Note di Matematica Vol. 18 - n. 2, 191-212 (1998)

FIXED POINTS IN RELATIONAL ALGEBRA

S. LEONESI, S. TULIPANI

Abstract. *The relational algebra system for queries on databases is surveyed. By the use of partial isomorphisms the limited expressive power of this formalism is established. Then, the fixed point operation is added to build new relations as solutions of particular systems of formulas. Moreover, the search for solutions is related to the usual logic programming machinery.*

1 Introduction.

According to the model proposed by Codd in the early 1970s (see [Co70], [Co72]), a database can be viewed as a collection of relations on a fixed finite domain. The query languages developed in this framework (see [Ma83], [Ul82]) are based on the relational algebra operations of union, intersection, difference, Cartesian product, projection and selection; these operations enable to compute new relations from the basic stored relations. However, some significant relation constructions do not fit in this schema. Among them is the transitive closure R^*, of a given relation R, which cannot be computed in a uniform way with mere operations of the relational algebra (see §5). The relation R^* can be built by the minimum fixed point of a suitable set-operator.

Given a set S, a fixed point of a function $f : S \to S$ is a $x \in S$ such that $f(x) = x$. Such points arose for functions defined on topological spaces and in partially ordered sets [Bi79]; significant applications were given to Analysis, to Topology and to Computer Science.

Here we consider fixed points of functions, called also operators,

$$F : \mathcal{P}(S) \to \mathcal{P}(S) ,$$

defined on a given set S; our purpose is to use fixed points for building relations.

The main aim of this paper is to present the relational algebra together with the fixed point operations. A *relational structure* $\mathcal{A} = (A, R_1, R_2, \dots, R_k)$ is given by a basic set A, called also domain, and a k−tuple of relations. For all $1 \leq i \leq k$, each R_i has a given number n_i of arguments; this means that $R_i \subseteq A^{n_i}$, in set notations. The sequence of non-negative integers $(n_1, n_2, \dots, n_k)$ will be called *signature* of the structure $\mathcal{A}$. Instead of considering only the relational algebra on a single structure $\mathcal{A}$, we are concerned with relational formulas in order to define relations in a uniform way on the whole class DB_σ of all the finite structures of a fixed signature $\sigma = (n_1, n_2, \dots, n_k)$. The relational formulas allow to think of a database not only as a collection of facts, but also as a set of rules for deducing new facts by means of calculation; the answer to a query on a database may require a deduction computation (see [GM84], [Ul82], [Da75]) rather than a simple retrieval of stored information. The expressive power of formulas based on the mere relational algebra is inadequate for relational queries;

here we present an extension of this formalism by introducing certain systems of formulas, called triangular systems. Such systems define in a uniform way suitable relations on every given member of DB_σ; thus, a query language for deductive databases can be based on triangular systems of formulas.

Finally, we will point out how to reduce the computation for the solution of special triangular systems, where the difference operation is not present, to the resolution of a goal in logic programming.

2 Fixed Points.

In this section we consider fixed points of special functions, called also *operators*, which are defined on the power set of a given set.

2.1. Definition. Let S be a set and $F : \mathcal{P}(S) \to \mathcal{P}(S)$ be a function. We say that the operator F is *monotone* if for all $X \subseteq Y \subseteq S$ it follows $F(X) \subseteq F(Y)$. We say that F is *continuous* if for every sequence $\{X_i\}_{i\in \mathbb{N}}$ of subsets of S such that $X_1 \subseteq X_2 \subseteq \cdots \subseteq X_n \subseteq \cdots$ it is

$$F\left(\bigcup_n X_n\right) = \bigcup_n F(X_n).$$

□

2.2. Remarks.

a) If F is continuous, then F is monotone. To this end let us consider $X, Y \in \mathcal{P}(S)$ such that

$$X \subseteq Y; \tag{1}$$

then (1) is a particular increasing sequence; hence, by the continuity of F, it follows

$$F(Y) = F(X \cup Y) = F(X) \cup F(Y);$$

this means that $F(X) \subseteq F(Y)$.

b) The other direction is not always true. We give a counterexample of a monotone F which is not continuous. Let us consider the operator $F : \mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{N})$ defined, for $X \subseteq \mathbb{N}$ by

$$F(X) = \begin{cases} \mathbb{N} & \text{if } X \text{ is infinite} \\ \emptyset & \text{if } X \text{ is finite.} \end{cases}$$

Clearly, F is monotone. Let us take, now, the sequence $\{X_n\}_{n\in\mathbb{N}}$ such that X_n is the (finite) interval $[0,n]$. Then $\bigcup_n X_n = \mathbb{N}$, hence $F\left(\bigcup_n X_n\right) = \mathbb{N}$. But, $\bigcup_{n\in\mathbb{N}} F(X_n) = \emptyset$ since every X_n is finite. This proves that F is not continuous. □

2.3. Definitions.

a) Let $F : \mathcal{P}(S) \to \mathcal{P}(S)$ be given. A set $X \in \mathcal{P}(S)$ is said to be *fixed point* for F if $F(X) = X$. Moreover, a fixed point X_0 will be said *minimum fixed point* for F if for every other fixed point $X_1 \in \mathcal{P}(S)$ it is $X_0 \subseteq X_1$.

b) X is said to be *pre-fixed point* for F if $F(X) \subseteq X$.

c) X is said to be *post-fixed point* for F if $X \subseteq F(X)$. □

Now, let us recall a classic result by Tarski [Ta55] which was previously proved in a particular case by Knaster [Kn28].

2.4. Theorem. *Let $F : \mathcal{P}(S) \to \mathcal{P}(S)$ be a monotone operator. Then F has a minimum fixed point M_0 and a maximum fixed point M_1; moreover,*

$$\begin{array}{ll} (a) & M_0 = \bigcap\{X : F(X) \subseteq X\} = \bigcap\{X : F(X) = X\} \\ (b) & M_1 = \bigcup\{X : X \subseteq F(X)\} = \bigcup\{X : F(X) = X\}. \end{array}$$

Proof. (a). Let $L = \{X : F(X) \subseteq X\}$ and define

$$M_0 = \bigcap L. \tag{1}$$

First we prove that $M_0 \in L$. For every $X \in L$ it is $M_0 \subseteq X$ by the definition of M_0. Then, by the monotony of F, $F(M_0) \subseteq F(X) \subseteq X$ for every $X \in L$. Therefore,

$$F(M_0) \subseteq \bigcap L = M_0 \tag{2}$$

which means that $M_0 \in L$. Now, again by monotony, $F(F(M_0)) \subseteq F(M_0)$. This means that $F(M_0) \in L$. Therefore, by (1) we have that

$$M_0 \subseteq F(M_0). \tag{3}$$

Thus, by (2) and (3), it follows that M_0 is a fixed point for F. Finally, define $M_0' = \bigcap\{X : F(X) = X\}$. Since M_0 is a fixed point, we have that $M_0' \subseteq M_0$. On the other hand, $\{X : F(X) = X\} \subseteq \{X : F(X) \subseteq X\}$ which implies that $M_0 \subseteq M_0'$. Thus, $M_0 = M_0'$.

(b). The proof is quite analogous to the previous one. □

Observe that the previous theorem guarantees the existence of the minimum fixed point for a monotone operator, however it does not give a method to determine it. The next lemma and its corollary indicate how to compute effectively the minimum fixed point for continuous operators.

2.5. Lemma. *Let $F : \mathcal{P}(S) \to \mathcal{P}(S)$ be a continuous, hence monotone, operator. Then, for every post-fixed point J for F there exists a minimum fixed point J_ω for F such that $J \subseteq J_\omega$. Moreover,*

$$J_\omega = \bigcup_{n \in \mathbb{N}} F^n(J),$$

where $F^0(J) = J$ and, inductively, $F^{n+1}(J) = F(F^n(J))$ for every $n \in \mathbb{N}$.

Proof. First observe that $F^0(J) = J \subseteq F(J) = F^1(J)$. Hence, by the monotony of F and by induction we get $F^n(J) \subseteq F^{n+1}(J)$ for every $n \in \mathbb{N}$. Now, by the continuity of F, it follows

$$F(J_\omega) = F\left(\bigcup_{n\in\mathbb{N}} F^n(J)\right) = \bigcup_{n\in\mathbb{N}} F^{n+1}(J) = J_\omega; \tag{1}$$

Therefore J_ω is a fixed point containing J. Finally, let I be any fixed point for F, such that $J \subseteq I$; hence, $F^0(J) = J \subseteq I$. Then, by $F(I) = I$, by the monotony of F and by induction we get $F^n(J) \subseteq I$ for every $n \in \mathbb{N}$. Therefore, $J_\omega = \bigcup_n F^n(J) \subseteq I$. This proves that J_ω is the minimum fixed point for F which contains J. □

2.6. Corollary. *Let $F : \mathcal{P}(S) \to \mathcal{P}(S)$ be a continuous operator. Then there exists the minimum fixed point M_0 for F; moreover,*

$$M_0 = \bigcup_{n\in\mathbb{N}} F^n(\emptyset).$$

Proof. Since $\emptyset$ is clearly a post-fixed point for F, the result follows from Lemma 2.5. □

2.7. Examples.

1) Let $\mathcal{G} = (G, \cdot, ^{-1}, \mathbb{1})$ be a group and $F : \mathcal{P}(\mathcal{G}) \to \mathcal{P}(\mathcal{G})$ be the operator defined by

$$F(X) = X \cup \{a \cdot b : a, b \in X\} \cup \{a^{-1} : a \in X\} \cup \{\mathbb{1}\}.$$

Then F is continuous. Observe that every fixed point for F is a subgroup of G and viceversa.

