Capturing Polynomial Time on Interval Graphs

Bastian Laubner
Institut für Informatik
Humboldt-Universität zu Berlin
laubner@informatik.hu-berlin.de

Abstract

We prove a characterization of all polynomial-time computable queries on the class of interval graphs by sentences of fixed-point logic with counting. More precisely, it is shown that on the class of unordered interval graphs, any query is polynomial-time computable if and only if it is definable in fixed-point logic with counting. This result is one of the first establishing the capturing of polynomial time on a graph class which is defined by forbidden induced subgraphs. For this, we define a canonical form of interval graphs using a type of modular decomposition, which is different from the method of tree decomposition that is used in most known capturing results for other graph classes, specifically those defined by forbidden minors. The method might also be of independent interest for its conceptual simplicity. Furthermore, it is shown that fixed-point logic with counting is not expressive enough to capture polynomial time on the classes of chordal graphs or incomparability graphs.

1 Introduction

Capturing results in descriptive complexity match the expressive power of a logic with the computational power of a complexity class. The most important open question in this area is whether there exists a natural logic whose formulas precisely define those queries which are computable in polynomial time (PTIME). While Immerman and Vardi showed in 1982 that fixed-point logic captures PTIME under the assumption that a linear order is present in each structure (cf. Theorem 2.4), there is no logic which is currently believed to capture PTIME on arbitrary unordered structures. Despite that limitation, precise capturing results for PTIME in the unordered case can be obtained for restricted classes of structures. Since all relational structures of a fixed finite vocabulary can be encoded efficiently as simple graphs, capturing results on restricted graph classes are of particular interest in this context.

This approach has been very fruitful in the realm of graph classes defined by lists of forbidden minors. Most of these results show that PTIME is captured by fixed-point logic with counting FP+C when restricting ourselves to one such class, such as planar graphs [12], graphs of bounded tree-width [15], or K_5-free graphs [13]. Grohe has recently announced a proof that FP+C captures PTIME on any graph class which is defined by a list of forbidden minors.

Given such deep results for classes of minor-free graphs, it is natural to ask if similar results can be obtained for graph classes which are defined by a (finite or infinite) list of forbidden induced subgraphs. Much less is known here. For starters, it is shown in [14] and in Section 3 that a general capturing result analogous to Grohe's is not possible for FP+C on subgraph-free graph classes, such as chordal graphs or graphs which are comparability graphs of partial orders. These two superclasses of interval graphs are shown to be a ceiling on the structural richness of graph classes on which capturing PTIME requires less effort than for general graphs.

Theorem 1.1. FP+C *fails to capture* PTIME *on the class of comparability graphs and on the class of chordal graphs.*

The main result in this paper is a positive one affirming that FP+C captures PTIME on the class of interval graphs. This means that a subset $\mathcal{K}$ of the class of interval graphs is decidable in PTIME if and only if there is a sentence of FP+C defining $\mathcal{K}$.

Theorem 1.2. FP+C *captures* PTIME *on the class of interval graphs.*

The result is shown by describing an FP+C-definable canonization procedure for interval graphs, which for any interval graph constructs an isomorphic copy on an ordered domain. The capturing result then follows from the Immerman-Vardi theorem. The proof of Theorem 1.2 also has a useful corollary.

Corollary 1.3. *The class of interval graphs is* FP+C-*definable.*

There has been persistent interest in the algorithmic aspects of interval graphs in the past decades, also spurred by their applicability to DNA sequencing (cf. [28]) and scheduling problems (cf. [25]). In 1976, Booth and Lueker presented the first recognition algorithm for interval graphs [1] running in time linear in the number of vertices and edges, which they followed up by a linear-time interval graph isomorphism algorithm [24]. These algorithms are based on a special data structure called *PQ-trees*. Using so-called perfect elimination orderings, Hsu and Ma [19] and Habib et al. [18] later presented linear-time recognition algorithms based on simpler data structures.

All these approaches have in common that they make inherent use of an underlying order of the graph, which is always available in PTIME computations as the order in which the vertices are encoded on the worktape. Particularly, the construction of a perfect elimination ordering by lexicographic breadth-first search needs to examine the children of a vertex in some fixed order. However, such an ordering is not available when defining properties of the bare unordered graph structure by means of logic. Therefore, most of the ideas developed in these publications cannot be applied in the canonization of interval graphs in FP+C.

We note that an algorithmic implementation of our method would be inferior to the existing linear-time algorithms for interval graphs. Given that our method must rely entirely on the inherent structure of interval graphs and not on an additional ordering of the vertices, we reckon that is the price to pay for the disorder of the graph structure.

The main commonality of existing interval graph algorithms and the canonical form developed here is the construction of a *modular decomposition* of the graph. Modules are subgraphs which interact with the rest of the graph in a uniform way, and they play an important algorithmic role in the construction of modular decompomposition trees (cf. [2]). As a by-product of the approach in this paper, we obtain a specific modular decomposition tree that is FP+C-definable. Such modular decompositions are fundamentally different from *tree decompositions*, which are the ubiquitous tool of FP+C-canonization proofs for the aforementioned minor-free graph classes (cf. [13] for a survey of tree decompositions in this context). Since tree decompositions do not appear to be very useful for defining canonical forms on subgraph-free graph classes, showing the definability of modular decompositions is a contribution to the systematic study of capturing results on these graph classes.

2 Preliminaries and notation

We write $\mathbb{N}$ and $\mathbb{N}_0$ for the positive and non-negative integers, respectively. For $m, n \in \mathbb{N}_0$, let $[m,n] := \{\ell \in \mathbb{N}_0 \mid m \leq \ell \leq n\}$ be the *closed interval of integers from m to n*, and let $[n] := [1,n]$. Tuples of variables $(v_1, \ldots, v_k)$ are often denoted by $\vec{v}$ and their length by $|\vec{v}|$.

2.1 Orders

Let us fix the terminology of the different types of orders which will be used in this paper.

Strict partial orders are defined in the usual way: a binary relation $<$ on a set X is a strict partial order if it is irreflexive and transitive, i.e., there is no $x \in X$ with $x < x$ and whenever $x < y$ and $y < z$, then also $x < z$. Irreflexivity and transitivity together imply antisymmetry, i.e., whenever $x < y$, it does not hold that $y < x$.

Two elements x, y of a partially ordered set X are called *incomparable* if neither $x < y$ nor $y < x$. We call $<$ a *strict weak order* if it is a strict partial order, and in addition, incomparability is an equivalence relation, i.e., whenever x is incomparable to y and y is incomparable to z, then x and z are also incomparable. If x, y are incomparable with respect to a strict weak order $<$, then $x < z$ implies $y < z$.

Finally, $<$ is a *(strict) linear order* if it is a strict partial order in which no two elements are incomparable. If $<$ defines a strict weak order on X and $\equiv$ is the equivalence relation defined by incomparability, then $<$ induces a linear order on $X/\!\equiv$.

2.2 Graphs

All graphs in this paper are assumed to be finite, simple, and undirected unless explicitly stated otherwise. Let $G = (V, E)$ be a graph with vertex set V and edge set E. Generally, E is viewed as a binary relation. Sometimes, we also find it convenient to view edges e as sets containing their two endpoints, as in $e = \{u, v\} \subseteq V$. For isomorphic graphs G and H we write $G \cong H$.

For $W \subseteq V$ a set of vertices, $G[W]$ denotes the *induced subgraph* of G on W. The *neighborhood* of a vertex $v \in V$, denoted $N(v)$, is the set of vertices adjacent to v under E, *including v itself*.

A subset $W \subseteq V$ is called a *clique* of G if $G[V]$ is a complete graph. A clique W of G is *maximal* if it is inclusion-maximal as a clique in G, i.e., if no vertex $v \in V \setminus W$ can be added to W so that $W \cup \{v\}$ is a clique of G. Since maximal cliques are central to the constructions in this paper, they will often just be called *max cliques*. *Cycles* in a graph are defined in the usual way.

