Generalized Implicit Definitions on Finite Structures

Stéphane Grumbach*1 and Zoé Lacroix*2 and Steven Lindell**3

1 University of Toronto and INRIA
2 Université de Paris-Sud and INRIA
3 Haverford College

Abstract. We propose a natural generalization of the concept of implicit definitions over finite structures, allowing non-determinism at an intermediate level of a (deterministic) definition. These generalized implicit definitions offer more expressive power than classical implicit definitions. Moreover, their expressive power can be characterized over unordered finite structures in terms of the complexity class NP ∩ co-NP. Finally, we investigate a subclass of these where the non-determinism is restricted to the choice of a unique relation with respect to an implicit linear order, and prove that it captures UP ∩ co-UP also over the class of all finite structures. These results shed some light on the expressive power of non-deterministic primitives.

1 Introduction

Let \( K \) be a class of structures over a vocabulary \( \sigma \), and \( R \) a relation symbol not in \( \sigma \). A first-order sentence \( \varphi(R) \) over \( \sigma \cup \{ R \} \) implicitly defines a relation on \( I \in K \), if there is a unique relation \( R' \), such that \( I \models \varphi(R') \). A \( k \)-ary relation \( R \) is explicitly definable on \( I \in K \) if there is a first-order formula \( \varphi(\bar{x}) \) over \( \sigma \), with \( k \) free variables such that \( R = \{ \bar{a} \in I^k : I \models \varphi(\bar{a}) \} \). An explicit definition expresses a property of the tuples in the relation, whereas an implicit definition expresses a property of the relation itself.

Beth’s definability theorem [Bet53] says that if a relation is implicitly definable over the class of all \( \sigma \)-structures, then it is explicitly definable. More generally, it is well known that this result fails when restricted to specific classes of models. In particular, this failure was illustrated by Gurevich for the class of finite structures [Gur84]. Implicit definitions over finite structures were then further investigated by Kolaitis, where their expressive power as definitions of

---

* I.N.R.I.A. Rocquencourt BP 105, 78153 Le Chesnay, France – E-mail: stephane.grumbach@inria.fr – Work supported in part by an EC-NSF grant DEUS EX MACHINA, and an NSERC fellowship in Canada.
** L.R.I., Université de Paris Sud, 91405 Orsay, France – E-mail: zoe@lri.fr – Work done in the Verso Project at INRIA.
*** Department of Computer Science, Haverford College, Haverford, PA 19041-1392 – E-mail: slindell@haverford.edu – Work partially supported by NSF grant CCR-9403447 and the John C. Whitehead faculty research fund at Haverford College.
queries was studied [Kol90]. A sentence \( \varphi(R) \) over \( \sigma \cup \{ R \} \) implicitly defines a query on \( K \), if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definable in IMP on \( K \) if for every structure \( I \) in \( K \), there is a unique relation \( R^I \), such that \( I \models \varphi(R^I) \).

More generally, Kolaitis introduced the class IMP of queries mapping finite structures over \( \sigma \) to some implicitly definable \( k \)-ary relation \( R \). A \( k \)-ary query \( Q \) is definit...
In this paper, we prove the general (non-boolean) case of the theorem in [Lin87].

**Theorem 4.1** GIMP = NP ∩ co-NP, on the class of all finite structures.

We then go on to investigate in more detail the precise expressive power resulting from the non-determinism allowed in the intermediate relation. In GIMP, it is possible to "guess" a linear order on the domain of a finite structure, and it is well known that the order relation plays a fundamental role in the characterization of complexity classes below NP by logical formalisms [Var82, Imm86]. We consider a natural restriction, LIMP, of GIMP, where the non-determinism is restricted to the choice of some order relation on the finite domain, and prove our main theorem.

**Theorem 5.2** LIMP = UP ∩ co-UP, on the class of all finite structures,

where UP denotes the class of queries computed by unambiguous Turing machines, i.e. NP machines with at most one accepting computation on every input [Val76].

This last theorem generalizes a similar result [Kol90] IMP = UP ∩ co-UP which applies only on the more restrictive class of all ordered structures. So the relative expressive power of various non-deterministic primitives is linked with difficult open problems in complexity theory.

## 2 Preliminaries

A *vocabulary* σ is a finite sequence \( \{P_1, \ldots, P_s\} \) of relation symbols of fixed arities. A σ-structure \( I \) is a set I, called the *universe*, along with a mapping associating for all \( i \in \{1, \ldots, s\} \) a relation \( P_i^I \) over I with the same arity as \( P_i \). A σ-structure is finite if its universe I is a finite set. \( \mathcal{K} \) denotes a class of σ-structures, \( \mathcal{F} \) the class of all finite σ-structures, and \( \mathcal{O} \) the class of finite ordered σ-structures. \( \mathcal{L} \) is a logic language.

For \( k \) a non-negative integer, let a \( k \)-ary query (often called a *global relation*) \( Q \) on \( \mathcal{K} \) be a mapping that associates to each σ-structure \( I \in \mathcal{K} \) a \( k \)-ary relation \( Q(I) \) on the universe I such that for all σ-structures \( I \) and \( J \) and isomorphisms \( f \) with \( f(I) = J \), tuple \( \langle a_1, \ldots, a_k \rangle \in Q(I) \) iff \( \langle f(a_1), \ldots, f(a_k) \rangle \in Q(J) \). If \( k = 0 \), then \( Q \) is called a *boolean query*, because \( Q^I \subseteq I^k = \{ \langle \rangle \} \) means \( Q \) is either \( 1 = \{ \langle \rangle \} \) or \( 0 = \emptyset \). So \( Q \) determines the characteristic function of a subclass of \( \mathcal{K} \) given by \( \{ I \in \mathcal{K} : Q(I) = 1 \} \). The complement of a \( k \)-ary query \( Q \) is \( \neg Q \) such that for each σ-structure \( I \), \( \neg Q(I) = I^k - Q(I) \).

