the edges of $\mathbf{G}$, while player II has only one pebble (robber) which is always on some vertex of $\mathbf{G}$. Initially, the robber is on the vertex u, and the cops are not on the board. In each round of the game, player II starts by moving the robber along some path of $\mathbf{G}$; naturally the robber is not allowed to go through any edge containing a cop. Player I answers then by picking (at most) n of the cops and putting them on some edges of $\mathbf{G}$. Player I wins the game if after some round he has blocked all the escape routes of the robber with his cops, i.e., all the edges adjacent to the robber's vertex contain a cop. (If the robber is not surrounded by the cops, player II is allowed to keep it on the same vertex.) Player II wins the game if the cops do not capture the robber in a finite number of rounds.

The connection between the two games $CR_n^k(\mathbf{G})$ and $BP_n^k(\mathbf{A}(\mathbf{G}), \mathbf{B}(\mathbf{G}))$ should now be more or less clear: the k cops are used for marking the edges of $\mathbf{G}$ which correspond to the elements in $\mathrm{dom}(p_i)$ after round i in $BP_n^k(\mathbf{A}(\mathbf{G}), \mathbf{B}(\mathbf{G}))$, and the robber marks the vertex $v \in G$ for which $f_{i,v}$ is the bad bijection. If the robber can escape the cops forever, then player II can keep choosing good bijections forever, and hence he wins the game $BP_n^k(\mathbf{A}(\mathbf{G}), \mathbf{B}(\mathbf{G}))$.

8.5. LEMMA. *If player II has a winning strategy in the game* $CR_n^k(\mathbf{G})$, *then he has a winning strategy in* $BP_n^k(\mathbf{A}(\mathbf{G}), \mathbf{B}(\mathbf{G}))$, *too.*

As a matter of fact, it is not difficult to see that the converse of Lemma 8.5 is also true. Hence the cops&robber game CR_n^k actually characterizes those graphs $\mathbf{G}$ for which the corresponding structures $\mathbf{A}(\mathbf{G})$ and $\mathbf{B}(\mathbf{G})$ are $\mathscr{L}^k_{\infty\omega}(\mathbf{Q}_n)$-equivalent.

We are now ready to prove the main result of this section:

8.6. THEOREM. *For each natural number n there is a* PTIME *computable Boolean query which is not definable in the logic* $\mathscr{L}^{\omega}_{\infty\omega}(\mathbf{Q}_n)$.

Proof. We will show that for each natural number $k > n$ there is a finite connected graph of degree $n+1$ such that player II has a winning strategy in $CR_n^k(\mathbf{G})$. The claim follows then from Lemmas 8.5 and 8.4 and Theorem 5.10.

Let $\mathbf{H}$ be a finite connected graph with $m \geqslant 2k+2$ elements such that the degree of each vertex is n. Assume further that $\mathbf{H}$ remains connected when less than n of its edges are removed. Let $\mathbf{H}_i = \langle H_i, E^{\mathbf{H}_i}\rangle$ be mutually disjoint isomorphic copies of $\mathbf{H}$ for $i \leqslant m$, and let $H_i = \{v_{ij} \mid j \leqslant m,\ j \neq i\}$ for each $i \leqslant m$. The graph $\mathbf{G} = \langle G, E^{\mathbf{G}}\rangle$ consists of these copies of $\mathbf{H}$ together with some additional edges connecting them:

- $G = \bigcup_{i \leqslant m} H_i = \{v_{ij} \mid i, j \leqslant m,\ i \neq j\}$,
- $E^{\mathbf{G}} = \bigcup_{i \leqslant m} E^{\mathbf{H}_i} \cup \{(v_{ij}, v_{ji}) \mid i, j \leqslant m,\ i \neq j\}$.

Thus, for each pair $i < j \leqslant m$ there is exactly one edge connecting $\mathbf{H}_i$ and $\mathbf{H}_j$, and similarly for each vertex $a \in H_i$ there is exactly one $j \leqslant m$ and $b \in H_j$ such that $j \neq i$ and $(a, b) \in E^{\mathbf{G}}$. In particular, $\mathbf{G}$ is a connected graph of degree $n+1$ with $m(m+1)$ elements.