If G is a *bipartite graph*, then we write $G = (U \dot\cup V, E)$ in order to emphasize that U and V are independent sets. Similarly, if G is a *split graph*, writing $G = (U \dot\cup V, E)$ emphasizes that $G[U]$ is a clique and V is an independent set.

The main result of this paper, Theorem 1.2, is about interval graphs, which we define and discuss now.

Definition 2.1 (Interval graph)**.** Let $\mathcal{I}$ be a finite collection of closed intervals $I_i = [a_i, b_i] \subset \mathbb{N}$. The graph $G_\mathcal{I} = (V, E)$ defined by $\mathcal{I}$ has vertex set $V = \mathcal{I}$ and edge relation $I_iI_j \in E :\Leftrightarrow I_i \cap I_j \neq \emptyset$. $\mathcal{I}$ is called an *interval representation* of a graph G if $G \cong G_\mathcal{I}$. A graph G is an *interval graph* if there is a collection of closed intervals $\mathcal{I}$ which is an interval representation of G.

If $v \in V$, then I_v denotes the interval corresponding to vertex v in $\mathcal{I}$. An interval representation $\mathcal{I}$ for an interval graph G is called *minimal* if the set $\bigcup \mathcal{I} \subset \mathbb{N}$ is of minimum size over all interval representations of G. Any interval representation $\mathcal{I}$ can be converted into a minimal interval representation by removing a set $K \subset \mathbb{N}$ from all intervals in $\mathcal{I}$ (and then considering the remaining points in $\mathcal{I}$ as an initial segment of $\mathbb{N}$).

If $\mathcal{I} = \{I_i\}_{i\in[n]}$ is a minimal interval representation of G, then there is an intimate connection between the maximal cliques of G and the sets $M(k) = \{I_i \mid k \in I_i\}$ for $k \in \mathbb{N}$. In fact, if $M(k) \neq \emptyset$ for some k, then $M(k)$ forms a clique which is maximal by the minimality condition on $\mathcal{I}$. Conversely, if M is a maximal clique of G, then $\bigcap_{v\in M} I_v = \{k\}$ for some $k \in \mathbb{N}$ by the minimality of $\mathcal{I}$, and $M(k) = M$. Thus, a connected graph G is an interval graph if and only if its max cliques can be arranged as a path so that each vertex of G is contained in consecutive max cliques. In this way, any minimal interval representation $\mathcal{I}$ of G induces an ordering $\lhd_\mathcal{I}$ of G's max cliques. We call a max clique C a possible *end* of G if there is a minimal interval representation $\mathcal{I}$ of G so that C is $\lhd_\mathcal{I}$-minimal.

Interval graphs are a classical example of an *intersection graph class* of certain objects. Intersection graphs have a collection of these objects $\{o_1, \dots, o_k\}$ as vertices with an edge between o_i and o_j if and only if $o_i \cap o_j \neq \emptyset$. Notice that any finite graph is the intersection graph of some collection of sets from $\mathbb{N}$, which is not the case when we restrict the allowed sets to intervals.

If $\mathcal{Y}$ is an intersection graph class, $G = (V, E) \in \mathcal{Y}$, and U is any subset of V, then $G[U]$ is also a member of $\mathcal{Y}$ since it is just the intersection graph of the objects in U. Any graph class $\mathcal{G}$ that is closed under taking induced subgraphs can also be defined by a possibly infinite list of *forbidden induced subgraphs*, by taking all those graphs not in $\mathcal{G}$ that are minimal with respect to the relation of being an induced subgraph. A complete infinite family of forbidden induced subgraphs defining the class of interval graphs is given by Lekkerkerker and Boland in [23].

Several further classes of graphs are important for this paper, and will be defined now.

Definition 2.2 (Chordal graph)**.** A graph is called *chordal* if all its induced cycles are of length 3.

It is easy to show that every interval graph is chordal. Chordal graphs can alternatively be characterized by the property that its maximal cliques can be arranged in a forest T, so that for every vertex of the graph the set of max cliques containing it is connected in T (cf. [4]).

Definition 2.3 (Comparability graph)**.** A graph $G = (V, E)$ is called a *comparability graph* if there exists a strict partial ordering $<$ of its vertex set V so that $uv \in E$ if and only if u, v are comparable with respect to $<$.

A graph is called an *incomparability graph* if its complement is a comparability graph. It is a well-known fact that every interval graph is an incomparability graph. In fact, a graph is an interval graph if and only if it is a chordal incomparability graph [8, 9].

2.3 Logics

We assume basic knowledge in logic, particularly of first-order logic FO. All structures considered in this paper are graphs $G = (V, E)$, i.e., relational structures with universe V and one binary relation E which is assumed to be symmetric and irreflexive. This section will introduce the fixed-point logics FP and FP+C. Detailed discussions of these logics can be found in [5, 10, 21].

If φ is a formula of some logic, we write $\varphi(x_1, \dots, x_k)$ to indicate that the free variables of φ are among $x_1, \dots, x_k$. If $v_1, \dots, v_k$ are vertices of a graph G, then $G \models \varphi[v_1, \dots, v_k]$ denotes that G satisfies φ if x_i is interpreted as v_i for all $i \in [k]$. Furthermore, $\varphi^G[v_1, \dots, v_{k-1}, \cdot]$ denotes the subset of vertices v_k in G for which $G \models \varphi[v_1, \dots, v_k]$, and similarly, $\varphi^G[\cdot, \dots, \cdot] = \{\vec{v} \in V^k \mid G \models \varphi[\vec{v}]\}$.

Inflationary fixed-point logic FP is the extension of FO by a fixed-point operator with inflationary semantics, which is defined as follows. Let $G = (V, E)$ be a graph, let X be a *relation variable* of arity r, and let $\vec{x}$ be a vector of r variables. Let φ be a formula whose free variables may include X as a free relation variable and $\vec{x}$ as free (vertex) variables. For any set $F \subseteq V^r$, let $\varphi[F]$ denote the set of r-tuples $\vec{v} \in V^r$ for which φ holds when X is interpreted as F and $\vec{v}$ is assigned to $\vec{x}$. Let the sets F_i be defined inductively by $F_0 = \varphi[\emptyset]$ and $F_{i+1} = F_i \cup \varphi[F_i]$. Since $F_i \subseteq F_{i+1}$ for all $i \in \mathbb{N}_0$, we have $F_k = F_{|V|^r}$ for all $k \geq |V|^r$. We call the r-ary relation $F_{|V|^r}$ the *inflationary fixed-point* of φ and denote it by $(\mathrm{ifp}_{X\leftarrow\vec{x}}\, \varphi)$. FP denotes the extension of FO with the ifp-operator.

In 1982, Immerman [20] and Vardi [27] showed that FP characterizes PTIME on classes of ordered structures[1].

[1] In fact, Immerman and Vardi showed this capturing result using a different fixed-point operator for *least fixed points*. Inflationary and least fixed points were shown to be equivalent by Gurevich and Shelah [17] and Kreutzer [22]. Also, Immerman and Vardi proved the result for general relational structures with an ordering, while we only state their theorem for graphs.

Theorem 2.4 (Immerman-Vardi). *Let $\mathcal{K}$ be a class of ordered graphs, i.e., graphs with an additional binary relation $<$ which satisfies the axioms of a linear order. Then $\mathcal{K}$ is* PTIME*-decidable if and only if there is a sentence of* FP *defining $\mathcal{K}$.*

When no ordering is present, then FP is not expressive enough to capture PTIME; in fact, it cannot even decide the parity of the underlying vertex set's size. For the capturing result in this paper, we will also need a stronger logic which is capable of such basic counting operations.

For this, let $G = (V, E)$ be a graph and let $N_V := [0, |V|] \subset \mathbb{N}_0$. Instead of G alone, we consider the two-sorted structure $G^+ := (V, N_V, E, <)$ with universe $V \dot\cup N_V$, so that E defines G on V and $<$ is the natural linear ordering of $N_V \subset \mathbb{N}_0$. Notice that E is not defined for any numbers from N_V, and also, $<$ does not give any order on V. Now we define FP-sentences on G^+ with the convention that all variables are implicitly typed. Thus, any variable x is either a vertex variable which ranges over V or a numeric variable which ranges over N_V.