**Definition 2.1** A \( k \)-ary query \( Q (k \geq 0) \) is *explicitly defined* in \( \mathcal{L} \) on \( \mathcal{K} \) if there is a formula of \( \mathcal{L} \) \( \varphi(x_1, \ldots, x_k) \) with \( k \) free variables such that for each \( I \in \mathcal{K} \), \( Q(I) = \{ \bar{a} \in I^k : I \models \varphi(\bar{a}) \} \).
Generally, when a query is said to be definable in the logic $L$, it means explicitly definable in $L$. In the following, we often omit the term explicitly when referring to queries (explicitly) definable in: FO (first-order logic); FP (fixed-point logic); $\Sigma^1_1$ (existential second-order logic); and $U\Sigma^1_1$ (unique existential second-order logic).

**Definition 2.2** A $k$-ary query $Q$ ($k \geq 0$) is definable in $\Sigma^1_1$ if there exists a first-order formula $\varphi(S, x_1, \ldots, x_k)$ over $\sigma \cup \{ S \}$ with $k$ free variables such that for each $I \in \mathcal{K}$, $Q(I) = \{ \bar{a} \in I^k : I \models \exists S \varphi(S, a_1, \ldots, a_k) \}$.

A query $Q$ is definable in $\Pi^1_1$ if $\neg Q$ is definable in $\Sigma^1_1$.

A query $Q$ is definable in $\Delta^1_1$ if both $Q$ and $\neg Q$ are definable in $\Sigma^1_1$.

There is a strong correspondence between computational complexity classes and these logics. A logic $L$ is said to capture a complexity class $C$ if:

- for any formula $\varphi$ of $L$, the set $\text{MOD}(\varphi)$ of the models of $\varphi$ (the set of structures that satisfy $\varphi$) is recognizable in $C$,
- for all class $\mathcal{K}$ of sets recognizable in $C$ that is closed under isomorphisms, there exists a formula $\varphi$ in $L$ such that $\mathcal{K} = \text{MOD}(\varphi)$.

Fagin’s theorem [Fag74] shows that the logic $\Sigma^1_1$ captures the computational complexity class NP. Valiant considered a subclass of NP called UP, which denotes the class of queries computed by unambiguous Turing machines, i.e., NP machines with at most one accepting computation on every input [Val76]. The corresponding logic $U\Sigma^1_1$, a subclass of $\Sigma^1_1$, was introduced by Kolaitis such that for any tuple that satisfies the formula there exists a unique relational witness [Kol90].

**Definition 2.3** A $k$-ary query $Q$ ($k \geq 0$) is definable in $U\Sigma^1_1$ if there exists a first-order formula $\varphi(S, x_1, \ldots, x_k)$ over $\sigma \cup \{ S \}$ with $k$ free variables such that for each $I \in \mathcal{K}$, $Q(I) = \{ \bar{a} \in I^k : I \models \exists S \varphi(S, a_1, \ldots, a_k) \}$ and $I \models \forall \bar{x} \exists S \varphi(S, x_1, \ldots, x_k) \rightarrow \exists S \varphi(S, x_1, \ldots, x_k)$ (where $\exists S$ means “there is exactly one relation $S$”).

A query $Q$ is definable in $\Pi^1_1$ if $\neg Q$ is definable in $U\Sigma^1_1$.

A query $Q$ is definable in $\Delta^1_1$ if both $Q$ and $\neg Q$ are definable in $U\Sigma^1_1$.

**Remark:** The set $\Sigma^1_1$ is a recursively enumerable set of formulas. In contrast, $U\Sigma^1_1$ is a co-recursively enumerable set, because the condition $I \models \forall \bar{x} \exists S \varphi(S, \bar{x})$ is expressible in $\Pi^1_1 : I \models \forall \bar{x} \forall S \forall T (\varphi(S, \bar{x}) \land \varphi(T, \bar{x}) \rightarrow \forall \bar{y} (S \bar{y} = T \bar{y}))$. In other words, a sentence $\exists S \varphi(S, \bar{x}) \notin U\Sigma^1_1$ if $\exists \bar{x} \exists S \exists T \varphi(S, \bar{x}) \land \varphi(T, \bar{x}) \land \exists \bar{y} (S \bar{y} = \neg T \bar{y})$ is finitely satisfiable.

On the class $O$, UP is characterized by the logic $U\Sigma^1_1$. Notice that a built-in order is required to get the correspondence, because it is not possible to postulate a unique order in general.
3 Implicit Definability

An explicit definition of a global relation $Q$ expresses a property of the tuples in $Q$, whereas an implicit definition expresses a property of the entire global relation $Q$. To illustrate this we present the notion of an individual global relation being implicitly definable, to be followed by the natural generalization to a sequence of such relations (as described in [Kol90]).

An individual global relation $Q$ of arity $k$ is said to be *implicitly defined* on a class of $\sigma$-structures $K$ if there exists a first-order sentence $\varphi(S)$ over $\sigma \cup \{S\}$, where $S$ is a new relation symbol of arity $k$, such that for each $I$ in $K$, $Q(I)$ is the unique relation that satisfies $\varphi$ on $I$. The global implicit definition of $Q$ over $K$ is a $k$-ary query. Let us use IMP$_1$ to denote the class of individual implicitly definable queries.

Note that an explicitly defined (FO) query is also implicitly definable. Generally, the converse is not true. On the class of all $\sigma$-structures, Beth’'s theorem says that explicit and implicit definability coincide [Bet53]. However, this fails on $K = \mathcal{F}$ as illustrated by Gurevich [Gur84].

Consider the following example of an IMP$_1$ query.