2) Let $F : \mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{N})$ be defined by

$$F(X) = \{n + 2 : n \in X\} \cup \{0\}.$$

Then F is continuous; the minimum fixed point is the set of even numbers.

3) Let $a \in \mathbb{N}$ be given and $F : \mathcal{P}(\mathbb{N}) \to \mathcal{P}(\mathbb{N})$ be defined by

$$F(X) = \{a \cdot b : b \in X\} \cup \{1\}.$$

Then F is continuous; the minimum fixed point is the set of powers of a.

4) Let $F : \mathcal{P}(\mathbb{N}^2) \to \mathcal{P}(\mathbb{N}^2)$ be defined by

$$F(X) = \{(n, n \cdot b) : (n-1, b) \in X\} \cup \{(0, 1)\}.$$

Then F is continuous; the minimum fixed point is the graph of the function $n \mapsto n!$ for $n \in \mathbb{N}$.

5) Let S be a Euclidean space and $F : \mathcal{P}(S) \to \mathcal{P}(S)$ be defined by

$$F(X) = \{y \in S : \exists p, q \in X, \quad \text{such that} \quad y \in [p,q]\}$$

where $[p,q]$ denotes the closed segment delimited by the points p, q. Then F is continuous; its fixed points are all the convex subsets of S.

6) Let (S,R) be a graph, where S is the vertex set and R is the adjacency relation. Let $F : \mathcal{P}(S) \to \mathcal{P}(S)$ be defined by

$$F(X) = \{b \in S : \exists a \in X, \ (a,b) \in R\}.$$

Then F is continuous; the fixed points for F are the empty set and the subgraphs which are union of connected components of (S,R).

3 Relational Algebra

A *database* is a collection of more or less permanent data which can be stored, retrieved or elaborated by the computer. The need for efficiently managing large amounts of data and for helping the user to conceptualize the organization of the database, the data must be structured along a certain database–model. A *database model* consists basically of two features:

1. A mathematical notation to represent data and connections between them;

2. A collection of operations on data to describe queries and other managements.

In our treatment we will take into consideration the *relational model* proposed by Codd (see [Co70], Co72]) in the early 1970s. This paradigm not only allows to retrieve stored relations but enables the computation of new relations which can be described by a collection of operations on the basic stored relations. The formalism is called *relational algebra*; on this schema various query languages were designed to recover information efficiently.

Given a positive integer n, an *n-ary relation* R over a set A, set-theoretically speaking, is a subset $R \subseteq A^n$. The number n is called *arity* or number of arguments of R; it will be denoted by $ar(R)$. If $a_1, a_2, \ldots, a_n$ is a sequence belonging to R, we will write $Ra_1a_2 \cdots a_n$ or shortly $R\vec{a}$ rather than using the set-theoretic notation $(a_1, a_2, \ldots, a_n) \in R$.

3.1. Definition. A *relational structure* $\mathcal{A}$ is a sequence $(A, R_1, R_2, \ldots, R_k)$ where A is a set, called domain or basic set, and $R_1, R_2, \ldots, R_k$ are relations on A. The sequence $(n_1, n_2, \ldots, n_k)$ of arities of $R_1, R_2, \ldots, R_k$, respectively, is called the *signature* of $\mathcal{A}$. When the domain A is finite the structure $\mathcal{A}$ will be called *relational database*. □

Often databases are dinamical structures where a priori not all the elements of A are known. Therefore, it is convenient to fix also a sequence $c_1, c_2, \ldots, c_k \in A$ of elements whose name appear in the query languages. For technical simplification we will avoid the use of constants; the treatment with them is substantially identical.

3.2. Definition. Given a relational structure $\mathcal{A}$, the *relational algebra* generated by $\mathcal{A}$, in notation $Rel(\mathcal{A})$, is defined as the minimal collection of relations over A such that:

1) the relations $R_1, R_2, \ldots, R_k$ are in $Rel(\mathcal{A})$; they will be called *basic relations*.

2) $Rel(\mathcal{A})$ is closed under the Boolean operations of *union* $\cup$ and *difference* $\setminus$. Namely, if $R, S \in Rel(\mathcal{A})$ have the same arity, or in other words $R \subseteq A^n$ and $S \subseteq A^n$, then

$$R \cup S \in Rel(\mathcal{A}) \quad , \quad R \setminus S \in Rel(\mathcal{A}).$$

3) $Rel(\mathcal{A})$ is closed under *Cartesian product*. Namely, if $R, S \in Rel(\mathcal{A})$ and $R \subseteq A^n$, $S \subseteq A^m$ then

$$R \times S \in Rel(\mathcal{A}) \quad \text{and} \quad ar(R \times S) = n + m.$$

4) $Rel(\mathcal{A})$ is closed under *projections*. This means that, for every $n \in \mathbb{N}$, for every $R \in Rel(\mathcal{A})$ such that $R \subseteq A^n$ and every $J = \{i_1, i_2, \ldots, i_k\} \subseteq \{1, \ldots, n\}$ where $i_1 < i_2 < \cdots < i_k$, the relation $P_J(R) \subseteq A^k$ is in $Rel(\mathcal{A})$. This relation is the image of R under the projection map $P_J : A^n \to A^k$ defined by

$$P_J(a_1, a_2, \ldots, a_n) = (a_{i_1}, a_{i_2}, \ldots, a_{i_k}).$$

5) $Rel(\mathcal{A})$ is closed under the *selection operations* ζ^i_j for $i, j \in \mathbb{N}$ and $i < j$; here $\zeta^i_j R$ is defined, when $j \leq ar(R) = n$, by

$$\zeta^i_j R a_1 a_2 \cdots a_n \quad \text{if and only if} \quad (R a_1 a_2 \cdots a_n \quad \text{and} \quad a_i = a_j)$$

In other words ζ^i_j returns the intersection K of R with the subset of A^n, hyperplane, defined by the equation $x_i = x_j$. The Figure 1 pictures an example with $n = 2$. $\square$