The connection between the vertex and the number sort is established by *counting terms* of the form $\#x\,\varphi$ where x is a vertex variable and φ is a formula. $\#x\,\varphi$ denotes the number from N_V of vertices $v \in V$ so that $G \models \varphi[v]$. FP+C is now obtained by extending FP in the two-sorted framework with counting terms.

We can encode numbers from $[0, |N_V|^k - 1]$ with k-tuples of number variables. With the help of the fixed-point operator, we can do some meaningful arithmetic on these tuples, such as addition, multiplication, and counting the number of tuples $\vec{x}$ satisfying a formula $\varphi(\vec{x})$ (cf. [10]).

With its power to handle basic arithmetic, FP+C is already more powerful than FP on unordered graphs. Still, it is not powerful enough to capture PTIME by a result of Cai, Fürer and Immerman [3]. This fact will be used in the next section to prove similar negative results for specific classes of graph. For this, we still need the notion of a *graph interpretation*, which is a restricted version of the more general concept of a syntactical interpretation.

Definition 2.5. An *ℓ-ary graph interpretation* is a tuple $\Gamma = (\ \gamma_V(\vec{x}), \gamma_\approx(\vec{x}, \vec{y}), \gamma_E(\vec{x}, \vec{y}))$ of FO-formulas so that $|\vec{x}| = |\vec{y}| = \ell$ and $\gamma_\approx$ defines an equivalence relation $\approx$. If $G = (V, E)$ is a graph, then $\Gamma[G] = (V_\Gamma, E_\Gamma)$ denotes the graph with vertex set $V_\Gamma = \gamma_V^G[\cdot]/\approx$ and edge set $E_\Gamma = \gamma_E^G[\cdot, \cdot]/\approx^2$.

Lemma 2.6 (Graph Interpretations Lemma). *Let Γ be an ℓ-ary graph interpretation. Then for any* FP+C*-sentence φ there is a sentence $\varphi^{-\Gamma}$ with the property that $G \models \varphi^{-\Gamma} \iff \Gamma[G] \models \varphi$.*

Idea. A proof of this fact for first-order logic can be found in [6]. It essentially consists in modifying occurences of the edge relation symbol and quantification in φ with the right versions of γ_V, $\gamma_\approx$ and γ_E. Lemma 2.7 below is needed in order to deal with counting quantifiers in a sensible manner. We omit the details. □

2.4 FP+C-definable canonization

Results that prove the capturing of PTIME on a certain graph class usually do so by showing that there is a logically definable canonization mapping from the graph structure to the number sort. Theorem 1.2 will also be proved in this way, showing that there is an FP+C-formula $\varepsilon(x, y)$ with numeric variables x and y so that any interval graph $G = (V, E)$ is isomorphic to $\big([|V|], \epsilon^G[\cdot, \cdot]\big)$. Since the number sort N_V is linearly ordered, the Immerman-Vardi Theorem 2.4 then implies that any PTIME-computable property of interval graphs can be defined in FP+C.

Cai, Fürer and Immerman have observed in [3] that for graph classes which admit FP+C-definable canonization, a generic method known as the Weisfeiler-Lehman (WL) algorithm can be used to decide graph isomorphism in polynomial time (cf. [7]). Thus by Theorem 1.2, the WL algorithm also decides isomorphism of interval graphs. In the light of efficient linear-time isomorphism algorithms for interval graphs, the novelty here lies in the fact that a simple combinatorial algorithm decides interval graph isomorphism without specifically exploiting these graphs' inherent structure. The algorithm is generic in the sense that it also decides isomorphism of planar graphs, graphs of bounded treewidth, and many others.

2.5 Basic formulas

We finish this section by noting some basic constructions that can be expressed in FP+C. The existence of these formulas is essentially folklore, and variants of them can for example be found in [10]. These results lay the technical foundation for a higher-level description of the canonization procedure in Section 4. The proofs of the following lemmas are contained in the appendix.

Lemma 2.7 (Counting equivalence classes). *Suppose $\sim$ is an* FP+C*-definable equivalence relation on k-tuples of V, and let $\varphi(\vec{x})$ be an* FP+C*-formula with $|\vec{x}| = k$. Assume that $\sim$ has at most $|V|$ equivalence classes. Then there is an* FP+C*-counting term giving the number of equivalence classes $[\vec{v}]$ of $\sim$ such that $G \models \varphi[\vec{u}]$ for some $\vec{u} \in [\vec{v}]$.*

Let $\vec{y}$ be a tuple of numeric variables and let $\varphi(\vec{x}, \vec{y})$ be some formula. Using the ordering on the number sort and fixing $\vec{x}$, $\varphi[\vec{x}, \cdot]$ can be considered a 0-1-string of truth values of length $|N_V|^{|\vec{y}|}$. If $\vec{x}$ is a tuple of elements so that the string defined by $\varphi[\vec{x}, \cdot]$ is the lexicographically least of all such strings, then $\varphi[\vec{x}, \cdot]$ is called the *lexicographic leader*. Observe that such $\vec{x}$ need not be unique.

Lemma 2.8 (Lexicographic leader)**.** *Let $\vec{x}$ be a tuple of variables taking values in $V^k \times N_V^\ell$ and let $\vec{y}$ be a tuple of number variables. Suppose $\varphi(\vec{x}, \vec{y})$ is an* FP+C-*formula and $\sim$ is an* FP+C-*definable equivalence relation on $V^k \times N_V^\ell$. Then there is an* FP+C-*formula $\lambda(\vec{x}, \vec{y})$ so that for any $\vec{v} \in V^k \times N_V^\ell$, $\lambda^G[\vec{v}, \cdot]$ is the lexicographic leader among the relations $\{\varphi^G[\vec{u}, \cdot] \mid \vec{u} \sim \vec{v}\}$.*

Definition 2.9 (Lexicographic disjoint union)**.** Let $\mathcal{G} = \{G_i = (V_i, E_i)\}_{i \in [k]}$ be graphs whose universes V_i are initial segments $[|V_i|]$ of $\mathbb{N}$. Let $<$ be the lexicographic order on $\mathcal{G}$. Let π be a permutation of $[k]$ so that $G_{\pi(1)}, \ldots, G_{\pi(k)}$ is in lexicographic order. Then the *lexicographic disjoint union of $\mathcal{G}$* is the graph $G = (V, E)$ with universe $V = \left[\sum_{i \in [k]} |V_i|\right]$ so that for each $i \in [k]$, the restriction of G to $\left[\sum_{j \in [i-1]} |V_{\pi(j)}| + 1, \sum_{j \in [i]} |V_{\pi(j)}|\right]$ is order isomorphic to $G_{\pi(i)}$.

Lemma 2.10. *Suppose $\sim$ is an* FP+C-*definable equivalence relation on $V^k \times [|V|]^\ell$ and let $\upsilon(\vec{x}, y)$, $\epsilon(\vec{x}, y, z)$ be* FP+C-*formulas with number variables y, z defining graphs $\left(\upsilon^G[\vec{v}, \cdot], \epsilon^G[\vec{v}, \cdot, \cdot]\right)$ on the numeric sort for each $\vec{v} \in V^k \times [|V|]^\ell$. Furthermore, assume that $\upsilon^G[\vec{v}, \cdot] = \upsilon^G[\vec{v}', \cdot]$ whenever $\vec{v} \sim \vec{v}'$, and that $\sum_{[\vec{v}] \in V/\sim} |\upsilon^G[\vec{v}, \cdot]| \leq |V|$. Then there is an* FP+C-*formula $\omega(y, z)$ defining the lexicographic disjoint union of the lexicographic leaders of $\sim$'s equivalence classes on $\left[\sum_{[\vec{v}] \in V/\sim} |\upsilon^G[\vec{v}, \cdot]|\right]$*

3 Non-capturing results

This section contains some negative results of FP+C not capturing PTIME on a number of graph classes. In particular, Theorem 1.1 will be proven using a simple construction and the machinery of graph interpretations (see Definition 2.5).