**Example 3.1** Let $\sigma$ be a vocabulary containing only one binary relation symbol $\prec$. The unary query $Q$ consisting of the set of even elements with respect to $\prec$, if $\prec$ is a linear order on the structure, and of the empty set, otherwise, is implicitly defined by the first-order sentence $\varphi(S)$ where $S$ is a new unary relation symbol.

$$\varphi(S) \equiv [\theta(\prec) \rightarrow \phi(S)] \land [\neg \theta(\prec) \rightarrow \forall x \neg S x],$$

where $\theta(\prec)$ asserts that $\prec$ is a linear order and

$$\phi(S) \equiv \forall x (\forall y \neg y < x ) \rightarrow \neg S x \land \forall x \forall y [\neg S x \rightarrow (\neg (\exists z, z < x < y \rightarrow S y)]$$

expresses that $S$ is the set of even elements with respect to the linear order $\prec$. More succinctly, if the domain is $\{1, \ldots, n\}$, $\phi(S)$ says $\neg S(1) \land \forall x (\neg S(x) \rightarrow S(x + 1))$. $Q$ is not explicit since there is no first-order formula that defines the set of even elements on finite total orders (use 0/1 laws [Fag76], or Ehrenfeucht-Fraïssé games [Fra54, Ehr61]).

It is easy to see that IMP$_1$ is closed under complementation: If $\varphi(S)$ is an implicit definition of a query $Q$, then $\varphi(T)$ obtained from $\varphi(S)$ by replacing each occurrence of $S y$ by $\neg T y$ is an implicit definition of $\neg Q$. However, the disjunction (resp. conjunction) of two such implicit definitions is not necessarily implicit (unless they define the same query). Moreover, it is impossible to individually define an implicit boolean query without making it explicit.

This leads us to generalize the notion of an implicit definition of an individual relation to an implicit definition of a *sequence of relations*. A sequence of global relations $(R_1, \ldots, R_m)$ is implicitly defined on a class of $\sigma$-structures $K$ by the sentence $\varphi(S_1, \ldots, S_m)$, where each $S_i \not\in \sigma$ is of the same arity as $R_i$, if for each $I$ in $K$, $(R_1, \ldots, R_m)$ is the unique sequence of relations that satisfies $\varphi$ on $I$. A query $Q$ of arity $k$ is implicitly definable in a sequence if it can be taken to be the
first global relation in such a sequence. The class of all such queries over a class of structures \( K \) is called \( \text{IMP}_m(K) \) [Kol90], where \( m \) is the number of relations defined in the sequence. Letting \( \text{IMP} = \bigcup \text{IMP}_m \), we get that \( \text{IMP} \) is closed under all first-order operations: complementation, conjunction, disjunction, and quantification.

It is possible to reduce the sequence to just two relations. Let \( Q \) be the query implicitly defined in the sequence determined by a sentence \( \varphi(S_1, \ldots, S_m) \) such that if \( \{ S'_1, \ldots, S'_m \} \) is the unique sequence of relations that satisfies \( \varphi \) on a structure \( I \), then \( Q(I) = S'_1 \). Then there exists an equivalent implicit definition given by \( \varphi(S, T) \) obtained from \( \varphi(S_1, \ldots, S_m) \), by replacing each occurrence of \( S_1 \) by \( S' \), and using \( T \) to encode the Cartesian product of \( S_2, \ldots, S_m \). Since a sequence of two relations suffices, \( \text{IMP} = \text{IMP}_2 \). The following example, which is an elaboration of an example in [Kol94], illustrates that \( \text{IMP}_1 \) is not as powerful.

**Fact:** The transitive-closure of a binary relation, TC, is not in \( \text{IMP}_1 \). Suppose that TC is implicitly definable in \( \text{IMP}_1 \), and defined by \( \psi(S) \) involving a unique symbol of relation \( S \) of arity 2. Then \( \psi' \), obtained from \( \psi(S) \) by replacing each occurrence of \( x \land y = y \), would be an explicit definition of connectivity (a graph is connected if its transitive closure is the Cartesian product of the domain). But connectivity is not first-order definable [Fag75].

**Example 3.2** Let \( \sigma \) be a vocabulary consisting in one binary relation \( E \). The binary query \( TC \) associated to a graph \( E \) its transitive closure has a definition \( \varphi(S, \leq) \) in \( \text{IMP} \) where \( S \) is of arity two and \( \leq \) of arity four. More precisely, \( \varphi(S, \leq) \) defines implicitly the sequence \( \langle TC, \leq \rangle \) where \( \psi(S, x, y) \equiv E x y \land \exists z (E x z \land S x y) \) is the fixed-point formula defining \( TC \) and \( \leq \) is similar to the stage comparison preorder defined in [Mos74]: for any binary tuples \( (a, b) \) and \( (a', b') \), \( (a, b) \leq \psi \) \( (a', b') \) iff \( d(a, b) \leq d(a', b') \), where \( d(a, b) \) measures the distance from \( a \) to \( b \), and is taken to be infinite is there is no path from \( a \) to \( b \). This construction follows Kolaitis’ one for positive fixed-point definitions [Kol90].

More generally, on the class \( \mathcal{F}, \text{FP} \subset \text{IMP} \). The inclusion was proved in [Lin87, Kol90]. Dawar, Hella and Kolaitis, showed that on the class \( \mathcal{F} \), there exist implicit queries which are not fixed-point definable [DHK95]. But simple polynomial-time queries such as \textit{Evenness} are not expressible in \( \text{IMP} \). In fact, over \( \mathcal{F} \), \( \text{IMP} \) is a complete co-recursively enumerable set of formulas. The problem in: we can enumerate (up to isomorphism) all finite \( \sigma \)-structures \( I \), and check if there is a \( R^I \) such that \( I \models \varphi(R^I) \) or if there are at least two different relations which satisfy \( \varphi(S) \) on \( I \).