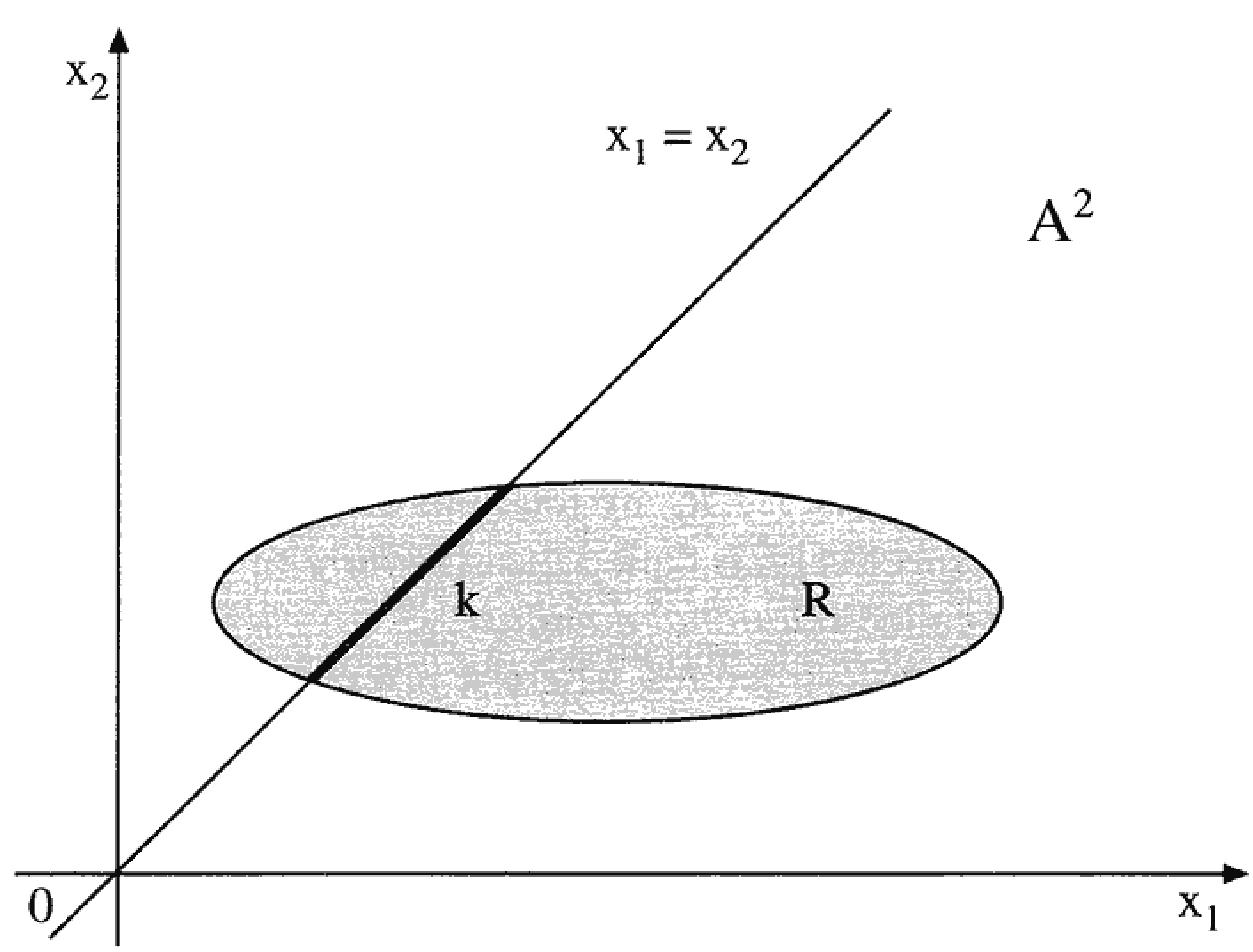

Figure 1

3.3. Remarks.

1) Note that the operation of intersection can be defined by the difference. In fact, if R, S have the same arity, then

$$R \cap S = R \setminus (R \setminus S).$$

Furthermore, all the Boolean operations can be performed in the relational algebra.

2) Observe that it is sufficient that $Rel(\mathcal{A})$ be closed only for the projections from $A^n \to A^{n-1}$, for every $n > 1$. Each such projection deletes a coordinate. In fact, for every $1 \leq i \leq n$, we may consider the projection $\pi_i : A^n \to A^{n-1}$ defined by

$$\pi_i(a_1, \ldots, a_i, \ldots, a_n) = (a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n).$$

It is easy to prove that, given $R \subseteq A^n$ and $P_J : A^n \to A^k$ defined as in 3.2.4, the relation $P_J(R)$ can be obtained from R by applying the $n-k$ projections π_i, for every $i \notin J$. From now on, we will use only the projections π_i defined above.

3) As we said before, here we do not make use of constants. They can be used to have additional selection operations. In fact, every $c \in A$ defines operations ε_i^c on $Rel(\mathcal{A})$ which are defined for every R of arity $n \geq i$ and every i by

$$\varepsilon_i^c R a_1 a_2 \cdots a_n \quad \text{if and only if} \quad (R a_1 a_2 \cdots a_n \quad \text{and} \quad a_i = c).$$

In other words the operation ε_i^c selects the subset K of A^n, hyperplane, defined by the equation $x_i = c$. The Figure 2 pictures an example with $n = 2$.

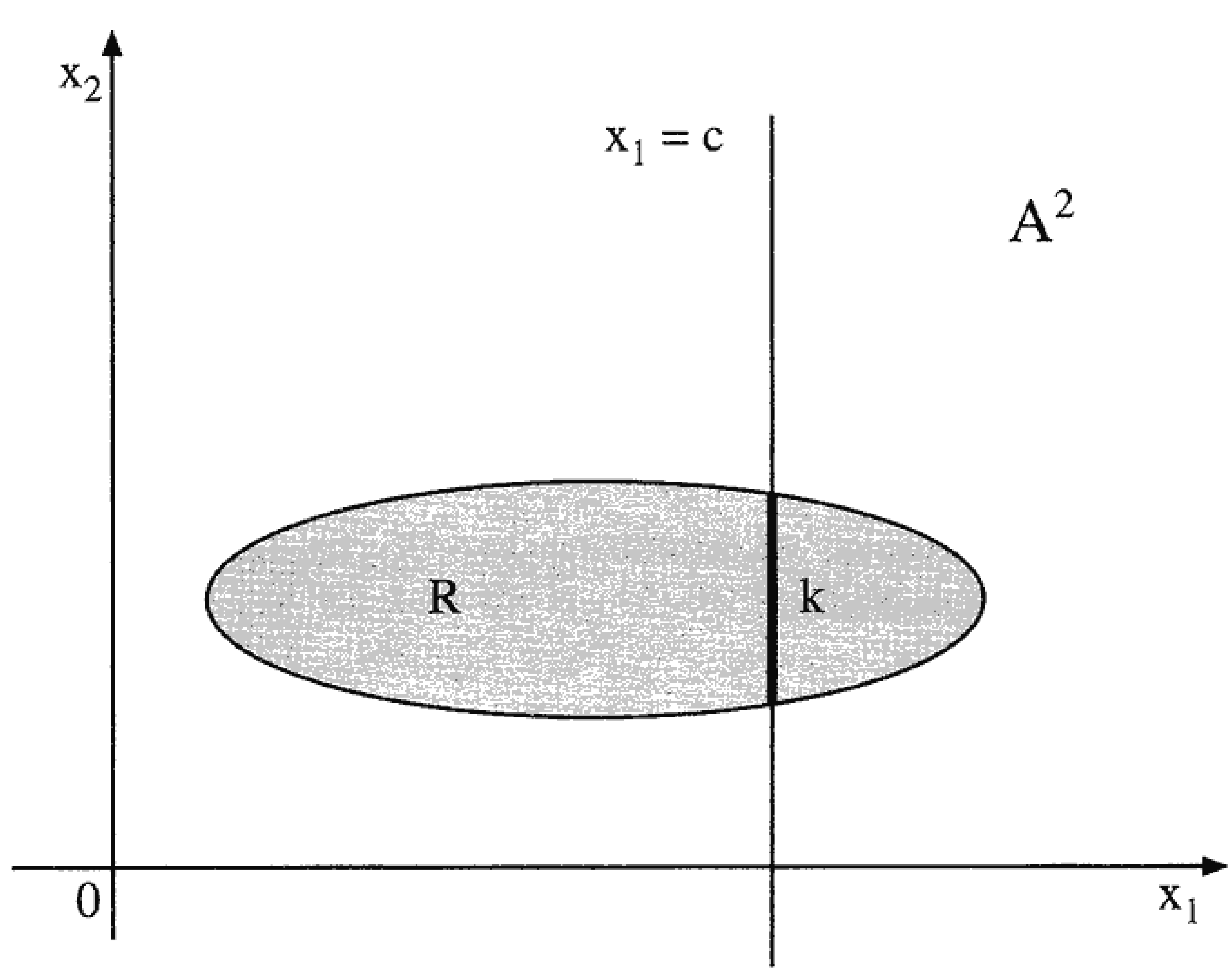

Figure 2

4) The *composition* operation $R \circ S$ between two binary relations is defined by

$$(R \circ S)ab \quad \text{if and only if} \quad (\exists c \in A : \ Rac \quad \text{and} \quad Scb).$$

Such operation can be obtained by the other operations already considered. In fact we have

$$R \circ S = \pi_2 \pi_3 \zeta_3^2 (R \times S).$$

□

4 Relational formulas

The reader might have realized that the relations in the relational algebra

$$Rel(A, R_1, R_2, \ldots, R_k)$$

are the interpretations of the first order logic formulas built on the equality symbol $=$ and on the atomic formulas which correspond to the symbols which interpret the basic relations $R_1, R_2, \ldots, R_k$. In fact, the operations $\cup$, $\cap$, $\setminus$ correspond to the Boolean connectives $\vee$, $\wedge$, $\neg$ and the projections corresponds to the existential quantification; this correspondence will be understood better after the Definition 4.4. Here we consider, rather than the symbolism of first order formulas, the quite analogous symbolism of relational formulas. The use of formulas to represent relations is useful to formulate queries and to treat collections of databases on the same signature.

4.1. Definition. We call *relational language* L_σ for a signature $\sigma = (n_1, n_2, \ldots, n_k)$ the following set of symbols:

1) $\{P_1, P_2, \ldots, P_k\}$, which are said symbols for the basic relations.

2) $\cup$, $\cap$, $\setminus$, $\times$, π_i, ζ_j^i for every $i, j \in \mathbb{N}$ with $i < j$. □

Now, we will define the set of formulas in the language L_σ. Each formula F is associated with an arity that we denote $ar(F)$.

4.2 Definition The set of formulas of signature $\sigma = (n_1, n_2, \ldots, n_k)$ is the minimum set of strings on the symbols of L_σ such that:

1) Every P_i is a formula and $ar(P_i) = n_i$, for $1 \leq i \leq n$. These are called also atomic formulas.

2) If F, G are formulas of arity n then $(F \cup G)$, $(F \cap G)$, $(F \setminus G)$ are formulas of arity n.

3) If F, G are formulas of arity m and n, respectively, then $(F \times G)$ is a formula of arity $m + n$.

4) If F is a formula of arity $n > 0$ then $\pi_i F$ is a formula of arity $n-1$, for every $1 \leq i \leq n$; moreover, $\zeta^i_j F$ is a formula of arity n, for every $1 \leq i < j \leq n$.