The results in this section are all based on the following theorem due to Cai, Fürer, and Immerman [3].

Fact 3.1. *There is a* PTIME*-decidable property $\mathcal{P}_{\text{CFI}}$ of graphs of degree* 3 *which is not* FP+C*-definable.*

For any graph $G = (V, E)$, the *incidence graph* $G^I = (V \dot\cup E, F)$ is defined by $ve \in F :\Leftrightarrow v \in V$ and $v \in e \in E$. G^I is bipartite and it is straightforward to define a graph interpretation Γ so that for any graph G it holds that $\Gamma[G] \cong G^I$. Furthermore, given a graph G^I, it is a simple PTIME-computation to uniquely reconstruct G from G^I. Also, since the two partitions of a bipartite graph can be found in linear time, it is clear how to decide whether a given graph H is isomorphic to G^I for some graph G.

Theorem 3.2. FP+C *does not capture* PTIME *on the class of bipartite graphs.*

Proof. Recall the PTIME-decidable query $\mathcal{P}_{\text{CFI}}$ from Fact 3.1 and let $\mathcal{P}^I := \{H \mid H \cong G^I$ for some $G \in \mathcal{P}_{\text{CFI}}\}$. By the remarks above, $\mathcal{P}^I$ is PTIME-decidable.

So suppose that FP+C captures PTIME on the class of bipartite graphs. Then there is an FP+C-sentence φ such that for every bipartite graph G it holds that $G \models \varphi$ if and only if $G \in \mathcal{P}^I$. By an application of the Graph Interpretations Lemma 2.6 we then obtain a sentence $\varphi^{-\Gamma}$ so that $G \models \varphi^{-\Gamma}$ if and only if $G^I \cong \Gamma[G] \models \varphi$. Thus, $\varphi^{-\Gamma}$ defines $\mathcal{P}_{\text{CFI}}$, contradicting Fact 3.1. ☐

Theorem 1.1 is now a simple corollary of the following Lemma.

Lemma 3.3. *Every bipartite graph $G = (U \dot\cup V, E)$ is a comparability graph.*

Proof. A suitable partial order $<$ on $U \dot\cup V$ is defined by letting $u < v$ if and only if $u \in U$, $v \in V$, and $uv \in E$. ☐

Corollary 3.4. FP+C *does not capture* PTIME *on the class of incomparability graphs.* ☐

This tells us that being a comparability or incomparability graph alone is not sufficient for a graph G to be uniformly FP+C-canonizable. Section 4, however, is going to show that this is possible if G is both chordal *and* an incomparability graph, i.e., an interval graph. In a way, this is not simply a corollary of a capturing result on a larger class of graphs, as it is shown now that FP+C does not capture PTIME on the class of chordal graphs, either. The construction is due to Grohe [14].

For any graph $G = (V, E)$, the *split incidence graph* $G^S = (V \dot\cup E, \binom{V}{2} \cup F)$ is given by $ve \in F :\Leftrightarrow v \in V$ and $v \in e \in E$. Notice that G^S differs from G^I only by the fact that all former vertices $v \in V$ form a clique in G^S.

Given the similarity of G^S and G^I, the analysis for split incidence graphs is completely analogous to the one for incidence graphs above. In particular, the class $\mathcal{P}^S := \{H \mid H \cong G^S$ for some $G \in \mathcal{P}_{\text{CFI}}\}$ is PTIME-decidable and given a split graph H, the graph G for which $G^S \cong H$ can be reconstructed in PTIME if such G exists. Also, there is a graph interpretation Γ' so that for any graph G: $\Gamma'[G] \cong G^S$. The proof of the following theorem is then clear, and the subsequent lemmas complete the analysis for chordal graphs.

Theorem 3.5. FP+C *does not capture* PTIME *on split graphs.* ☐

Lemma 3.6. *Every split graph $G = (K, V, E)$ is chordal.* ☐

Corollary 3.7 (Grohe [14])**.** FP+C *does not capture* PTIME *on the class of chordal graphs.* ☐

Remark. In fact, the proofs here admit even stronger conclusions: any *regular logic* (cf. [6]) captures PTIME on the class of comparability graphs (respectively chordal graphs) if and only if it captures PTIME on the class of all graphs.

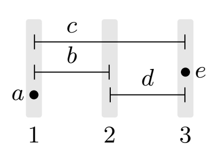
Let us conclude this section by noting some non-capturing results for further intersection graph classes. A t-interval graph is the intersection graph of sets which are the union of t intervals. By a result of Griggs and West [11], any graph of maximum degree 3 is a 2-interval graph, so Fact 3.1 directly implies that FP+C does not capture PTIME on t-interval graphs for $t \geq 2$. In [26], Uehara gives a construction that implies such a non-capturing result for intersection graphs of axis-parallel line segments in the plane. It follows that FP+C does not capture PTIME on boxicity-d graphs for $d \geq 2$, where a boxicity-d graph is the intersection graph of axis-parallel boxes in $\mathbb{R}^d$ and the boxicity-1 graphs are just the interval graphs.

4 Capturing PTIME on interval graphs

The goal of this section is to prove Theorem 1.2 by canonization. We will exhibit a numeric FP+C-formula $\varepsilon(x, y)$ so that for any interval graph $G = (V, E)$, $([|V|], \varepsilon^G[\cdot])$ defines a graph on the numeric sort of FP+C which is isomorphic to G. The canonization essentially consists of finding the lexicographic leader among all possible interval representations of G. For this, as discussed above, it is enough to bring the maximal cliques of G in the right linear order. The first lemma shows that the maximal cliques of G are FO-definable.

Lemma 4.1. *Let $G = (V, E)$ be an interval graph and let M be a maximal clique of G. Then there are vertices $u, v \in M$, not necessarily distinct, such that $M = N(u) \cap N(v)$.*

This is fairly intuitive. Consider the following minimal interval representation of a graph G.



Max cliques 1 and 3 are precisely the neighborhoods of vertices a and e, respectively. The vertex pairs (a, a), (a, b), and (a, c) all define max clique 1, and similarly three different vertex pairs define max clique 3. Max clique 2 is not the neighborhood of any single vertex, but it is uniquely defined by $N(b) \cap N(d)$.

Proof of Lemma 4.1. Let $\mathcal{I}$ be a minimal interval representation of G. First assume that M is the $\lhd_{\mathcal{I}}$-least maximal clique. The lemma is trivial if M is the only maximal clique of G, otherwise let X be M's $\lhd_{\mathcal{I}}$-successor. Since $M \neq X$ there is $v \in M \setminus X$, and M is the only maximal clique of G that v is contained in (as v is contained in $\lhd_{\mathcal{I}}$-consecutive max cliques). Hence, $M = N(v)$. A symmetric argument holds if M is the $\lhd_{\mathcal{I}}$-greatest maximal clique. Now assume that M is neither $\lhd_{\mathcal{I}}$-least nor maximal and let X, Y be M's immediate $\lhd_{\mathcal{I}}$-predecessor and successor, respectively. There exist $x \in M \setminus X$ and $y \in M \setminus Y$, and we claim that $N(x) \cap N(y) = M$. In fact, since any vertex in M is contained both in $N(x)$ and $N(y)$, we have $M \subseteq N(x) \cap N(y)$. Now let $u \in N(x) \cap N(y)$ and write $I_u = [a, b]$. Let k be the (unique) integer such that $M(k) = M$. Then $ux \in E$ implies $b \geq k$, and $vx \in E$ implies $a \leq k$. Thus, $k \in I_u$ and hence $u \in M$, which proves the claim. □

Now, whether or not a vertex pair $(u, v) \in V^2$ defines a max clique is easily definable in FO, as is the equivalence relation on V^2 of vertex pairs defining the same max clique. Lemma 4.1 tells us that *all* max cliques can be defined by such vertex pairs. For any $v \in V$, let the *span of* v, denoted $\mathrm{span}(v)$, be the number of max cliques of G that v is contained in. Since equivalence classes can be counted by Lemma 2.7, $\mathrm{span}(x)$ is FP+C-definable on the class of interval graphs by a counting term with x as a free vertex variable.