More than the number of the implicitly defined relations in the sentence, it is the arity of the implicitly defined relation that has an impact on the expressive power. We prove that the full expressive power of implicit definitions in \( \text{IMP} \) can be obtained by only one implicitly defined relation associated with an explicit first-order transformation, instead of two implicitly defined relations.
Proposition 3.1 A k-ary query $Q$ ($k \geq 0$) is definable in IMP iff it has a definition of the form $\exists U \Psi(U) \land \varphi(U, \bar{x})$, where $\Psi(U)$ is a sentence in IMP, over $\sigma \cup \{U\}$, and $\varphi(U, \bar{x})$ is a formula with $k$ free variables.

Proof: Let $\psi(S, T)$ be a sentence of IMP that defines $Q$, $S$ and $T$ are relation symbols of arity respectively $k$ and $n$, that are not in $\sigma$. On every finite $\sigma$-structure $I$ there exists a unique sequence of relations $(S^I, T^I)$ such that $I \models \psi(S^I, T^I)$ and $S^I = Q(I)$. It is easy to encode the Cartesian product of $S$ and $T$ in a new relation symbol $U$ of arity $(n + k)$. There is a critical situation when on a structure $I$, the relation $T^I$ is the empty set. In fact we want to make sure that when $U^I$ is the empty set then $S^I$ is the empty set.

Consider the following definition $[\psi(S, T) \land \exists \bar{x}T\bar{x}] \lor \eta(S)$ where $\eta$ is obtained from $\psi(S, T)$ by replacing each occurrence of $T\bar{u}$ by $\bar{u} \neq \bar{u}$ (that forces $T$ to be the empty set). $[\psi(S, T) \land \exists \bar{x}T\bar{x}] \lor \eta(S)$ is a definition in IMP of $Q$. For each $\sigma$-structure $I$, $I \models \psi(S^I, T^I)$ then $I \models [\psi(S^I, T^I) \land \exists \bar{x}T^I\bar{x}] \lor \eta(S^I)$. $(S^I, T^I)$ is the unique sequence of relations that satisfy the latter definition.

Let $\psi'(U)$ be the sentence obtained from $[\psi(S, T) \land \exists \bar{x}T\bar{x}]$ replacing each occurrence of $S\bar{u}$ (resp. $T\bar{u}$) by $\exists \bar{u}\bar{v}\bar{w}$ (resp. $\exists \bar{v}\bar{u}\bar{w}$) in $[\psi(S, T) \land \exists \bar{x}T\bar{x}]$ and $\eta'(U)$ the one obtained from $\eta(S)$ replacing each occurrence of $S\bar{u}$ by $\forall \bar{u}\bar{v}\bar{w}$ in $\eta(S)$. The disjunction $\psi'(U) \lor \eta'(U)$ is still exclusive. Then, let $U^I$ be a $(k + n)$-ary relation on a $\sigma$-structure $I$ such that $I \models \psi'(U^I) \lor \eta'(U^I)$. $\Psi(U) \equiv \psi'(U) \lor \eta'(U) \in$ IMP since $\psi(S, T) \in$ IMP. More precisely, we have:

- if $U^I$ is not empty, then
  - $I \models \eta'(U^I)$ implies $U^I = S^I \times T^I$ and $I \models \psi(S^I, \emptyset)$ then $S^I \neq \emptyset$ and $T^I = \emptyset$,
  - $I \models \psi'(U^I)$ implies $U^I = S^I \times T^I$ and $I \models \psi(S^I, T^I)$ then $S^I, T^I \neq \emptyset$,
- if $U^I$ is empty, then
  - $I \models \eta'(U^I)$ implies $U^I = \emptyset \times I^I$ and $I \models \psi(\emptyset, \emptyset)$ then $S^I = T^I = \emptyset$,
  - $I \models \psi'(U^I)$ implies $U^I = \emptyset \times T^I$ and $I \models \psi(\emptyset, T^I)$ then $S^I = \emptyset$ and $T^I \neq \emptyset$.

Then $Q(I) = \{ \bar{a} : I \models \exists U \Psi(U) \land \exists \bar{x}U \bar{x} \bar{y} \}$. $Q$ is defined by $\Psi(U) \land \exists \bar{x}U \bar{x} \bar{y}$.

Conversely, if $Q$ is defined by the formula $\Psi(U) \land \varphi(U, \bar{x})$ where $\Psi$ is implicit, it is clear that $\psi(S, U) \equiv \Psi(U) \land \forall \bar{x} (S\bar{x} \rightarrow \varphi(U, \bar{x}))$ is a definition of $Q$ in IMP.

In the following, this normal form motivates our generalized implicit definitions.

4 Generalized Implicit Definitions

In this section, we extend the notion of an implicit definition. Instead of requiring a sentence to have a unique solution, we only require the uniqueness of solutions “modulo” another formula. Given a sentence $\psi(S)$ and a formula $\varphi(S, \bar{x})$, we require that on each structure, $\psi(S)$ always has at least one solution $S^I$, and
that for any such solution, \( \varphi(S', \bar{x}) \) is unique. In a sense, \( S \) is being used as a non-deterministic "working area" for the purpose of defining a deterministic query. We call these \textit{generalized} implicit definitions, and the class of queries they define, GIMP.

A definition of a \( k \)-ary query \( Q \), \( k \geq 0 \), in GIMP is split into two successive steps:

(i) \textit{non-deterministic}: the choice of a solution \( S' \) that satisfies the sentence \( \psi(S) \),

(ii) \textit{deterministic}: the evaluation of the formula \( \varphi(S', \bar{x}) \) with \( k \) free variables at \( S' \).