Note that it is possible to write formulas without parenthesis after a stipulation of a precedence in the symbols; this guarantees uniqueness of readability. □

4.3. Definition. Let DB_σ denote the class of all finite structures of signature σ; let n be an integer $n \geq 0$. We will call n-ary *global relation*, or *query*, on DB_σ any function Q defined on DB_σ such that

1) Q maps every $\mathcal{A} \in DB_\sigma$ on a relation $Q_\mathcal{A} \subseteq A^n$. In case $n = 0$, $Q_\mathcal{A}$ is a Boolean value (true or false) and Q is said *Boolean global relation.*

2) Q is invariant for isomorphism. This means that, for every isomorphism $f : \mathcal{A} \to \mathcal{B}$ with $\mathcal{A}$, $\mathcal{B} \in DB_\sigma$ and for every $a_1, a_2, \ldots, a_n \in A^n$,

$$Q_\mathcal{A} a_1 a_2 \cdots a_n \quad \text{if and only if} \quad Q_\mathcal{B} f(a_1) \cdots f(a_n).$$

□

4.4. Definition. Every relational formula F of arity n in the signature σ defines a global n-ary relation on DB_σ. The map associated with F is

$$\mathcal{A} \mapsto F^\mathcal{A},$$

where $F^\mathcal{A}$ is defined by induction according to the Definition 4.2 of F. In more details we have:

1) $F^\mathcal{A} = R_i$, if F is the atomic formula P_i for some $1 \leq i \leq k$.

2) $F^\mathcal{A} = F_1^\mathcal{A} \cup F_2^\mathcal{A}$, if F is the formula $F_1 \cup F_2$.

3) $F^\mathcal{A} = F_1^\mathcal{A} \setminus F_2^\mathcal{A}$, if F is the formula $F_1 \setminus F_2$.

4) $F^\mathcal{A} = F_1^\mathcal{A} \times F_2^\mathcal{A}$, if F is the formula $F_1 \times F_2$.

5) $(\pi_i F)^\mathcal{A}$, when $ar(F) = n > 0$ is the $n-1$-ary relation defined by

$$(\pi_i F)^\mathcal{A} a_1 a_2 \cdots a_{i-1} a_{i+1} \cdots a_n \qquad \text{if and only if}$$

$$\exists b \in A \quad \text{such that} \quad F^\mathcal{A} a_1 a_2 \cdots a_{i-1} b\, a_{i+1} \cdots a_n.$$

Observe that for $n = 1$ we have that $(\pi_1 F)^\mathcal{A}$ is *true* if $F^\mathcal{A}$ is not empty and *false* otherwise; in other words $\pi_1 F$ defines a Boolean global relation.

6) $(\zeta^i_j F)^\mathcal{A} a_1 a_2 \cdots a_n$ if and only if $(a_i = a_j$ and $F^\mathcal{A} a_1 a_2 \cdots a_n)$. □

All the symbols $\cup, \setminus, \times$ are interpreted in the corresponding set-theoretic operation. We did not give the interpretation for the symbol $\cap$ since, as we saw in 1) of 3.3, such an operation can be defined by $\setminus$. Moreover, we remark that every relation in $Rel(\mathcal{A})$, defined in 3.2, is the interpretation of a suitable relational formula. However, two different formulas F, G may have the same interpretation in some structure $\mathcal{A}$ and different interpretation in some other structure $\mathcal{B}$; namely, $F^{\mathcal{A}} = G^{\mathcal{A}}$ and $F^{\mathcal{B}} \neq G^{\mathcal{B}}$.

5 Expressive power of relational formulas

As we have seen, every relational formula on a given signature σ determines a global relation on the class of relational structures of signature σ. So, the set of formulas can be thought as a general query language for a database DB_σ. The complexity of answer time grows with the dimension of the formula F; a detailed analysis could establish that the complexity grows faster when the projection symbols in F increase.

Now, we will prove that there exist important global relations which are not definable by any relational formula (Proposition 5.2). Let us consider a signature σ with at least an arity of value 2; so, let P be a binary symbol in L_σ. Given $\mathcal{A} \in DB_\sigma$, let us denote by R the interpretation $P^{\mathcal{A}}$ of P in $\mathcal{A}$. Then, we consider the map T on DB_σ defined by

$$T : \mathcal{A} \to R^* \tag{1}$$

where R^* denote the *transitive closure* of the relation R. The relation R^* is defined by

$$R^*xy \quad \text{if and only if} \quad \exists n \exists z_0, \ldots, \exists z_n \quad x = z_0,\ y = z_n,\ Rz_{i-1}z_i, \quad \text{for} \quad i = 1, \ldots, n.$$

Hence, we have

$$R^* = \bigcup_{n \geq 1} R^n$$

where R^n is the composition "$\circ$" of R with itself n times; in other words R^n is defined inductively by

$$R^1 = R \quad \text{and} \quad R^{k+1} = R \circ R^k.$$

Since the domain A is finite there exist $k \in \mathbb{N}$, not greater than the cardinality of A, such that $R^* = R^k$. However, such k, dependent on $\mathcal{A}$, may be not bounded when $\mathcal{A}$ ranges in DB_σ. For this reason a relational formula which represents the global relation T may not exist. Before proving this fact we give an example.

5.1. Example. Let $\mathcal{A}_n$, for $n \in \mathbb{N}$, $n > 0$, be a structure on the domain $A_n = \{0, 1, \ldots, n\}$ with the binary relation, called R_n, of successor. Namely,

$$R_n = \{(i-1, i) :\ 1 \leq i \leq n\}$$

Then we have

$$\begin{array}{lll}
(0,1) \in R_n^1 & & \\
(0,2) \in R_n^2 & \text{since} & (0,1) \in R_n,\ (1,2) \in R_n \\
(0,3) \in R_n^3 & \text{since} & (0,2) \in R_n^2,\ (2,3) \in R_n \\
\ldots & \ldots & \\
(0,n) \in R_n^n & \text{since} & (0,n-1) \in R_n^{n-1},\ (n-1,n) \in R_n
\end{array}$$

With a similar proof we have that

$$R_n^n = \{(i,j) : 0 \leq i < j \leq n\} = R_n^*.$$

But $R_{n-1}^n \neq R_n^*$ since $(0,n) \notin R_n^{n-1}$. □

Now we prove a proposition which delimits the expressive power of the relational formulas.

5.2 Proposition. *Let σ be a signature with at least a binary symbol, say P. Let T be the global relation over DB_σ determined by the transitive closure of P. In other words, for every $\mathcal{A} \in DB_\sigma$*

$$T : \mathcal{A} \mapsto TC(P^{\mathcal{A}})$$

where $TC(P^{\mathcal{A}})$ denotes the transitive closure of the binary relation $P^{\mathcal{A}}$. Then there exist no relational formula F of signature σ such that $F^{\mathcal{A}} = TC(P^{\mathcal{A}})$ for every $\mathcal{A} \in DB_\sigma$. Before

the proof of this proposition we need the notion of partial isomorphism and two auxiliary lemmas.

5.3. Definition. Let $\mathcal{A}, \mathcal{B}$ be structures with same signature σ and basic sets A, B respectively. A *partial isomorphism* from $\mathcal{A}$ to $\mathcal{B}$ is a one-to-one map f from a subset of A, denoted by $dom(f)$, and a subset of B, denoted by $range(f)$, such that, for every basic relation $P \in L_\sigma$ and every $a_1, a_2, \ldots, a_n \in dom(f)$,

$$P^{\mathcal{A}} a_1 a_2 \cdots a_n \quad \text{if and only if} \quad P^{\mathcal{B}} f(a_1) \cdots f(a_n). \tag{1}$$

A partial isomorphism g *extends* f if $dom(f) \subseteq dom(g)$ and $f(x) = g(x)$ for every $x \in dom(f)$; in other words, the graph of g contains the graph of f.

Let $I_m, I_{m-1}, \ldots, I_0$ be a family of non empty sets of partial isomorphisms from $\mathcal{A}$ to $\mathcal{B}$. We say that the family has the *back and forth* property if for every $0 < k \leq m$ and every $f \in I_k$:

(forth) $\forall a \in A$, $a \notin dom(f)$, $\exists g \in I_{k-1}$, g extends f and $a \in dom(g)$;

(back) $\forall b \in B$, $b \notin range(f)$, $\exists h \in I_{k-1}$, g extends f and $b \in range(h)$. □

Note that the property back for f is the property forth for the inverse partial isomorphism f^{-1} from $\mathcal{B}$ to $\mathcal{A}$.

Now the following lemma points out the convenience of the notion of partial isomorphism. For every relational formula F, we will denote by $pr(F)$ the number of projection symbols π_i in F.