Generally representing max cliques by pairs of variables $(x, y) \in V^2$ allows us to treat max cliques as first-class objects that can be quantified over. For reasons of conceptual simplicity, the syntactic overhead which is necessary for working with this representation will not be made explicit in the remainder of this proof.

4.1 Extracting information about the order of the maximal cliques

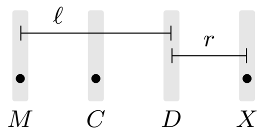
Now that we are able to handle maximal cliques, we would like to simply pick an end of the interval graph G and work with the order which this choice induces on the rest of the maximal cliques. Of course, the choice of an end does not necessarily impose a linear order on the maximal cliques. However, the following recursive procedure turns out to recover all the information about the order of the max cliques induced by choosing an end of G.

Let $\mathcal{M}$ be the set of maximal cliques of an interval graph $G = (V, E)$ and let $M \in \mathcal{M}$. The binary relation $\prec_M$ is defined recursively on the elements of $\mathcal{M}$ as follows:

$$\text{Initialization:} \quad M \prec_M C \text{ for all } C \in \mathcal{M} \setminus \{M\}$$

$$C \prec_M D \quad \text{if} \begin{cases} \exists v \in C \setminus D \text{ and } \exists E \in \mathcal{M} \text{ with} \\ \qquad E \prec_M D \text{ and } v \in E, \text{ or} \\ \exists v \in D \setminus C \text{ and } \exists E \in \mathcal{M} \text{ with} \\ \qquad C \prec_M E \text{ and } v \in E. \end{cases} \tag{$\bigstar$}$$

The following interval representation of a graph G illustrates this definition.



Suppose we have picked max clique M, then $C \prec_M X$ and $D \prec_M X$ since $\ell \in C \cap D \cap M \setminus X$ and $M \prec_M X$ by the initialization step. In a second step, it is determined that $C \prec_M D$ since $r \in D \cap X \setminus C$ and $C \prec_M X$. So in this example, $\prec_M$ actually turns out to be a strict linear order on the max cliques of G. This is not the case in general, but $\prec_M$ will still be useful when M is a possible end of G. The definition of $\prec_M$ seems natural to me for the task of ordering the max cliques of an interval graphs. However, I am not aware of it appearing previously anywhere in the literature.

It is readily seen how to define $\prec_M$ using the inflationary fixed-point operator, where maximal cliques are defined by pairs of vertices from G. The following lemmas prove important properties of $\prec_M$.

Lemma 4.2. *If $\prec_M$ is antisymmetric, then it is transitive. Thus, if $\prec_M$ is antisymmetric, then it is a strict partial order.*

Proof. By a derivation chain of length k we mean a finite sequence $X_0 \prec_M Y_0$, $X_1 \prec_M Y_1$, …, $X_k \prec_M Y_k$ such that $X_0 = M$ and for each $i \in [k]$, the relation $X_i \prec_M Y_i$ follows from $X_{i-1} \prec_M Y_{i-1}$ by one application of (★). Clearly, whenever it holds that $X \prec_M Y$ there is a derivation chain that has $X \prec_M Y$ as its last element.

Suppose $A \prec_M B \prec_m C$ and let a derivation chain $(L_0, \ldots, L_a)$ of length a be given for $A \prec_M B$. The proof is by induction on a. If $a = 0$, then $A = M$ and $A \prec_M C$ holds. For the inductive step, suppose $a = n$ and consider the second to last element L_{n-1} in the derivation chain. There are two cases:

- $L_{n-1} = (X \prec_M B)$ and there is a vertex $v \in (X \cap A) \setminus B$: By induction it holds that $X \prec_M C$. Now if we had $v \in C$, the fact that $A \prec_M B$ would imply $C \prec_M B$, which contradicts antisymmetry of $\prec_M$. Hence, $v \notin C$ and one more application of (★) yields $A \prec_M C$.

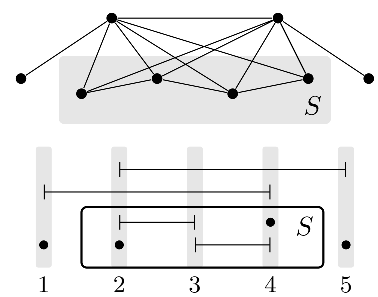
- $L_{n-1} = (A \prec_M X)$ and there is a vertex $v \in (X \cap B) \setminus A$: If $v \in C$, then we immediately get $A \prec_M C$. If $v \notin C$, then $X \prec_M C$. Thus we can derive $A \prec_M X \prec_M C$ where the left derivation chain has length $n - 1$. By induction, $A \prec_M C$ follows.

This completes the inductive step. □

Lemma 4.3. *Suppose M is a max clique of G so that $\prec_M$ is a strict partial order and $\mathcal{C}$ is a maximal set of incomparable max cliques. Let $D \in \mathcal{M} \setminus \mathcal{C}$. Then $D \cap C = D \cap C'$ for all $C, C' \in \mathcal{C}$.*

Proof. Suppose there are $C, C' \in \mathcal{C}$ and $D \in \mathcal{M} \setminus \mathcal{C}$ with $C \cap D \neq C' \cap D$. Without loss of generality assume that there is an element $x \in (C \cap D) \setminus C'$ (otherwise, swap C for C'). As $\mathcal{C}$ is maximal, there is some $C_D \in \mathcal{C}$ such that $C_D \prec_M D$ or $D \prec_M C_D$. Without loss of generality assume the former. Now if $x \notin C_D$, then (★) implies $C_D \prec_M C$, which contradicts the assumption that the elements of $\mathcal{C}$ are mutually incomparable. But if $x \in C_D$, then (★) implies that $C' \prec_M C_D$ since $x \notin C'$, again contradicting incomparability. □

Lemma 4.3 says that if we are lucky enough to choose a max clique M of G so that $\prec_M$ becomes a strict partial order, then incomparable max cliques interact with the rest of $\mathcal{M}$ in a uniform way. Let us make this notion more precise. A *module* of G is a set $S \subseteq V$ so that for any vertex $x \in V \setminus S$, S is either completely connected or completely disconnected to x. In other words, for all $u, v \in S$ and all $x \in V \setminus S$ it holds that $ux \in E \leftrightarrow vx \in E$. The next drawing illustrates the occurrence of a module in an interval graph.



Corollary 4.4. *Suppose M is a max clique of G so that $\prec_M$ is a strict partial order and $\mathcal{C}$ is a maximal set of incomparable max cliques. Then*

- $S_\mathcal{C} := \bigcup_{C \in \mathcal{C}} C \setminus \bigcup_{D \in \mathcal{M} \setminus \mathcal{C}} D$ *is a module of G, and*

- $S_\mathcal{C} = \{v \in \bigcup \mathcal{C} \mid \operatorname{span}(v) \leq |\mathcal{C}|\}$.

Proof. Let $u, v \in S_\mathcal{C}$ and $x \in V \setminus S_\mathcal{C}$ and suppose that $ux \in E$, but $vx \notin E$. There is a max clique $C \in \mathcal{M}$ with $u, x \in C$, but $v \notin C$, and since $u \in S$ we must have $C \in \mathcal{C}$. By the definition of $S_\mathcal{C}$, x is also contained in some max clique $D \in \mathcal{M} \setminus \mathcal{C}$. Finally, let C' be some max clique in $\mathcal{C}$ containing v, so $x \notin C'$. Thus, $D \cap C \neq D \cap C'$, contradicting Lemma 4.3.

For the second statement, let $v \in \bigcup \mathcal{C}$. If $v \in S_\mathcal{C}$, then clearly $\operatorname{span}(v) \leq |\mathcal{C}|$. But if $v \notin S_\mathcal{C}$, then it is contained in some $D \in \mathcal{M} \setminus \mathcal{C}$, and by Lemma 4.3 v must also be contained in all max cliques in $\mathcal{C}$. Thus, $\operatorname{span}(v) > |\mathcal{C}|$, proving the statement. □

This characterization of the modules occuring when defining the relations $\prec_M$ will be central in the canonization procedure of G. There is another corollary of Lemma 4.3 which proves that $\prec_M$ has a particularly nice structure.