We say that \( \varphi(S, \bar{x}) \mod \psi(S) \) defines a generalized implicit query if for every finite \( \sigma \)-structure, there is a solution which makes \( \psi(S') \) true, and for each such solution \( \varphi(S', \bar{x}) \) determines the same relation.

Notice that although non-determinism is involved in defining the semantics of a query in GIMP, each query in GIMP has a deterministic semantics. The use of non-determinism to compute deterministic queries has been studied in [ASV90, Sch90, SZ90, GSZ95]. For a non-deterministic logical language \( \mathcal{L} \), Abiteboul, Simon and Vianu introduce two different deterministic semantics, the \textit{possibility} semantics, and the \textit{certainty} semantics [ASV90]. In the \textit{possibility} semantics, each formula of \( \mathcal{L} \) defines the set of tuples satisfying that formula for at least one of the non-deterministic choices. That is, the semantics is defined as the union of the possible non-deterministic outputs of the formula. In the \textit{certainty} semantics, each formula of \( \mathcal{L} \) defines the set of tuples satisfying that formula for every non-deterministic choice. That is, the semantics is defined as the intersection of the possible non-deterministic outputs of the formula.

Generalized implicit definitions are related to these semantics in the following way. \( \varphi(S, \bar{x}) \mod \psi(S) \) defines a GIMP query \( Q \) if the possibility semantics, \( \{ \bar{a} \in I^k : I \models \exists S \psi(S) \land \varphi(S, \bar{a}) \} \), and the certainty semantics, \( \{ \bar{a} \in I^k : I \models \forall S (\psi(S) \rightarrow \varphi(S, \bar{a})) \} \), coincide for every finite \( \sigma \)-structure \( I \). This leads to a new \textit{Possible-is-Certain} deterministic semantics defined as follows.

\textbf{Definition 4.1} For a non-deterministic language \( \mathcal{L} \), a formula of \( \mathcal{L} \) has the \textit{Possible-is-Certain} semantics if the set of tuples satisfying the formula for at least one of the non-deterministic choices coincides with the set of tuples satisfying the formula for all non-deterministic choices.

However, it is important to note that not every pair admits a Possible-is-Certain semantics, and that GIMP is precisely the queries defined by pairs which admit such a semantics.

More formally, we define GIMP queries as follows.

\textbf{Definition 4.2} A GIMP \( k \)-ary query \( Q \) over a class \( K \) of \( \sigma \)-structures, \( S \not\in \sigma \), is given by \( \varphi(S, \bar{x}) \mod \psi(S) \) where \( \psi(S) \) is a sentence such that

\( (i) \quad I \models_k \exists S \psi(S) \)
and \( \varphi(S, \bar{x}) \) is a formula with \( k \) free variables such that

\[
(ii) \quad \psi(S) \models_K \varphi(S, \bar{x}) \leftrightarrow Q(\bar{x}).
\]

**Remark**: Condition (ii) implies the \( S \)-invariance of \( \varphi(S, \bar{x}) \) with respect to \( \psi(S) \), that is:

\[
(ii') \quad \psi(S), \psi(S') \models_K \varphi(S, \bar{x}) \leftrightarrow \varphi(S', \bar{x}).
\]

**Example 4.1** The *Evenness* query (true on a finite structure \( B \) iff the cardinality of \( B \) is even) is in GIMP. Let \( S \) be a relational symbol of arity 3 that is not in the vocabulary. The formula \( \psi(S) \) will define the Cartesian product of a (strict) linear order and the set of even numbered elements with respect to it.

\[
\psi(S) \equiv \forall x \forall y \exists u Sxyu \vee x = y \vee \exists u Syxu \\
\wedge \forall x \forall y \forall u \forall v ((Sxu \wedge Syv) \rightarrow \exists w Sxzu) \\
\wedge \forall x [x = Min \rightarrow \forall y \forall z \neg Syxz] \\
\wedge \forall x \forall y \forall u \forall v [(Sxyu \wedge v = Suc(u)) \rightarrow \forall z \forall t \neg Sztv] \\
\wedge \forall x \forall y \forall u \forall v [(-Sxyu \wedge \exists w Sxuyw \wedge v = Suc(u)) \rightarrow Sxvy] 
\]

where \( x = Min \equiv \forall y \forall u \neg Syxu \) and

\[
v = Suc(u) \equiv \exists u Sxyu \wedge \forall z \forall u [Sxzv \rightarrow (\exists v Syzv \vee z = y)] \\
\wedge \forall z \forall u [Szyv \rightarrow (\exists w Szxv \vee z = x)].
\]

The first three expressions say that the projection of the first two components is a linear order: (1) says that \( < \) is total; (2) says that \( < \) is anti-symmetric; and (3) says that \( < \) is transitive. The next three expressions define the set of \( EVEN \) elements with respect to the order induced on the domain \( \{1, \ldots, n\} \) by the two first components: (4) says \( 1 \notin EVEN \); (5) says \( x \in EVEN \rightarrow x + 1 \notin EVEN \); and (6) says roughly that \( x - 1 \notin EVEN \rightarrow x \in EVEN \). Finally,

\[
\forall x [\forall y \forall u \neg Sxyu \rightarrow \exists y \exists z Syzx] \mod \psi(S)
\]

which says that maximal element is in \( EVEN \) is a definition of *Evenness* in GIMP.

It is easy to see that GIMP generalizes IMP\(_1\) by choosing \( \varphi(S, \bar{x}) \equiv S\bar{x} \). GIMP generalizes IMP since a definition in normal form \( \psi(U) \wedge \varphi(U, \bar{x}) \) of a query in IMP introduced in section 3 gives \( \varphi(U, \bar{x}) \mod \psi(U) \) in GIMP. Moreover, GIMP is closed under complementation, union, intersection, projection. The expressible power of GIMP admits a precise characterization in terms of complexity classes. We prove the following result.