5.4. Lemma. *Let $\mathcal{A}$, $\mathcal{B}$ be structures of DB_σ and F be a formula of arity s in the signature σ. Assume that $pr(F) \leq m$, for some $m \in \mathbb{N}$, and there exists a family $I_m, I_{m-1}, \ldots, I_0$ of partial isomorphisms from $\mathcal{A}$ to $\mathcal{B}$ with the back and forth property. Then, for every $f \in I_m$ and $\vec{a} \in A^s$, $\vec{b} \in B^s$ such that $f : \vec{a} \mapsto \vec{b}$, i.e. $f(a_i) = b_i$ $\forall i = 1, \ldots, s$, it follows*

$$F^{\mathcal{A}} \vec{a} \quad \textit{if and only if} \quad F^{\mathcal{B}} \vec{b} \tag{1}$$

Proof. By reduction to absurdum, assume that the thesis is false. Then, there is a suitable instance of (1) which is false. Among such counterexamples take those for which m is minimum and in these choose one in which the number of all operations in F is minimum; we denote $\sharp F$ such number. Thus, since (1) is falsified by F, we may assume that

$$F^{\mathcal{A}}\vec{a} \quad \text{is true and} \quad F^{\mathcal{B}}\vec{b} \quad \text{is false} \quad . \tag{2}$$

In the symmetrical case the proof is analogous by changing $\mathcal{A}$ for $\mathcal{B}$ and the isomorphism f for f^{-1}. Now we distinguish several cases and we find an absurdum.

Case 1: $\sharp F = 0$. Then F is a some basic relation symbol, say P. Since f is a partial isomorphism and $f : \vec{a} \mapsto \vec{b}$ we have

$$P^{\mathcal{A}}\vec{a} \quad \text{if and only if} \quad P^{\mathcal{B}}\vec{b}.$$

This contradicts (2).

Case 2: $F = F_1 * F_2$, where $*$ is among $\cup$, $\setminus$, $\times$. Then we have that $\sharp F_1 < \sharp F$ and $\sharp F_2 < \sharp F$. Hence (1) is true for F_1 and F_2 by the choice of F. In other words

$$F_1^{\mathcal{A}}\vec{a} \quad \text{if and only if} \quad F_1^{\mathcal{B}}\vec{b}, \quad F_2^{\mathcal{A}}\vec{a} \quad \text{if and only if} \quad F_2^{\mathcal{B}}\vec{b}. \tag{3}$$

So, by the Definition 4.4, for the interpretation of $\cup$, $\setminus$, $\times$, the statements in (3) imply that (1) is true for F. This again contradicts (2).

Case 3: F is $\zeta^i_j G$ for some G. Since $\sharp G < \sharp F$ we have that (1) is true for G. This contradicts (2) since $a_i = a_j$ if and only if $b_i = b_j$; in fact, f is a partial isomorphism and $f : \vec{a} \mapsto \vec{b}$.

Case 4: F is $\pi_k G$ for some G. We have that the arity of G is $s > 1$ and $1 \le i \le s$; moreover, $pr(G) \le m-1$. By (2) there is $c \in A$ such that

$$G^{\mathcal{A}} a_1 \ldots a_{i-1} c\, a_{i+1} \ldots a_s. \tag{4}$$

Now, by the extension property there exists $g \in I_{m-1}$ extending f and such that $c \in Dom(g)$. Hence

$$g : a_1 \ldots a_{i-1} c\, a_{i+1} \ldots a_s \mapsto b_1 \ldots b_{i-1} d\, b_{i+1} \ldots b_s$$

and by (4) and the choice of F we have

$$G^{\mathcal{B}} b_1, \ldots b_{i-1} d\, b_{i+1} \ldots b_s$$

since $pr(G) \le m-1$. This implies that $F^{\mathcal{B}}\vec{b}$ is true which again contradicts (2).

Thus, in every case the negation of the thesis takes to contradiction. □

Now, for every $n \in \mathbb{N}$, $n > 0$ let G_n be the cycle graph of length n. In other words $G_n = (A_n, E_n)$ where $A_n = \{0, 1, \ldots, n-1\}$ and

$$E_n = \{(i, i+1),\ (i+1, i) : 0 < i < n-1\} \cup \{n-1, 0),\ (0, n-1)\}.$$

We denote by $G_k \uplus G_h$ the graph which is the disjoint union of the cycle G_k and the cycle G_h. Then, we have the following

5.5. Lemma. *For every $m \in \mathbb{N}$ and all integers $n,k,h > 2^m$ there exists a family $I_m \subseteq I_{m-1} \subseteq \ldots \subseteq I_1 \subseteq I_0$ of partial isomorphisms from G_{2n} to $G_k \uplus G_h$* with the back and forth property.

Proof. Let $A = \{a_0, a_1, \ldots, a_{2n-1}\}$ be the set of vertexes of G_{2n} and $B = \{b_0, b_1, \ldots, b_{k-1}\}$, $C = \{c_0, c_1, \ldots, c_{h-1}\}$ be the vertexes of G_k and G_h respectively, where A, B and C are pairwise disjoints sets. Let W be a set contained either in A or in $B \cup C$, we call *bridge-walk for* W any walk $x_0, x_1, \ldots, x_s$ where $s > 1$, $x_0 \in W$, $x_s \in W$ and $x_i \notin W$ for every $0 < i < s$.

Now, for every $0 \leq k \leq m$ we define a set I_k of partial isomorphisms from G_{2n} to $G_k \uplus G_h$ as follows:

$f \in I_k$ if and only if f is a partial isomorphism with non-empty domain and

a) Every bridge-walk $x_0, x_1, \ldots, x_s$ for $dom(f)$ has length $s \geq 2^k$.

b) Every bridge-walk $y_0, y_1, \ldots, y_s$ for $range(f)$ has length $s \geq 2^k$.

From the definition of the family $\{I_k\}_k$ it follows that $I_m \subseteq I_{m-1} \subseteq \ldots \subseteq I_1 \subseteq I_0$; moreover, every set I_k is non-empty. In fact, $f_0 \in I_m$ where

$$f_0 : a_0 \mapsto b_0 \qquad f_0 : a_n \mapsto c_0 \tag{1}$$

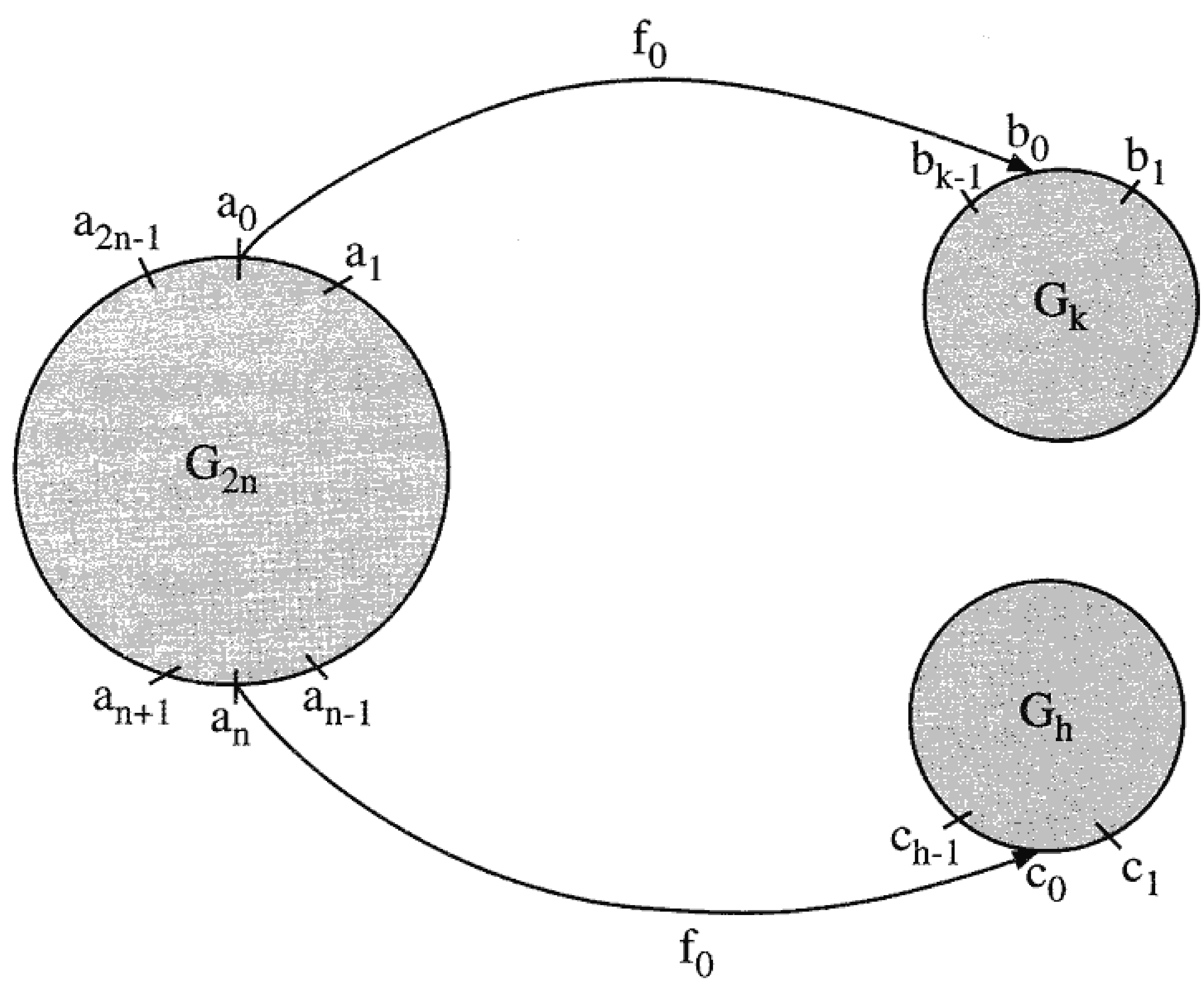

Figure 3

This follows from the fact that the only possible bridge-walk for $dom(f)$ is $a_0, a_1, \ldots, a_{n-1}, a_n, \ldots, a_{2n-1}, a_0$ and the only bridges-walks for $range(f)$ are $b_0, b_1, \ldots, b_{h-1}, b_0$ and $c_0, c_1, \ldots, c_{k-1}, c_0$; all of them have length greater than 2^m.