Consider then the game $CR_n^k(\mathbf{G})$. We say that a vertex v_{ij} is *safe* in a position of the game if no cop is on an edge adjacent to a vertex in H_i or H_j. Since player I has only k cops and $m \geqslant 2k+2$, there is a safe vertex in every possible situation of the game. Assume that player II has just put the robber on a safe vertex v_{ij}. Player I moves then at most n of the cops to new positions on the edges of $\mathbf{G}$, and there are two possibilities. Either there are less than n cops on the edges of $\mathbf{H}_i$, or there are still no cops on edges adjacent to the vertices of $\mathbf{H}_j$. In the first case, by our assumption on $\mathbf{H}$, there is a path not containing cops from v_{ij} to any vertex in H_i, while in the second case there is such a path from v_{ij} to any vertex in H_j. In both cases it is clear that in the next round player II can move the robber to another safe vertex. Thus, we conclude that player II wins the game if he always puts the robber on a safe vertex. ▮

8.7. COROLLARY. *There does not exist a finite set* $\mathbf{Q}$ *of generalized quantifiers such that* $\mathrm{PTIME} \equiv_{\mathscr{F}} \mathrm{IFP}(\mathbf{Q})$.

Note that we have actually proved an optimal result in terms of width of partial order. If for each natural number r, $\mathscr{O}_r$ denotes the class of all finite structures containing a partial order of width at most r, then the corollary above holds even if the class $\mathscr{F}$ of all finite structures is replaced with $\mathscr{O}_2$, while $\mathrm{PTIME} \equiv_{\mathscr{O}_1} \mathrm{IFP}$ is just a restatement of the Immerman–Vardi theorem.

Theorem 8.6 also implies a strong hierarchy result for generalized quantifiers on finite structures: for each n, there are $(n+1)$-ary quantifiers which are not definable in the logic $\mathscr{L}^{\omega}_{\infty\omega}(\mathbf{Q}_n)$. Indeed, the quantifier Q_q, where q is the query of Lemma 8.4, is an example. However, this example is not particularly natural, although it is computationally feasible. Thus, it is of interest to prove the hierarchy result using more concrete (and familiar) quantifiers. Let $h\colon \omega \to \omega$ be the function $h(m) = \lfloor m/2 \rfloor$, and let Q_h^n denote the nth Ramsey quantifier corresponding to this function (see Example 3.3(d)).

8.8. Corollary. *For each $n \geqslant 2$ the Ramsey quantifier Q_h^{n+1} is not definable in the logic $\mathscr{L}^{\omega}_{\infty\omega}(\mathbf{Q}_n)$.*

Proof. It is clearly enough to give a sentence φ of $\mathrm{FO}(Q_h^{n+1})$ such that $\mathbf{A}(\mathbf{G}) \models \varphi$ and $\mathbf{B}(\mathbf{G}) \not\models \varphi$ for any finite connected graph $\mathbf{G}$ of degree $n+1$.

Let $\theta(x, y)$ be a formula stating that some $(n+1)$-tuple $\mathbf{z}$ containing x and y is in the relation R, and let $\eta(x, y)$ be the formula $\neg(x<y) \wedge \neg(y<x)$. Thus, intuitively $\theta(x, y)$ says (for structures of the form $\mathbf{A}(\mathbf{G})$ or $\mathbf{B}(\mathbf{G})$) that x and y are in the same copy of the set C, and $\eta(x, y)$ says that x and y are in the same c, d-pair. We define now φ to be the sentence $Q_h^{n+1} x_1, ..., x_{n+1}\, \psi(x_1, ..., x_{n+1})$, where $\psi(x_1, ..., x_{n+1})$ is the conjunction of the formulas

- $\neg\eta(x_1, x_2)$,
- $\left(\bigwedge_{1 \leqslant i<j \leqslant n+1} \theta(x_i, x_j) \wedge \bigwedge_{1 \leqslant i<j \leqslant n+1} (x_i < x_j)\right) \rightarrow R(x_1, ..., x_{n+1})$, and
- $\forall z(E(x_1, z) \wedge \eta(z, x_2) \rightarrow z = x_2))$.