**Theorem 4.1** On the class \( \mathcal{F} \) of all finite structures, GIMP = NP \( \cap \) co-NP.
Proof: We first show the easy direction. Consider a query $Q$ in GIMP, given by $\varphi(S, \bar{x}) \mod \psi(S)$. Observe that on each finite $\sigma$-structure $I$,

$$Q(I) = \{\bar{a} \in I^k : I \models \exists S \psi(S) \land \varphi(S, \bar{a})\} = \{\bar{a} \in I^k : I \models \forall S(\psi(S) \rightarrow \varphi(S, \bar{a}))\}. $$

Hence, $Q$ is included in both $\Sigma_1$ and $\Pi_1$. Therefore GIMP $\subseteq \Delta_1$. By Fagin’s result [Fag74], NP is characterized by $\Sigma_1$ on finite structures, so the queries computable in NP $\cap$ co-NP are exactly those that have a $\Delta_1$ definition. In fact, our proof will show that over any class of structures, GIMP is precisely $\Delta_1$.

Conversely, let $Q$ be a $k$-ary query definable in $\Delta_1$. Without loss of generality, we can assume that there exist formulas $\exists S \phi(S, \bar{x})$ and $\exists T \theta(T, \bar{x})$, where $\phi$ and $\theta$ are first-order formulas with relations $S$ and $T$ of the same arity $n$, respectively defining $Q$ and $\neg Q$ in $\Sigma_1$.

Let $I$ be a finite $\sigma$-structure with domain $I$. For all $\bar{a} \in I^k$, $\bar{a} \in Q(I)$ (respectively $\bar{a} \notin Q(I)$) iff there exists a relation $S_\bar{a}$ (respectively $T_\bar{a}$) such that $I \models \phi(S_\bar{a}, \bar{a})$ (respectively $I \models \theta(T_\bar{a}, \bar{a})$). Therefore, $I \models \forall \bar{x} [\exists S \phi(S, \bar{x}) \lor \exists T \theta(T, \bar{x})]$. Since the relation symbols $S$ and $T$ do not belong to $\sigma$, and are of the same arity, we can identify them to obtain $I \models \forall \bar{x} [\exists S \phi(S, \bar{x}) \lor \theta(T, \bar{x})]$. Let $S'$ be a new relation symbol of arity $(n + k)$, that intuitively encodes the skolemized dependency of $S$ relatively to $\bar{x}$, interpreted by one of the sets \{$(\bar{a}, \bar{b}) : \bar{a} \in Q(I)$ and $\bar{b} \in S_\bar{a}$, or $\bar{a} \notin Q(I)$ and $\bar{b} \in T_\bar{a}$\}. If we let $\psi(S')$ be the sentence $\forall \bar{x} [\phi'(S', \bar{x}) \lor \theta'(S', \bar{x})]$, where $\phi'(S', \bar{x})$ (resp. $\theta'(S', \bar{x})$) is obtained from $\phi(S, \bar{x})$ (resp. $\theta(S, \bar{x})$) by replacing each occurrence of $S(\bar{a})$ by $S'(\bar{x}, \bar{a})$, then we can see that $I \models \exists S' \psi(S')$.

To conclude we claim that $\phi'(S', \bar{x}) \mod \psi(S')$ defines $Q$ in GIMP. Let $I$ be a finite $\sigma$-structure and $S_\bar{a}$ any relation of arity $(n + k)$ such that $I \models \psi(S_\bar{a})$. We will show that $I \models \phi'(S_\bar{a}, \bar{a}) \Rightarrow \bar{a} \in Q(I)$. Indeed, $I \models \phi'(S_\bar{a}, \bar{a}) \Rightarrow I \models \phi'(\bar{b} : (\bar{a}, \bar{b}) \in S_\bar{a}, \bar{a} = \bar{a}) \in Q(I)$ by the $\Sigma_1$ definition of $Q$. For the other direction, $I \models \phi'(S_\bar{a}, \bar{a}) \Rightarrow I \models \phi'(\bar{b} : (\bar{a}, \bar{b}) \in S_\bar{a}, \bar{a} = \bar{a}) \Rightarrow I \models \phi'(\bar{b} : (\bar{a}, \bar{b}) \in S_\bar{a}, \bar{a} = \bar{a}) \Rightarrow \bar{a} \notin Q(I)$ by the $\Sigma_1$ definition of $\neg Q$. The proof is now complete.

Like IMP or $\Delta_1^1$, GIMP is not a recursively enumerable class of queries. The question whether or not $\varphi(S, \bar{x}) \mod \psi(S)$ is a correct definition in GIMP is clearly co-recursively enumerable. And indeed it is co-recursively enumerable complete since the validity problem of first-order sentences over finite structures can be reduced to it. For each first-order sentence $\theta$ over a vocabulary $\sigma$ containing some binary relation symbol, we consider $\neg \exists S \exists x (\theta \land \forall x (Sx)) \lor (\neg \exists \theta \land \exists x (Sx))$ which is a correct implicit definition in GIMP on the class of finite structures (of cardinality at least 2) if $\theta$ is valid. Trakhtenbrot’s theorem asserts that the set of first-order sentences that are finitely valid is a complete co-recursively enumerable set [Tra50].

The question of whether there exists a recursively enumerable logic for GIMP is connected with the existence of a complete problem for NP $\cap$ co-NP. Gurevich [Gur88] showed that if there exists a logic that captures NP $\cap$ co-NP then NP $\cap$ co-NP has a complete problem with respect to polynomial time reducibility. Since our correspondence is constructive, a recursively enumerable logic for GIMP would testify to the existence of a complete problem for NP $\cap$ co-NP. Gurevich
conjectures that if that were the case, something drastic would happen (like \( \text{NP} \cap \text{co-NP} = \text{P} \)).