Corollary 4.5. *If M is a max clique of G so that $\prec_M$ is a strict partial order, then $\prec_M$ is a strict weak order.*

Proof. We need to prove that incomparability is a transitive relation of max cliques of G. So let (A, B) and (B, C) be incomparable pairs with respect to $\prec_M$. Then by (★) the following holds:

$\neg\exists v \in A \setminus B:\ \exists F \in \mathcal{M}$ comparable to A and $v \in F$, and
$\neg\exists v \in C \setminus B:\ \exists F \in \mathcal{M}$ comparable to C and $v \in F$

Suppose for contradiction that A and C were comparable, and let $\mathcal{C}_{AB}$ and $\mathcal{C}_{BC}$ be maximal set of incomparables containing $\{A, B\}$ and $\{B, C\}$, respectively. Since $A \notin \mathcal{C}_{BC}$ and $C \notin \mathcal{C}_{AB}$, Lemma 4.3 implies that $A \cap B = A \cap C = C \cap B$. But then $A \setminus B = A \setminus C$ and $C \setminus B = C \setminus A$, and the two statements above become

$\neg\exists v \in A \setminus C:\ \exists F \in \mathcal{M}$ comparable to A and $v \in F$, and
$\neg\exists v \in C \setminus A:\ \exists F \in \mathcal{M}$ comparable to C and $v \in F$.

Thus, A and C are incomparable, contradicting the above assumption and proving the corollary. □

At this point, let us put the pieces together and show that picking an arbitrary max clique M as an end of G and defining $\prec_M$ is a useful way to obtain information about the structure of G.

Lemma 4.6. *Let M be a max clique of an interval graph G. Then $\prec_M$ is a strict weak order if and only if M is a possible end of G.*

Proof. If M is a possible end of G, then let $\mathcal{I}$ be a minimal interval representation of G which has M as its first clique. Let $\lhd_{\mathcal{I}}$ be the linear order $\mathcal{I}$ induces on the max cliques of G. In order to show antisymmetry it is enough to observe that, as relations, we have $\prec_M \subseteq \lhd_{\mathcal{I}}$. It is readily verified that this holds true of the initialization step in the recursive definition of $\prec_M$, and that whenever max cliques C, D satisfy (★) with $\prec_M$ replaced by $\lhd_{\mathcal{I}}$, then it must hold that $C \lhd_{\mathcal{I}} D$. This shows antisymmetry, and by Lemma 4.2 and Corollary 4.5 $\prec_M$ is a strict weak order.

Conversely, suppose $\prec_M$ is a strict weak order. The first aim is to turn $\prec_M$ into a linear order. Let $\mathcal{C}$ be a maximal set of incomparable max cliques, and recall the set $S_{\mathcal{C}} = \bigcup_{C\in\mathcal{C}} C \setminus \bigcup_{D\in\mathcal{M}\setminus\mathcal{C}} D$. Since $G[S_{\mathcal{C}}]$ is an interval graph, we can pick an interval representation $\mathcal{I}_{S_{\mathcal{C}}}$ for $G[S_{\mathcal{C}}]$. The set of max cliques of $G[S_{\mathcal{C}}]$ is given by $\{C \cap S_{\mathcal{C}} \mid C \in \mathcal{C}\}$, and since $S_{\mathcal{C}}$ is a module, $C \cap S_{\mathcal{C}} \neq C' \cap S_{\mathcal{C}}$ for any $C \neq C'$ from $\mathcal{C}$. Thus, $\mathcal{I}_{S_{\mathcal{C}}}$ induces a linear order $\lhd_{\mathcal{C}}$ on the elements of $\mathcal{C}$. Now let $C \lhd_M D$ if and only if $C \prec_M D$, or $C, D \in \mathcal{C}$ for some maximal set of incomparables $\mathcal{C}$ and $C \lhd_{\mathcal{C}} D$. This is a strict linear order since $\prec_M$ is a strict weak order. We claim that $\lhd_M$ is an ordering of the max cliques which is isomorphic to the linear order induced by some interval representation of G. This will imply that M is a possible end of G.

In order to prove the claim, it is enough to show that each vertex $v \in V$ is contained in consecutive max cliques. Suppose for contradiction that there are max cliques $A \lhd_M B \lhd_M C$ and v is contained in A and C, but not in B. Certainly, this cannot be the case if A, B, C are incomparable with respect to $\prec_M$, so assume without loss of generality that $A \prec_M B$. Now, since $v \in (A \cap C) \setminus B$, (★) implies that $C \prec_M B$, which contradicts the antisymmetry of $\lhd_M$. □

Remark. The recursive definition of $\prec_M$ and Lemma 4.2 through Corollary 4.5 do not depend on G being an interval graph. However, the proof of Lemma 4.6 shows that $\prec_M$ only turns out to be a partial order if the max cliques can be brought into a linear order, modulo the occurrence of modules. In particular, defining $\prec_M$ in a general chordal graph does not yield any useful information if the graph's tree decomposition into max cliques requires a tree vertex of degree 3 or more, which is the case for all chordal graphs which are not interval graphs.

4.2 Canonizing when $\prec_M$ is a linear order

Since $\prec_M$ is FP-definable for any max clique M, and since antisymmetry of $\prec_M$ is FO-definable, Lemma 4.6 gives us a way to define possible ends of interval graphs in FP. Moreover, if M is a possible end of $G = (V, E)$, then $\prec_M$ contains precisely the ordering imposed on the max cliques of G by the choice of M as the first clique.

First, suppose that $G = (V, E)$ is an interval graph and $\prec$ is a linear order on the max cliques which is induced by an interval representation of G. Define the binary relation $<^G$ on the vertices of G as follows. For $x \in V$, let A_x be the $\prec$-least max clique of G containing x. Then let

$$x <^G y :\Leftrightarrow \begin{cases} A_x \prec A_y, \text{ or} \\ A_x = A_y \text{ and } \operatorname{span}(x) < \operatorname{span}(y). \end{cases}$$

It is readily verified that $<^G$ is a strict weak order on V, and if x, y are incomparable, then $N(x) = N(y)$. Now it is easy to canonize G: if $[v]$ denotes the equivalence class of vertices incomparable to v, then $[v]$ is represented by the numbers from the interval $[a + 1, a + |[v]|]$, where a is the number of vertices which are strictly $<^G$-smaller than v. Since all vertices in $[v]$ have precisely the same neighbors in $G\setminus[v]$ and $[v]$ forms a clique, it is also clear how to define the edge relation on the number sort.

Now if G is any interval graph and M is a possible end, we can still define an ordering for those vertices that are not contained in a module. Let $\sim_M^G$ be the equivalence relation on V for which $x \sim_M^G y$ if and only if $x = y$ or there is a nonsingular maximal set of incomparables $\mathcal{C}$ with respect to $\prec_M$ so that $x, y \in S_\mathcal{C}$. Denote the equivalence class of $x \in V$ under $\sim_M^G$ by $[x]$, and define the edge relation E_M of the graph $G_M = (V/\sim_M^G, E_M)$ by $[u][v] \in E_M :\Leftrightarrow \exists x \in [u], y \in [v]$ s.t. $xy \in E$. It follows directly from the definition of $\sim_M^G$ that if A is a max clique which is $\prec_M$-comparable to all other max cliques in G, then all $v \in A$ are in singleton equivalence classes $[v] = \{v\}$. If $\mathcal{C}$ is a nonsingular maximal set of incomparables, then there is precisely one max clique C in G_M which contains all the equivalence classes associated with $\mathcal{C}$, i.e., $C = \{[v] \mid v \in \bigcup \mathcal{C}\}$. Thus $\prec_M$ induces a strict linear order on the max cliques of G_M. In fact, this shows that G_M is an interval graph with a valid interval representation induced by $\prec_M$.