Consider then the structures $\mathbf{A}(\mathbf{G})$ and $\mathbf{B}(\mathbf{G})$ for some finite connected graph $\mathbf{G}$ of degree $n+1$. Let U be the subset $G \times P$ of $A(\mathbf{G}) = G \times C$. Clearly U contains exactly half of the elements of $A(\mathbf{G})$ and $(a_1, ..., a_{n+1}) \in \psi^{\mathbf{A}(\mathbf{G})}$ for any distinct elements $a_1, ..., a_{n+1} \in U$, whence $\mathbf{A}(\mathbf{G}) \models \varphi$. On the other hand, if $V \subseteq B(\mathbf{G})$ is homogeneous for the relation $\psi^{\mathbf{B}(\mathbf{G})}$, then, by the first conjunct of ψ, $|V| < h(|B(\mathbf{G})|)$ unless V contains exactly one element of each c, d-pair. If this would be the case, then the second conjunct of ψ would imply that V contains an odd number of elements of the type (v, d_i). However, this is impossible, since, by the third conjunct of ψ, there are no E-edges between elements of V and $B(\mathbf{G}) \setminus V$. Hence, we conclude that $\mathbf{B}(\mathbf{G}) \not\models \varphi$. ∎

9. CONCLUDING REMARKS

In this paper we have proved a two-step hierarchy theorem for the class of polynomial time computable queries in terms of arity of generalized quantifiers: PTIME cannot be captured by adding any generalized quantifiers of bounded arity to fixpoint logic, and, on the other hand, fixpoint logic is already too strong to be captured by adding quantifiers of bounded arity to first-order logic. In particular, it is impossible to find a finite set $\mathbf{Q}$ of quantifiers such that $\mathrm{PTIME} \equiv_{\mathscr{F}} \mathrm{IFP}(\mathbf{Q})$, or such that $\mathrm{LFP} \equiv_{\mathscr{F}} \mathrm{FO}(\mathbf{Q})$.

Both of these results are proved by giving, for each natural number n, a concrete PTIME computable query q on τ-structures and then using bijective Ehrenfeucht–Fraïssé games to show that q is not definable in terms of n-ary quantifiers. In both cases the vocabulary τ of q depends on n: there is an $(n+1)$-ary relation symbol in τ. This non-uniformity of vocabulary in Theorems 7.1 and 8.6 is unavoidable, since, for any vocabulary σ containing at most n-ary relation symbols, all queries on σ-structures are definable in $\mathrm{FO}(\mathbf{Q}_n)$ (see the discussion in the end of Section 3).

Nevertheless, it makes sense to ask whether Corollary 8.7 holds for uniform vocabularies, i.e., whether we could prove that for some (or, for every) fixed vocabulary σ there does not exist any finite set $\mathbf{Q}$ of quantifiers such that $\mathrm{PTIME} \equiv_{\mathscr{F}[\sigma]} \mathrm{IFP}(\mathbf{Q})$, where $\mathscr{F}[\sigma]$ is the class of all finite σ-structures. By the remark above, it is clear that the method based on bijective Ehrenfeucht–Fraïssé games does not help in solving problems of this type. However, this question is settled in Dawar and Hella (1995) with a completely different method.

Another interesting problem is, what happens to our hierachy results in the presence of linear order. Of course, it is not possible to prove Theorem 8.6 or even Corollary 8.7 for the class $\mathscr{O}$ of ordered finite structures, since already IFP, without any additional quantifiers, captures PTIME on $\mathscr{O}$. On the other hand, it is conceivable that Theorem 7.1, or at least Corollary 7.2, remains true on the class $\mathscr{O}$. Note however, that any element of an ordered finite structure is uniquely determined by the number of its predecessors, and hence $\mathbf{A} \equiv_1^{m+1} \mathbf{B}$ implies $\mathbf{A} \cong \mathbf{B}$, whenever $\mathbf{A}, \mathbf{B} \in \mathscr{O}$ are τ-structures such that all relations in τ are at most m-ary. Thus, bijective Ehrenfeucht–Fraïssé games cannot be used to prove any non-definability results on ordered structures whatsoever, and consequently, we leave the problem whether $\mathrm{PTIME} \equiv_{\mathscr{O}} \mathrm{FO}(\mathbf{Q})$ for some finite set $\mathbf{Q}$ of quantifiers open.