**Proposition 4.2** Let \( \psi(S) \) be a FO definition and \( \mathcal{K} \) a class a finite \( \sigma \)-structures such that \( \mathcal{K} \models \exists S \psi(S) \). Let \( Q \) be a \( k \)-ary query which has a definition in GIMP over \( \sigma \cup \{ S \} \).

If \( Q \) is \( S \)-invariant mod \( \psi(S) \) on \( \mathcal{K} \), then \( Q \) has a definition in GIMP over \( \sigma \).

**5 Generalized Implicit Definitions with Order**

The generalized implicit definitions of GIMP proposed in Section 4 give a logical characterization of \( \text{NP} \cap \text{co-NP} \) over finite structures. In this section, we examine what happens if we restrict the general non-determinism to the choice of a total order, while retaining the classical (unique) implicit definitions for any other further implicitly defined relations. We’ll call this LIMP, a restriction of GIMP and a generalization of IMP, where the choice consists of a linear order \( < \) over the domain of the finite structure and a unique relation \( S \) on \( < \). More formally, we define LIMP queries as follows.

**Definition 5.1** A \( k \)-ary query \( Q \) over the class \( \mathcal{F} \) of finite \( \sigma \)-structures, with \( \prec, S \not\in \sigma \), is in LIMP if it can be defined by \( \varphi(\prec, S, \bar{x}) \mod \psi(\prec, S) \) where \( \psi(\prec, S) \) is a sentence such that for each total linear order \( \prec \) on \( I \in \mathcal{F} \) there exists a unique \( S_0 \) such that

(i) \( I \models \psi(\prec, S_0) \),

and \( \varphi(\prec, S, \bar{x}) \) is a formula with \( k \) free variables satisfying the following for each finite \( \sigma \)-structure \( I \), for each pair \( (\prec, S) \) satisfying \( \psi(\prec, S) \) on \( I \),

(ii) \( I \models \varphi(\prec, S, \bar{x}) \equiv Q(\bar{x}) \).

It is easy to see that LIMP \( \subseteq \text{GIMP} \) by encoding together the order \( \prec \) and the relation \( S \) into a new relation symbol \( U \). The *Evenness* query has a definition in LIMP since its non-deterministic part consists of the choice of a linear order \( \prec \) and the set of the even elements with respect to \( \prec \) (that is unique).

**Example 5.1** The *Evenness* query, true on a finite structure \( B \) iff the cardinality of \( B \) is even, has a definition in LIMP. Let \( S \) be a relational symbol of arity 1 that is not in the vocabulary. The formula \( \psi(\prec, S) \) defines the set \( S \) of even elements with respect to the chosen order \( \prec \).

\[
\psi(\prec, S) \equiv \forall x \left[ (\forall y \prec y x) \rightarrow \neg Sx \right] \\
\land \forall x \forall y \left[ Sx \leftarrow [\neg(\exists z \prec z \prec y) \rightarrow \neg Sy] \right]
\]

Then \( \forall x[\forall y \neg x \prec y \rightarrow Sx] \mod \psi(\prec, S) \) is a definition of *Evenness* in LIMP.
Since a definition in LIMP consists of the non-deterministic choice of a linear order, followed by an order-invariant part, it is natural to consider a one-one map from the class of queries \( Q \) over finite structures to the class of queries \( Q^* \) over finite ordered structures that are order-invariant. For each finite \( \sigma \)-structure \( I \) of size \( n \), let \( I_1, \ldots, I_n^* \) be the \( n! \) extensions of \( I \) over \( \sigma \cup \{<\} \). For each finite \( \sigma \cup \{<\} \)-structure \( I^* \), let \( I \) be its restriction to \( \sigma \), i.e., the finite \( \sigma \)-structure \( I \) with the same domain \( I \) and the same interpretation of the symbols in \( \sigma \).

For any \( k \)-ary query \( Q \) over the class of finite \( \sigma \)-structures, there exists a \( k \)-ary \( Q^* \) over finite \( \sigma \cup \{<\} \)-structures where \( < \) is interpreted by a linear order, such that, for each finite \( \sigma \)-structure \( I \) of size \( n \), \( Q(I) = Q^*(I_1^*) = \cdots = Q^*(I_n^*) \). It is clear that such a \( Q^* \) is order-invariant. Conversely, for each query \( Q^* \) on finite \( \sigma \cup \{<\} \)-structures that is order-invariant there is a query \( Q \) over finite \( \sigma \)-structures such that for all finite \( \sigma \cup \{<\} \)-structures \( I^* \), \( Q(I) = Q^*(I^*) \).

An interesting example of an order invariant query that is not first-order definable without the order relation is presented in ([AHV94], Exercise 17.27, page 462). The presence of an order has an important impact on definability because it rigidifies a finite structure. The decision problem of whether a first-order formula \( \phi(<, S, \bar{x}) \mod \psi(<, S) \), defines a semantically correct query in LIMP over finite \( \sigma \)-structures is undecidable. This is proved by an easy reduction from the validity of first-order sentences on finite structures. Let \( a \) be a first-order sentence over \( \sigma \) (containing some binary relation symbol), then consider \( \phi(<, x) \equiv (\neg a \land \forall y < y < x) \land (a \land x = x) \). \( \phi(<, x) \mod < \) is in LIMP iff \( \models \phi(a) \). Furthermore, there is a first-order \( \sigma \cup \{<\} \)-sentence \( \varphi \) such that the decision problem whether \( \varphi \) is order invariant on a \( \sigma \)-structure \( I \) is co-NP complete [Gur88].