Now, we prove that the family $\{I_k\}_k$ has the back and forth extension property. Let $0 < k \leq m$, $f \in I_k$ and $x \in A$, such that $x \notin dom(f)$, be fixed. Take $a \in dom(f)$ with minimum distance from x; we call *distance* between two vertexes the minimum length of walks which join them. Then there exists a bridge-walk $x_0, x_1, \ldots, x_q, \ldots, x_r$ in the cycle G_{2n}, where $x_0 = a$, $x_q = x$, $r > 1$, $x_r \in dom(f)$ and $x_i \notin dom(f)$ for every $0 < i < r$. So, recalling that $f \in I_k$ and the condition a) in the definition of I_k, we have $r \geq 2^k$.

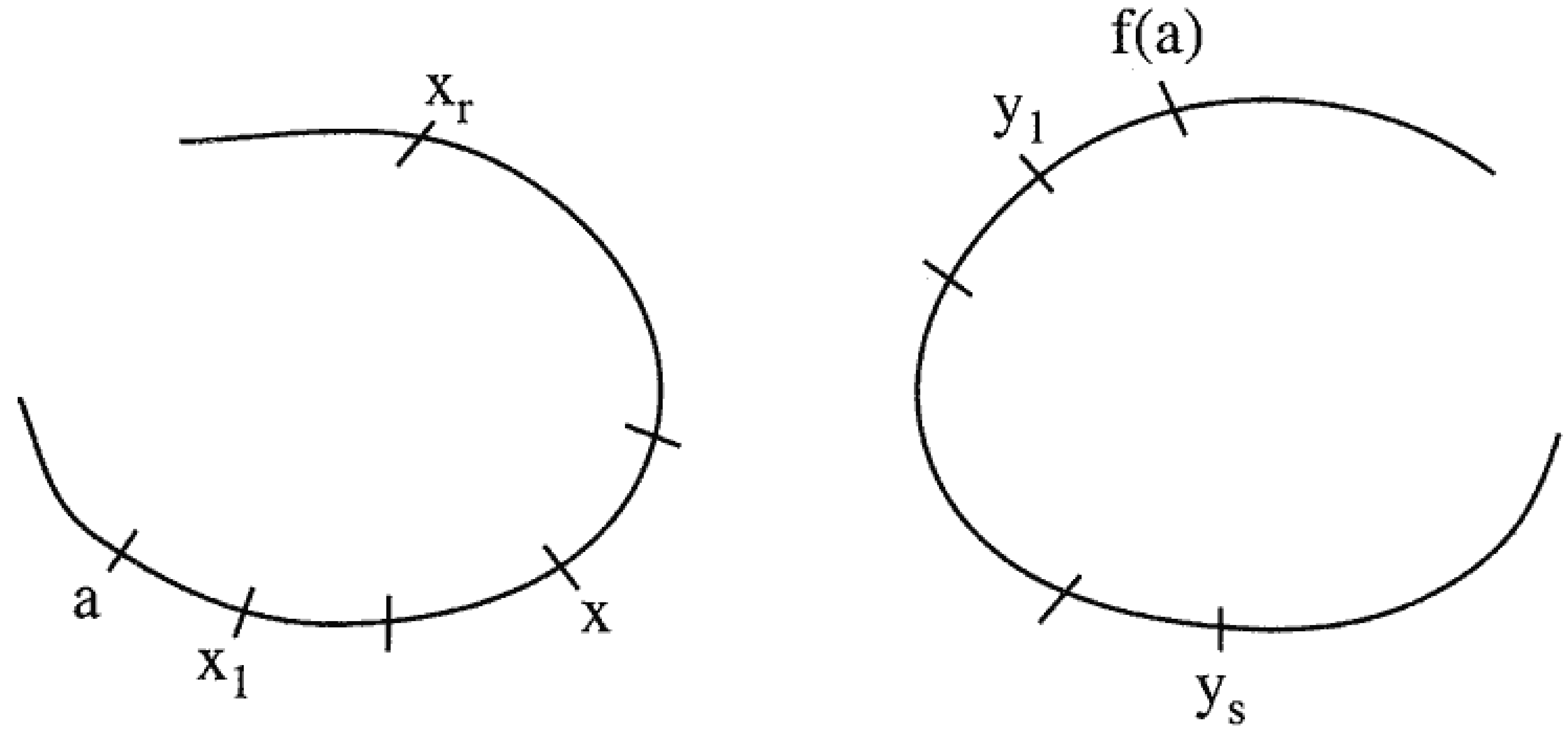

Figure 4

Since, by definition, partial isomorphisms preserve adjacencies, at least one between the two neighbours of $f(a)$, call it y_1, is not in $range(f)$, otherwise also the neighbour x_1 of a would be in $dom(f)$. Then there exists a bridge-walk $f(a) = y_0, y_1, \ldots, y_s = d$ where $d \in range(f)$; possibly $d = f(a)$ since we are in a cycle. Moreover, by condition b) in the definition of I_k, we have $s \geq 2^k$. Now, we distinguish two cases.

Case 1: $q < 2^{k-1}$. Then $q < s$. So, let g be extension of f where

$$dom(g) = dom(f) \cup \{x_1, x_2, \ldots, x_q\}$$

and $g(x_i) = y_i$ for $1 \leq i \leq q$. The mapping g is a partial isomorphism since it preserves adjacencies. Moreover, $x_q, \ldots, x_r$ is the only bridge-walk for $dom(g)$ and not for $dom(f)$, and $y_q, \ldots, y_s$ is the only bridge-walk for $range(g)$ and not for $range(f)$. Then, $g \in I_{k-1}$ since $s - q$ and $r - q$ are both $\geq 2^{k-1}$.

Case 2: $q \geq 2^{k-1}$. Then, by the choice of x, also $r - q \geq 2^{k-1}$. So, let g be an extension of f where

$$dom(g) = dom(f) \cup \{x\} \quad \text{and} \quad g(x) = y_{2^{k-1}}.$$

Observe that the unique bridge-walks for $dom(g)$, which were not already for $dom(f)$, are $x_0, x_1, \ldots, x_q$ and $x_q, \ldots, x_r$; similarly the unique bridge-walks for $range(g)$ and not $range(f)$ are $y_1, y_2, \ldots, y_{2^{k-1}}$ and $y_{2^k+1}, \ldots, y_s$. Therefore, also in this case, $g \in I_{k-1}$. This concludes the proof of the forth property for the family $\{I_k\}_{0 \leq k \leq m}$; we omit the proof of the back property which is quite analogous. □

Finally, we go back to the

Proof of Proposition 5.2. Let us assume by absurdum that the global relation determined by the transitive closure, on the class of finite structures with a binary relation, be definable by some formula F. Let $m = pr(F)$ be the number of projection in F. Then, fix $n, k, h > 2^m$ and, by the Lemma 5.5, a family $I_m \subseteq I_{m-1} \subseteq \cdots \subseteq I_0$ of partial isomorphism between G_{2n} and

$G_k \uplus G_h$. Moreover, let $f_0 \in I_m$ be defined as in (1) of Lemma 5.5, i.e.

$$f_0 : a_0 \mapsto b_0, \qquad f_0 : a_n \mapsto c_0;$$

see figure 3. Then, by Lemma 5.4, we would have

$$F^{G_{2n}} a_0 a_n \quad \text{if and only if} \quad F^{(G_k \uplus G_h)} b_0 c_0.$$

But, this is a contradiction since a_0, a_n are joined by a walk and b_0, c_0 are not connected. So, F cannot define the transitive closure in both G_{2n} and $G_k \uplus G_h$. □

6 Relational formulas and fixed point.

In the previous section we proved that the global relation of transitive closure cannot be defined by a relational formula. However, this global relation can be obtained as fixed point as follows.