Although we have established that PTIME cannot be captured by a finite set of generalized quantifiers, our results do not rule out the possibility of finding some uniformly defined sequence $Q_0, Q_1, ...$ of quantifiers (with increasing arities) such that $\mathrm{PTIME} \equiv_{\mathscr{F}} \mathrm{FO}(\{Q_n \mid n \in \omega\})$. Indeed, partially ordering first-order quantifier prefixes leads to the uniformly defined set $\mathbf{H}$ of Henkin quantifiers, which captures NP on the class of all finite structures (see Example 3.3(e)). Hence, if $\mathrm{PTIME} = \mathrm{NP}$, then $\mathrm{PTIME} \equiv_{\mathscr{F}} \mathrm{FO}(\mathbf{H})$ (and vice versa).

Besides partially ordering quantifier prefixes, there are many other ways of defining uniform sequences of quantifiers. Ramseyfication (Example 3.3(d)) is another way, but in general it leads to NP-hard quantifiers. Perhaps the most natural and interesting way of forming a sequence of quantifiers from a given one is *resumption*, or *vectorization*. If $Q = Q_q$ is a quantifier of vocabulary $\tau = \langle R_1, ..., R_m \rangle$, with $\mathrm{ar}(R_i) = n_i$, then its rth resumption Q^r is defined by the query q^r such that

$$q^r(\langle A, S_1, ..., S_m \rangle) = q(\langle A^r, S_1{}^{(r)}, ..., S_m{}^{(r)} \rangle),$$

where $\mathrm{ar}(S_i) = r \cdot n_i$ and $S_i{}^{(r)} = \{(\mathbf{a}_1, ..., \mathbf{a}_{n_i}) \in (A^r)^{n_i} \mid \mathbf{a}_1{}^\frown \cdots {}^\frown \mathbf{a}_{n_i} \in S_i\}$, for each $1 \leqslant i \leqslant m$. It is easy to see that if Q is a PTIME computable quantifier, then so is Q^r for each r. Hence, it is natural to pose the question

Does there exist a quantifier Q such that

$$\text{PTIME} \equiv_{\mathscr{F}} \text{FO}(\{Q^r \mid r \in \omega\})?$$

As a matter of fact, it has recently turned out that this is just a restatement of Question 2.2: by a result of Dawar (1995), if there exists any Gurevich logic $\mathscr{L}$ at all capturing PTIME, then there is one of the form $\text{FO}(\{Q^r \mid r \in \omega\})$.

ACKNOWLEDGMENTS

I am grateful to Kerkko Luosto and Jouko Väänänen for their useful comments and suggestions on the manuscript of this paper. I also give my special thanks to Phokion Kolaitis, from whom I have learned everything I know about fixpoint logics, DATALOG, and finite variable logics.

Received September 9, 1994; final manuscript received April 8, 1996

REFERENCES

Abiteboul, S., and Vianu, V. (1991), Generic computation and its complexity, *in* "Proceedings, 23rd ACM Symposium on Theory of Computing," pp. 209–219.

Blass, A., and Gurevich, Y. (1986), Henkin quantifiers and complete problems, *Ann. Pure Appl. Logic* **32**, 1–16.

Cai, J., Fürer, M., and Immerman, N. (1992), An optimal lower bound on the number of variables for graph identification, *Combinatorica* **12**, 389–410.

Chandra, A., and Harel, D. (1982), Structure and complexity of relational queries, *J. Comput. System Sci.* **25**, 99–128.

Dawar, A. (1995), Generalized quantifiers and logical reducibilities, *J. Logic and Comput.* **5**, 213–226.

Dawar, A., and Hella, L. (1995), The expressive power of finitely many generalized quantifiers, *Inform. and Comput.* **123**, 172–184.

Dawar, A., Lindell, S., and Weinstein, S. (1995), Infinitary logic and inductive definability over finite structures, *Inform. and Comput.* **119**, 160–175.

Ebbinghaus, H.-D. (1985), Extended logics: the general framework, *in* "Model-Theoretic Logics" (J. Barwise and S. Feferman, Eds.), pp. 25–76, Springer-Verlag, Berlin/New York.