If \( Q \) is a class of queries on finite ordered structures, then let Inv\(_<\)(\( Q \)) be the class of order-invariant queries in \( Q \). The following lemma says that an order-invariant query \( Q^* \) is in LIMP iff it is computable in UP \( \cap \) co-UP on finite ordered structures.

**Lemma 5.1** On the class \( \mathcal{O} \) of finite ordered structures,

\[
\text{Inv}\_<(\text{LIMP}) = \text{Inv}\_<(\text{U}\Delta_1^1) = \text{Inv}\_<(\text{UP} \cap \text{co-UP}).
\]

**Proof:** We first prove that \( \text{Inv}\_<(\text{LIMP}) = \text{Inv}\_<(\text{U}\Delta_1^1) \) on the class of finite structures ordered by \(< \). Consider a \( k \)-ary query \( Q^* \) in Inv\(_<\)(LIMP) defined by \( \phi(<, S, \bar{x}) \mod \psi(<, S) \) over \( \sigma \cup \{<\} \). The \( \Sigma_1^1 \) formula \( \exists S \phi(<, S, \bar{x}) \) is obtained from \( \psi(<, S) \) and \( \phi(<, S, \bar{x}) \) by replacing each occurrence of \(< \) by \( < \), also defines \( Q^* \) on the class of \( \sigma \cup \{<\} \)-structures ordered by \(< \). By definition of LIMP, we know that for any \( \sigma \cup \{<\} \)-structure \( I^* \), there exists a unique relation \( S_\bar{y}^* \) such that \( I^* \models \psi(<, S_\bar{y}^*) \). Hence, for all \( \bar{a} \in I^k, I^* \models \exists S \phi(<, S, \bar{a}) \). Therefore, \( Q^* \) has a definition in \( \text{U}\Delta_1^1 \). Since LIMP is closed under complementation, \( Q^* \) is definable in \( \text{U}\Delta_1^1 \). Recall that \( Q^* \) was order-invariant, so \( Q^* \in \text{Inv}_<(\text{U}\Delta_1^1) \).

In the other direction, let \( Q^* \) be a \( k \)-ary query in Inv\(_<\)(\( U\Delta_1^1 \)). So \( Q^* \) has a definition in \( U\Delta_1^1 \) that does not depend on the order. From [Kol90], we know
that IMP = UΔ₁ on any class of structures. So Q* has a definition in IMP. Since LIMP is a generalization of IMP, it follows that Inv< (IMP) ⊆ Inv< (LIMP). Again, since Q* is order-invariant, Q* ∈ Inv< (LIMP).

The second equality Inv< (UΔ₁) = Inv< (UP ∩ co-UP) comes from the fact that on finite ordered structures, UP = UΣ₁ [Kol90]. It follows that UP ∩ co-UP = UΣ₁ ∩ co-UΣ₁, and hence that Inv< (UΔ₁) = Inv< (UP ∩ co-UP). □

The following theorem shows more generally that LIMP also characterizes a complexity class on finite structures.

**Theorem 5.2** On the class ℱ of all finite structures, LIMP = UP ∩ co-UP.

**Proof:** Using the previous definitions, we first show that a query Q on finite structures is definable in LIMP iff its (order-invariant) extension Q* on finite ordered structures is definable in LIMP. The following are equivalent: (i) Q ∈ LIMP over σ, and (ii) Q* ∈ Inv< (LIMP) over σ ∪ {<}.

(i) ⇒ (ii) Let Q* be an order-invariant query defined by ϕ(<, S, x) mod ψ(<, S) over σ ∪ {<}. Its restriction Q has a definition ϕ'<(<, S, x) mod ψ(,<, S) in LIMP over σ obtained from ϕ(<, S, x) mod ψ(<, S) by replacing each occurrence of < by <.

Next we show that an order-invariant query Q* is computable in UP ∩ co-UP iff its restriction Q is in UP ∩ co-UP. The following are equivalent: (iii) Q ∈ UP ∩ co-UP over σ, and (iv) Q* ∈ Inv< (UP ∩ co-UP) over σ ∪ {<}.

(iii) ⇒ (iv) is immediate.

(iv) ⇒ (iii) Let Q* be a query in Inv< (UP ∩ co-UP) over σ ∪ {<}. Let ℳ* be an unambiguous Turing machine that computes Q*. Let ℳ be a machine whose input is a linear encoding of a σ-structure I that first uses the order of entrance of the elements of I to determine an order and then follows exactly the same transitions as ℳ*. ℳ is unambiguous (the encoding of the order is deterministic and the following computation is done by ℳ*) and computes Q. The same method gives an unambiguous machine that computes ¬Q. It follows that Q ∈ UP ∩ co-UP over σ.

The result now follows from Lemma 5.1. □

6 Conclusion

We have studied two natural extensions of first-order implicit definitions, and show that they characterize well known complexity classes, even over unordered finite structures. These extensions are based on the ability to non-deterministically choose an (intermediate) relation R satisfying a sentence φ(R), and use it to (deterministically) compute a query, independent of the particular choice of R. The first extension GIMP, in which an arbitrary relation is non-deterministically chosen, captures NP ∩ co-NP. The second extension LIMP, in which only an order relation on the domain is non-deterministically chosen, captures UP ∩ co-UP.
It might seem that the non-deterministic choice of an order relation should be enough to capture the full expressive power of GIMP (since it rigidifies the structure), and that maybe the definitions of LIMP could be seen as a normal form for GIMP queries. Nevertheless, it follows from the results presented here that the non-determinism can be reduced to the choice of an order relation if and only if \( UP \cap \text{co-UP} = \text{NP} \cap \text{co-NP} \).

Acknowledgments: The authors wish to thank Phokion Kolaitis, Richard Lassaigne, Michel de Rougemont and Victor Vianu for their suggestions and helpful discussions.

References


This article was processed using the \LaTeX\ macro package with LLNCS style.