6.1 Example. Let A be a set, R a binary relation on A. Consider the operator

$$T : \mathcal{P}(A^2) \to \mathcal{P}(A^2)$$

defined by $T(S) = R \cup (S \circ R)$. Then, it is easy to see that T is continuous. Therefore, by Lemma 2.5, there exists the minimum fixed point for T. This is the minimum transitive relation containing R. In fact, by iterating T, for $S = \emptyset$ we get

$$T_0 = \emptyset, \quad T_1 = T(\emptyset) = R, \quad T_2 = T(R) = R \cup (R \circ R) = R \cup R^2, \ldots,$$

$$, \ldots, T_n = \bigcup_{i=0}^{n} \underbrace{(R \circ \cdots \circ R)}_{i-\text{times}}.$$

Then the minimum fixed point for T is

$$T_\omega = \bigcup_{i=0}^{\infty} \underbrace{(R \circ \cdots \circ R)}_{i-\text{times}}.$$

□

The previous example shows that the transitive closure can be obtained as a fixed point of an operator defined by a relational algebra formula. Now, we add the fixed point operation to extend the class of relational formulas in order to define new global relations on the database DB_σ. To this end we need to enrich the language L_σ by countably many symbols

$$Y_1^1, Y_2^1, \ldots, Y_1^2, Y_2^2, \ldots, Y_m^n, \ldots \tag{1}$$

which we intend to interpret on relations; Y_m^n will be interpreted on a relation of arity n. To simplify notation we will use symbols $X, Y, Z, \ldots, X_1, \ldots$ in place of symbols in (1). Such symbols are said also *relational variables*; on the contrary the symbols $P_1, P_2, \ldots, P_k$ of L_σ for the basic relation, are also said *relational constants*. Let $X_1, X_2, \ldots, X_r$ be relational variable symbols of given arity, we call *formula in the language* $L_\sigma \cup \{X_1, X_2, \ldots, X_r\}$ any relational formula built as in the Definition 4.2 on the relation symbols $P_1, P_2, \ldots, P_k, X_1, X_2, \ldots, X_r$.

6.2 Definition. We call *triangular system of relational formulas* in a given signature σ any system

$$\begin{cases} X_1 & = F_1 \\ X_2 & = F_2 \\ \cdots & = \cdots \\ X_r & = F_r \end{cases} \tag{1}$$

where, for every $i = 1, 2, \ldots, r$, each F_i is a formula in the language $L_\sigma \cup \{X_1, X_2, \ldots, X_i\}$, and the difference operator "$\setminus$" is not in front of the relational variable X_i.

Given a structure $\mathcal{A} = (A, P_1^{\mathcal{A}}, P_2^{\mathcal{A}}, \ldots, P_k^{\mathcal{A}})$ in DB_σ, we say that the system (1) has *solution* in $\mathcal{A}$ if there exist relations $S_1, S_2, \ldots, S_r$ where S_i has the same arity as X_i, and $S_i = F_i^{\mathcal{A}^*}$, for every $i = 1, 2, \ldots, r$. The value $F_i^{\mathcal{A}^*}$ of the formula F_i is computed in the structure

$$\mathcal{A}^* = (A, P_1^{\mathcal{A}}, P_2^{\mathcal{A}}, \ldots, P_k^{\mathcal{A}}, S_1, S_2, \ldots, S_{i-1}).$$

A solution $S_1, S_2, \ldots, S_r$ will be said *minimal* if for any other solution $S_1', S_2', \ldots, S_r'$ it follows $S_i \subseteq S_i'$, for every $i = 1, 2, \ldots, r$. □

The following proposition shows that we may extend the class of global relations definable by a relational formula to the larger class of global relations definable by triangular systems of relational formulas.

6.3 Proposition. *A triangular system of relational formulas in a signature σ has a unique minimal solution in every structure $\mathcal{A}$ of DB_σ.*

Proof. Let us consider a triangular system, as in Definition 6.2, in a signature σ and let $\mathcal{A} = (A, P_1^{\mathcal{A}}, P_2^{\mathcal{A}}, \ldots, P_k^{\mathcal{A}})$ be in DB_σ. The result will be proved by induction on the number of equations in the system. In other words, given a minimal solution $S_1, S_2, \ldots, S_{i-1}$ of the first $i-1$ equations, we build S_i such that $S_1, S_2, \ldots, S_{i-1}, S_i$ is a minimal solution of the first i equations, for every $1 \leq i \leq r$.

Assume first $i = 1$. The formula F_1 may contain at most X_1 among the relational variables $\{X_0, X_1, \ldots, X_r\}$; let s be the arity of X_1. Then the formula F_1, thought as function of X_1, determines a map

$$F_1 : \mathcal{P}(A^s) \to \mathcal{P}(A^s),$$

which is a monotone operator since the difference operator is not in front of X_1. Observe that all the operations,

$$(X,Y) \to X \cap Y,\ (X,Y) \to X \cup Y,\ (X,Y) \to X \times Y,\ X \to \pi_j X,\ X \to \zeta_j^i X,$$

but the difference, are monotone in each component; in fact, they determine continuous operators. Then, by the Theorem 2.4, there exists a unique minimal fixed point, say it S_1, which solves the equation $X_1 = F_1$ in $\mathcal{A}$. Now, let $S_1, S_2, \dots, S_{i-1}$ be a solution of the first $i-1$ equations for $i > 1$. Then we may consider the structure $\mathcal{A}' = (A, P_1^{\mathcal{A}}, P_2^{\mathcal{A}}, \dots, P_k^{\mathcal{A}} S_0, S_1, \dots, S_{i-1})$, and we may think the relational variables $X_0, X_1, \dots, X_{i-1}$ in the system as relational constants with interpretation in $\mathcal{A}'$ given by $S_1, \dots, S_{i-1}$, respectively. Thus, as before we may compute S_i in $\mathcal{A}'$ such that $S_1, S_2, \dots, S_{i-1}, S_i$ is a minimal solution of the first i equations. □

6.4 Definition. A global relation Q, defined on the class DB_σ, is said to be *definable in the relational algebra with minimum fixed point* if there exists a triangular system of relational formulas as in (1) such that, for every $\mathcal{A} \in DB_\sigma$ it is $Q_{\mathcal{A}} = X_r^{\mathcal{A}}$ where $X_1^{\mathcal{A}}, X_2^{\mathcal{A}}, \dots, X_r^{\mathcal{A}}$ is a minimal solution of the system in $\mathcal{A}$. □

6.5 Remarks.

1) The class of global relations definable by triangular systems extends the class of queries definable by relational formulas. In fact, the global relation defined by the formula F can also be defined by the system with the single equation $X = F$, where X is a relational variable of the same arity as F.

2) The transitive closure with respect to a binary symbol P of L_σ is definable on DB_σ by the single equation
$$X = P \cup (P \circ X)$$
or more precisely
$$X = P \cup \pi_2 \pi_3 \zeta_3^2 (P \times X)$$
which is written with only symbols in $L_\sigma \cup \{X\}$; remember Remark 4) in 3.3. □

We close this section by discussing how the computation of a minimal solution for a triangular system where the difference does not appear can be carried out in logic program. We give a brief outline assuming that the reader be acquainted with the main notions.

We recall that a *Horn logic program* (see [Ll87]) $\mathcal{HP}$ is a finite list of *clauses* of the form
$$A \leftarrow B_1, B_2, \dots, B_m \tag{1}$$
where $A, B_1, B_2, \dots, B_m$ are atomic formulas. Each atomic formula is interpreted as a relation and each clause is interpreted as a sentence where the individual variables are universally quantified. Since a *atomic formula* is of the form $P(t_1, t_2, \dots, t_h)$ where P is a relation symbol, the clauses in (1) assert that if certain relations among certain elements are true then also some other relations are true. When $m = 0$ the clause is an assertion without condition of the form $P(t_1, t_2, \dots, t_h)$. In this case the clause is called *unitary*; namely, the unitary clauses codify relations.

A *goal* in a logic program $\mathcal{HP}$ is a clause of the form
$$\leftarrow G_1, G_2, \dots, G_m. \tag{2}$$
Its role is to establish if the sentence $\exists \vec{x} (G_1 \wedge \cdots \wedge G_m)$ is or not logic consequence of $\mathcal{HP}$. Here, we are interested in case $m = 1$ and the goal has no variables. More precisely we consider a structure $\mathcal{A} = (A, P_1^{\mathcal{A}}, P_2^{\mathcal{A}}, \dots, P_k^{\mathcal{A}})$ in a signature σ and we enrich σ to obtain a

signature $\sigma_{\mathcal{A}}$ with a constant symbol for every element of $\mathcal{A}$; for every $a \in \mathcal{A}$ we denote with the same symbol the element a and its name which is a constant in $\sigma_{\mathcal{A}}$. For every triangular system and every $\mathcal{A}$ we are going to build a finite set of clauses, a program, in $\sigma_{\mathcal{A}}$ of the form $\mathcal{HP} \cup D(\mathcal{A})$ where $\mathcal{HP}$ is a suitable set of clauses in the signature $\sigma_{\mathcal{A}}$ and $D(\mathcal{A})$ is the set of all the unitary clauses without variables which are true in $\mathcal{A}$, i.e. $D(\mathcal{A})$ is the set

$$P_i(\vec{a}) \leftarrow \quad \text{for} \quad i = 1, 2, \ldots, n \qquad \text{and} \quad \vec{a} \quad \text{such that} \quad P_i^{\mathcal{A}}\vec{a} \quad \text{holds in} \quad \mathcal{A}.$$

The clauses of $D(\mathcal{A})$ codify the database $\mathcal{A}$ and the clauses in $\mathcal{HP}$ are emploied to deduce new relations. If R is a relation symbol in $\mathcal{HP}$, not necessarily of the signature of $\mathcal{A}$, we may ask if $R(\vec{a})$ is a logic consequence of $\mathcal{HP} \cup D(\mathcal{A})$. This is true if and only if $R(\vec{a})$ is true in the minimal Herbrand model of $\mathcal{HP} \cup D(\mathcal{A})$. But this fact, by well-known theory of logic programming, can be tested by building the resolution tree for the goal $\leftarrow R(\vec{a})$ with respect to the program $\mathcal{HP} \cup D(\mathcal{A})$. Moreover, this can be done mechanically in one of the usual computer implementation of logic programming (see [Ll87]).