4.3 Canonizing general interval graphs

What is left is to deal with the sets $S_\mathcal{C}$ coming from maximal sets of incomparables. Let $P' = \{(M, n) \mid M \in \mathcal{M}, n \in [|V|]\}$. For each $(M, n) \in P'$ define $V_{M,n}$ as the set of vertices of the connected component of $V \setminus \{v \in V \mid \operatorname{span}(v) > n\}$ which intersects M (if nonempty). Notice that $M_n := M \cap V_{M,n}$ is a max clique of $G[V_{M,n}]$. Finally, let P be the set of those $(M, n) \in P'$ for which defining $\prec_{M_n}$ in $G[V_{M,n}]$ yields a strict partial order of $G[V_{M,n}]$'s max cliques.

It is immediate from Corollary 4.4 that for any maximal set of incomparable max cliques $\mathcal{C}$, $S_\mathcal{C} = \bigcup_{C \in \mathcal{C}} V_{C,|\mathcal{C}|}$. In this situation, for any $C \in \mathcal{C}$, the set $V_{C,|\mathcal{C}|}$ defines a component of $S_\mathcal{C}$, and $(C, |\mathcal{C}|) \in P$ if and only if $C \cap S_\mathcal{C}$ is a possible end of (one of the components of) $G[S_\mathcal{C}]$. This gives us enough structure to perform canonization.

Proof of Theorem 1.2. We define the relation $\epsilon(M, n, x, y)$ inductively, where $(M, n) \in P$ and x, y are number variables. $([|V_{M,n}|], \epsilon^G[M, n, \cdot, \cdot])$ will be an isomorphic copy of $G[V_{M,n}]$ on the numeric sort. To this end, start defining ϵ for all $(M, 1) \in P$, then for all $(M, 2) \in P$, and so on, up to all $(M, |V|) \in P$.

Suppose we want to define ϵ for $(M, n) \in P$, then first compute the strict weak order $\prec_{M_n}$ on the interval graph $G[V_{M,n}]$. Consider any nonsingular maximal set of incomparables $\mathcal{C}$ and let $m := |\mathcal{C}|$. Let $H_1, \ldots, H_h$ be a list of the components of $G[S_\mathcal{C}]$ and let H_i be such a component. By the above remarks, there exist at least two $C \in \mathcal{C}$ so that $V_{C,m} = H_i$ and $(C, m) \in P$.

Notice that by the definitions of P and $\prec_{M_n}$, we have $m < n$ and therefore all $\epsilon^G[C, m, \cdot, \cdot]$ with $C \in \mathcal{C}$ have already been defined. Let $\sim$ be the equivalence relation on $P \cap (\mathcal{C} \times \{m\})$ defined by $(C, m) \sim (C', m) :\Leftrightarrow V_{C,m} = V_{C',m}$. Using Lemma 2.10, we obtain the lexicographic disjoint union $\omega_\mathcal{C}(x, y)$ of the lexicographic leaders of $\sim$'s equivalence classes.

Finally, let $<_M^{G_{M,n}}$ be the strict partial order on $V_{M,n}/\sim_M^{G_{M,n}}$ defined above. Let $c_1, \ldots c_k$ be the list of non-singular equivalence classes of $\sim_M^{G_{M,n}}$. Each c_i is associated with a unique maximal set of incomparables $\mathcal{C}_i$, and $c_i = S_{\mathcal{C}_i}$ as sets. We aim at canonizing $G_{M,n}$ using $<_M^{G_{M,n}}$, inserting the graph defined by $\omega_{\mathcal{C}_i}(x, y)$ in place of each c_i. Here is how: each $[v] \in V_{M,n}/\sim_M^{G_{M,n}}$ is represented by the interval $[a{+}1, a{+}|[v]|]$, where a is the number of vertices in equivalence classes strictly $<_M^{G_{M,n}}$-smaller than $[v]$. Since all vertices in $[v]$ have the same neighbors in all of $G \setminus [v]$, it is clear how to define the edge relation between $[v]$ and $G \setminus [v]$. If $[v]$ is not a singleton set, then $c_i = [v]$ for some i and the edge relation on c_i is given by $\omega_{\mathcal{C}_i}(x, y)$.

It is clear from the construction that $([|V_{m,n}|], \epsilon^G[M, n, \cdot, \cdot]) \cong G[V_{M,n}]$. Also, $\epsilon(M, n, x, y)$ can be defined in FP+C for all $(M, n) \in P$ using a fixed point-operator iterating n from 1 to $|V|$. Finally, let $\varepsilon(x, y)$ be the lexicographic disjoint union of the lexicographic leaders canonizing the components of G, each of which is defined by some $(M, |V|) \in P$. Then $([|V|], \varepsilon^G[\cdot, \cdot]) \cong G$, which concludes the canonization of G. □

Proof of Corollary 1.3. It is clear from the proof of Theorem 1.2 that alongside the canonical form, it is possible to construct a FP+C-formula $\prec$ defining a *linear order* on the original graph's max cliques which corresponds to the linear order of the canonical form's max cliques. $\prec$ defines a strict weak order $<_G$ on the vertices of G in the same way as defined in Section 4.2.

Now given any graph G, there are three possibilities what can happen when we attempt its canonization as an interval graph. If the canonization procedure fails to produce a graph on the number sort, then G is not an interval graph. If ϵ does define a graph on the number sort, then it must be in orderly one-to-one correspondence with G's vertices ordered by $<_G$. If this is not the case, then G is not an interval graph. If it is, then ϵ defines an isomorphic copy of G on the number sort, and the PTIME-property of being an interval graph can be verified by the Immerman-Vardi theorem. □

5 Conclusion

We have proved that the class of interval graphs admits FP+C-definable canonization. Thus, FP+C captures PTIME on the class of interval graphs, which was shown not to be the case for any of the two obvious superclasses of interval graphs: chordal graphs and incomparability graphs.

The result implies that the combinatorial Weisfeiler-Lehman algorithm solves the isomorphism problem for interval graphs. Verbitsky has confirmed in private communication that using the methods of this paper, the WL algorithm can even be shown to find a canonical labeling of interval graphs in a logarithmic number of steps. By a result of Grohe and Verbitsky [16], this implies the existence of a TC^1 parallel algorithm for the interval graph isomorphism problem, which is an improvement over the currently known NC^2 procedures.

Among the graph classes considered in this paper, the only class whose status is not settled with respect to FP+C-canonization is the class of chordal comparability graphs. While it appears that the methods employed for chordal incomparability graphs here do not carry over (see Remark 4.1), I believe I have found a different solution, which will be contained in the journal version of this paper.

So far, little is known about logics capturing complexity classes on classes of graphs which are defined by a (finite or infinite) list of forbidden induced subgraphs. This paper makes a contribution in this direction. It seems that chordal graphs, even though they do not admit FP+C-canonization themselves, can often be handled effectively in fixed-point logic as soon as additional properties are satisfied (being a line graph, incomparability or comparability graph). It would be instructive to unify these properties. In this context, I would also like to point to Grohe's conjecture [14] that FP+C captures PTIME on the class of claw-free chordal graphs.

Acknowledgement

I would like to thank Martin Grohe for bringing the question of capturing PTIME on interval graphs to my attention and for helpful discussions on the subject.

References

[1] K. S. Booth and G. S. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity using pq-tree algorithms. *J. Comput. Syst. Sci.*, 13(3):335–379, 1976.

[2] A. Brandstädt, V. B. Le, and J. P. Spinrad. *Graph Classes: a Survey*. Monographs on Discrete Mathematics and Applications. SIAM, 1999.

[3] J. Cai, M. Fürer, and N. Immerman. An optimal lower bound on the number of variables for graph identification. *Combinatorica*, 12(4):389–410, 1992.

[4] R. Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer, 3rd edition, 2006.

[5] H.-D. Ebbinghaus and J. Flum. *Finite model theory*. Springer-Verlag, 2nd edition, 1999.