Fagin, R. (1974), Generalized first-order spectra and polynomial-time recognizable sets, *in* "Complexity of Computation" (R. M. Karp, Ed.), SIAM–AMS Proceedings, Vol. 7, pp. 43–73.

Gottlob, G. (1995), Relativized logspace and generalized quantifiers over finite structures, *in* "Proceedings, 10th IEEE Symposium on Logic in Computer Science," pp. 65–78.

Grädel, E. (1990), On logical descriptions of some concepts in structural complexity theory, *in* "Proceedings, 3rd Workshop on Computer Science Logic," Lecture Notes in Computer Science, Vol. 440, pp. 163–175, Springer-Verlag, Berlin/New York.

Gurevich, Y. (1988), Logic and the challenge of computer science, *in* "Trends in theoretical computer science" (E. Börger, Ed.), pp. 1–57, Comput. Sci. Press, New York.

Gurevich, Y., and Shelah, S. (1986), Fixed-point extensions of first-order logic, *Ann. Pure Appl. Logic* **32**, 265–280.

Hella, L. (1989), Definability hierarchies of generalized quantifiers, *Ann. Pure Appl. Logic* **43**, 235–271.

Hella, L., and Sandu, G. (1995), Partially ordered connectives and finite graphs, *in* "Quantifiers: Logics, Models and Computation, Volume II" (M. Krynicki *et al.*, Eds.), pp. 79–88, Kluwer, Dordrecht.

Henkin, L. (1961), Some remarks on infinitely long formulas, *in* "Infinitistic Methods: Proceedings of the Symposium on Foundations of Mathematics," pp. 167–183, Państwowe Wydawnictwo Naukowe, Warsaw.

Immerman, N. (1986), Relational queries computable in polynomial time, *Inform. and Control* **68**, 86–104.

Immerman, N. (1987), Languages which capture complexity classes, *SIAM J. Comput.* **16**, 760–778.

Immerman, N., and Lander, E. (1990), Describing graphs: a first-order approach to graph canonization, *in* "Complexity Theory Retrospective" (A. Selman, Ed.), pp. 59–81, Springer-Verlag, Berlin/New York.

Kolaitis, Ph. G. (1991), The expressive power of stratified logic programs, *Inform. and Comput.* **90**, 50–66.

Kolaitis, Ph. G., and Väänänen, J. (1995), Generalized quantifiers and pebble games on finite structures, *Ann. Pure Appl. Logic.* **74**, 23–75.

Kolaitis, Ph. G., and Vardi, M. (1992a), Infinitary logics and 0–1 laws, *Inform. and Comput.* **98**, 258–294.

Kolaitis, Ph. G., and Vardi, M. (1992b), Fixpoint logic vs. infinitary logic in finite-model theory, *in* "Proceedings, 7th IEEE Symposium on Logic in Computer Science," pp. 46–57.

Lindström, P. (1966), First order predicate logic with generalized quantifiers, *Theoria* **32**, 186–195.

Makowsky, J. A., and Pnueli, Y. B. (1994), Oracles and quantifiers, *in* "Proceedings, 7th Workshop on Computer Science Logic," Lecture Notes in Computer Science, Vol. 832, pp. 189–222, Springer-Verlag, Berlin/New York.

Makowsky, J. A., and Pnueli, Y. B. (1995), Logics capturing relativized complexity classes uniformly, *in* "Proceedings, Logic and Computations Complexity" (D. Leivant, Ed.), Lecture Notes in Computer Science, Vol. 960, pp. 463–479, Springer-Verlag, Berlin/New York.

Moschovakis, Y. N. (1974), "Elementary Induction on Abstract Structures," North-Holland, Amsterdam.

Mostowski, A. (1957), On a generalization of quantifiers, *Fundam. Math.* **44**, 12–36.

Väänänen, J. (1986), A hierarchy theorem for Lindström quantifiers, *in* "Logic and abstraction" (M. Furberg *et al.*, Ed.), Acta Philosophica Gothoburgensia, Vol. 1, pp. 317–323.

Vardi, M. (1982), The complexity of relational query languages, *in* "Proceedings, 14th ACM Symposium on Theory of Computing," pp. 137–146.

Walkoe, W. J. (1970), Finite partially-ordered quantification, *J. Symbolic Logic* **35**, 535–555.