The next Proposition asserts how logic programs can be used to solve triangular systems of relational formulas.

6.6 Proposition. *Let $\mathcal{S}$ be a triangular system of relational formulas in the language $L_\sigma \cup \{X_1, X_2, \ldots, X_r\}$ and the difference symbol "$\setminus$" is not present in $\mathcal{S}$. Then, it is possible to build a logic program $\mathcal{HP}$ in the first-order language with relation symbols $\{P_1, P_2, \ldots, P_k, X_1, X_2, \ldots, X_r\}$ such that, for every structure $\mathcal{A}$ of signature σ, the program $\mathcal{HP} \cup D(\mathcal{A})$ has the minimal model*

$$\mathcal{A}^* = (A, P_1^{\mathcal{A}}, P_2^{\mathcal{A}}, \ldots, P_k^{\mathcal{A}}, X_1^{\mathcal{A}}, X_2^{\mathcal{A}}, \ldots, X_r^{\mathcal{A}}) \tag{3}$$

where $X_1^{\mathcal{A}}, X_2^{\mathcal{A}}, \ldots, X_r^{\mathcal{A}}$ is the minimal solution of $\mathcal{S}$ in $\mathcal{A}$. Hence, as consequence we have that, for testing if $X_i^{\mathcal{A}}(\vec{a})$ is true, it is sufficient to run the goal $\leftarrow X_i(\vec{a})$ in the logic program $\mathcal{HP} \cup D(\mathcal{A})$.

Proof. Let $\mathcal{S}$ be a triangular system as defined in (1) of 6.2 where the difference symbol "$\setminus$" does not appear. We say that an equation $X = F$ is *simple* if the formula F is in one of the following forms:

a) P, a basic relation symbol;

b) $Y \cap Z$, where $ar(Y) = ar(Z) = n$;

c) $Y \cup Z$, where $ar(Y) = ar(Z) = n$;

d) $Y \times Z$, where $ar(Y) = m$, $ar(Z) = n$;

e) $\pi_i Y$, where $ar(Y) = n \geq 1$ and $1 \leq i \leq n$;

f) $\zeta_j^i Y$, where $ar(Y) = n$ and $1 \leq i < j \leq n$;

with the convention that X, Y, Z range in the relational variables $X_1, X_2, \ldots, X_r$ or in the relational constants $P_1, P_2, \ldots, P_k$. Now, the procedure to build $\mathcal{HP}$ acts in two steps:

1) transforms $\mathcal{S}$ in a triangular system $\mathcal{S}'$ where every equation is simple;

2) transforms $\mathcal{S}'$ in the wanted program $\mathcal{HP}$.

A triangular system $\mathcal{S}'$ which contains only simple equations can be transformed in the program $\mathcal{HP}$ by replacing each equation with suitable clauses. Every relation symbol U present in $\mathcal{S}'$ will be used also as relation symbol of the same arity in the program $\mathcal{HP}$. Moreover $u_1, u_2, \ldots, u_n, \ldots, v_1, v_2, \ldots, v_m \ldots$ will denote distinct first order variables. Then, the procedure replaces each equation $X = F$ where F is in cases a)–f), respectively, as follows:

a) with the clause $X(u_1, u_2, \ldots, u_n) \leftarrow P(u_1, u_2, \ldots, u_n)$;

b) with the clause $X(u_1, u_2, \ldots, u_n) \leftarrow Y(u_1, u_2, \ldots, u_n), Z(u_1, u_2, \ldots, u_n)$;

c) with the two clauses

$$X(u_1, u_2, \ldots, u_n) \leftarrow Y(u_1, u_2, \ldots, u_n) \quad \text{and} \quad X(u_1, u_2, \ldots, u_n) \leftarrow Z(u_1, u_2, \ldots, u_n);$$

d) with the clause

$$X(u_1, u_2, \ldots, u_n, v_1, v_2, \ldots, v_m) \leftarrow Y(u_1, u_2, \ldots, u_n), Z(v_1, v_2, \ldots, v_m);$$

e) with the clause $X(u_1, u_2, \ldots, u_{n-1}) \leftarrow Y(u_1, u_2, \ldots, u_{i-1}, v, u_i, \ldots, u_{n-1})$;

f) with the clause $X(u_1, u_2, \ldots, u_n) \leftarrow Y(u_1, u_2, \ldots, u_i, \ldots, u_{j-1}, u_i, u_{j+1}, \ldots, u_n)$.

Now, it is easy to prove that the semantics of Horn clauses in the minimal model (see [Ll87]) implies that the structure $\mathcal{A}^*$, defined as in (3), is the minimal model of $\mathcal{HP} \cup D(\mathcal{A})$ for every $\mathcal{A} \in DB_\sigma$.

Finally, the procedure for transforming every triangular system $\mathcal{S}$ into a system $\mathcal{S}'$ of simple equations operates inductively on each equation $X = F$ with respect to the number of operations which appear in F. If F is not already simple then $F = F_1 * F_2$, where " $*$ " is a suitable operation symbol and F_1, F_2 are not both relational symbols. Then, the procedure replaces the equation $X = F$ with

$$X = Y_1 * Y_2, \quad Y_1 = F_1, \quad Y_2 = F_2;$$

where Y_1, Y_2 are new relational variables which are not present in the formulas of the system and have convenient arity. Repeating the routine as long as possible we get a simple system. □

To clarify the procedure we consider the following

6.7. Example. Let us consider the equation

$$X = P \cup \pi_2 \pi_3 \zeta_3^2 (P \times X) \tag{1}$$

which defines the transitive closure of P, already met in 6.1. Then the transformation produces

$$\begin{aligned} X &= P \\ X &= \pi_2 Y \\ Y &= \pi_3 Z \\ Z &= \zeta_3^2 T \\ T &= P \times X. \end{aligned}$$

□

The procedure for transforming triangular systems into logic programs works well also if the difference operation appears in the system with the condition that the second member of the difference be one of the basic relation symbol among $P_1, P_2, \ldots, P_k$. In this case the resulting program is a DATALOG program and no more Horn (see [EF95]).

References

[1] Birkhoff, G., *Lattice Theory*, 3^{rd} edition, Amer. Math. Soc., New York (1979).

[2] Codd, E.F., *A Relational Model for Large Shared Databases*, Commun. of the ACM 13, 6 (1970).

[3] Codd, E.F., *Relational Completeness of Database Sublanguage*, Data Base System (Rustin ed.) Prentice Hall (1972).

[4] Date, C.J. *An Introduction to Database Systems*, 3^{rd} edition, Addison-Wesley Publishing Company (1975).

[5] Ebbinghaus, H.D., Flum, J., *Finite Model Theory*, Springer–Verlag Berlin Heidelberg (1995).

[6] Gallaire, H., Minker, J., Nicolas, J.M., *Logic and Data Bases: a Deductive Approach*, Assoc. Comput. Mach. Comput. Surveys, 16 (1984), 153-185.

[7] Lloyd, J.W., *Foundations of Logic Programming*, 2^{nd} edition, Springer–Verlag Berlin, (1987).

[8] Maier, D., *The Theory of Relational Databases*, Computer Science Press (1983).

[9] Knaster, B., *Un Théoreme sur les Functiones d'Ensemble*, Ann. Soc. Poln. Math. 6 (1928), p.133-134.

[10] Tarski, A., *A Lattice Theoretical Fixpoint Theorem and its Applications*, Pacific. Jour. of Math. 5 (1955), 283-309.

[11] Ullman, J.D., *Database Systems*, 2^{nd} edition, Computer Science Press (1982).

Stefano Leonesi
Dipartimento di Matematica
Università di Siena
via Del Capitano
53100 Siena, ITALY
Dipartimento di Matematica e Fisica

Università di Camerino
via Madonna delle Carceri
62032 Camerino (MC), Italy
e-mail: leonesi@campus.unicam.it

Sauro Tulipani
Dipartimento di Matematica e Informatica
Università di Perugia
via Vanvitelli, 1
06100 Perugia, Italy
e-mail: tulipani@dipmat.unipg.it