[6] H.-D. Ebbinghaus, J. Flum, and W. Thomas. *Mathematical Logic*. Springer-Verlag, 2nd edition, 1994.

[7] S. Evdokimov, M. Karpinski, and I. Ponomarenko. On a new high dimensional Weisfeiler-Lehman algorithm. *J. Algebraic Comb.*, 10(1):29–45, 1999.

[8] P. C. Gilmore and A. J. Hoffman. A characterization of comparability graphs and of interval graphs. *Canad. J. Math.*, 16:539–548, 1964.

[9] M. Golumbic. *Algorithmic Graph Theory and Perfect Graphs (Annals of Discrete Mathematics, Vol 57)*. North-Holland Publishing Co., Amsterdam, The Netherlands, The Netherlands, 2004.

[10] E. Grädel, P. Kolaitis, L. Libkin, M. Marx, J. Spencer, M. Vardi, Y. Venema, and S. Weinstein. *Finite Model Theory and Its Applications*. Texts in Theoretical Computer Science. Springer, Berlin, Germany, 2007.

[11] J. Griggs and D. West. Extremal values of the interval number of a graph. *SIAM J. Alg. Discrete Methods*, 1:1–7, 1979.

[12] M. Grohe. Fixed-point logics on planar graphs. In *LICS '08*, pages 6–15. IEEE Computer Society, 1998.

[13] M. Grohe. Definable tree decompositions. In *LICS '08*, pages 406–417. IEEE Computer Society, 2008.

[14] M. Grohe. Fixed-point definability and polynomial time on chordal graphs and line graphs. *arXiv*, 1001.2572, 2010.

[15] M. Grohe and J. Mariño. Definability and descriptive complexity on databases of bounded tree-width. In *ICDT '99*, volume 1540 of *LNCS*, pages 70–82. Springer, 1999.

[16] M. Grohe and O. Verbitsky. Testing graph isomorphism in parallel by playing a game. In *ICALP (1)*, pages 3–14, 2006.

[17] Y. Gurevich and S. Shelah. Fixed-point extensions of first-order logic. In *26th Annual Symposium on Foundations of Computer Science*, pages 346–353, 1985.

[18] M. Habib, R. M. McConnell, C. Paul, and L. Viennot. Lex-BFS and partition refinement. *Theor. Comput. Sci.*, 234(1-2):59–84, 2000.

[19] W.-L. Hsu and T.-H. Ma. Fast and simple algorithms for recognizing chordal comparability graphs and interval graphs. *SIAM J. Comput.*, 28(3):1004–1020, 1999.

[20] N. Immerman. Upper and lower bounds for first-order expressibility. *Journal of Computer and System Sciences*, 25:76–98, 1982.

[21] N. Immerman. *Descriptive Complexity*. Springer, 1999.

[22] S. Kreutzer. Expressive equivalence of least and inflationary fixed-point logic. *Annals of Pure and Applied Logic*, 130(1-3):61–78, 2004.

[23] C. G. Lekkerkerker and J. C. Boland. Representation of a finite graph by a set of intervals on the real line. *Fundamenta Mathematicae*, 51:45–64, 1962.

[24] G. S. Lueker and K. S. Booth. A linear time algorithm for deciding interval graph isomorphism. *J. ACM*, 26(2):183–195, 1979.

[25] R. Möhring. *Graphs and Order*, volume 147 of *NATO ASI Series C, Mathematical and Physical Sciences*, pages 41–102. D. Reidel, 1984.

[26] R. Uehara. Simple geometrical intersection graphs. In *WALCOM*, pages 25–33, 2008.

[27] M. Vardi. The complexity of relational query languages. In *STOC '82*, pages 137–146, 1982.

[28] P. Zhang, E. A. Schon, S. G. Fischer, E. Cayanis, J. Weiss, S. Kistler, and P. E. Bourne. An algorithm based on graph theory for the assembly of contigs in physical mapping of DNA. *Bioinformatics*, 10(3):309–317, 1994.

A Appendix

A.1 Proof of Lemma 2.7

Lemma (Counting equivalence classes)**.** *Suppose $\sim$ is an* FP+C*-definable equivalence relation on k-tuples of V, and let $\varphi(\vec{x})$ be an* FP+C*-formula with $|\vec{x}| = k$. Assume that $\sim$ has at most $|V|$ equivalence classes. Then there is an* FP+C*-counting term giving the number of equivalence classes $[\vec{v}]$ of $\sim$ such that $G \models \varphi[\vec{u}]$ for some $\vec{u} \in [\vec{v}]$.*

Proof. The idea is to construct the sum slicewise for each cardinality of equivalence classes first, which gives us control over the number of classes rather than the number of elements in these classes. Let $\vec{s}, z, a, b$ be number variables. Define the relation $R(\vec{s}, \vec{x})$ to hold if $\vec{x}$ is contained in a $\sim$-equivalence class of size $\vec{s}$ which contains some element making φ true. Using the fixed-point operator, it is then easy to define relation $S(\vec{s}, z) \;:\Leftrightarrow\; z \leq \sum_{\vec{i} \in [\vec{s}]} \#a \;\; \exists b \;\; \Big(a < b \;\wedge\; b \cdot \vec{i} = \#\vec{x}\; R(\vec{i}, \vec{x})\Big)$ for $\vec{s}$ from 1 to $|V|^k$. Then $\#z \;\; S(|V|^k, z)$ is the desired counting term. □

A.2 Proof of Lemma 2.8

Lemma (Lexicographic leader)**.** *Let $\vec{x}$ be a tuple of variables taking values in $V^k \times N_V^\ell$ and let $\vec{y}$ be a tuple of number variables. Suppose $\varphi(\vec{x}, \vec{y})$ is an* FP+C*-formula and $\sim$ is an* FP+C*-definable equivalence relation on $V^k \times N_V^\ell$. Then there is an* FP+C*-formula $\lambda(\vec{x}, \vec{y})$ so that for any $\vec{v} \in V^k \times N_V^\ell$, $\lambda^G[\vec{v}, \cdot]$ is the lexicographic leader among the relations $\big\{\varphi^G[\vec{u}, \cdot] \;\big|\; \vec{u} \sim \vec{v}\big\}$.*

Proof. To start, there is a FO-sentence $\psi(\vec{x}, \vec{y})$ so that $\psi[\vec{u}, \vec{v}]$ holds if and only if $\varphi[\vec{u}, \cdot]$ is lexicographically smaller or equal to $\varphi[\vec{v}, \cdot]$. Now λ is given by

$$\lambda(\vec{x}, \vec{y}) = \exists \vec{z}\,(\vec{x} \sim \vec{z} \,\wedge\, \varphi(\vec{z}, \vec{y}) \,\wedge\, \forall \vec{w}(\vec{w} \sim \vec{z} \,\rightarrow\, \psi(\vec{z}, \vec{w})))$$

□

A.3 Proof of Lemma 2.10

Lemma. *Suppose $\sim$ is an* FP+C*-definable equivalence relation on $V^k \times [|V|]^\ell$ and let $\upsilon(\vec{x}, y)$, $\epsilon(\vec{x}, y, z)$ be* FP+C*-formulas with number variables y, z defining graphs $\big(\upsilon^G[\vec{v}, \cdot], \epsilon^G[\vec{v}, \cdot, \cdot]\big)$ on the numeric sort for each $\vec{v} \in V^k \times [|V|]^\ell$. Furthermore, assume that $\upsilon^G[\vec{v}, \cdot] = \upsilon^G[\vec{v}', \cdot]$ whenever $\vec{v} \sim \vec{v}'$, and that $\sum_{[\vec{v}] \in V/\sim} |\upsilon^G[\vec{v}, \cdot]| \leq |V|$. Then there is an* FP+C*-formula $\omega(y, z)$ defining the lexicographic disjoint union of the lexicographic leaders of $\sim$'s equivalence classes on $\Big[\sum_{[\vec{v}] \in V/\sim} |\upsilon^G[\vec{v}, \cdot]|\Big]$*

Proof. Let $<$ be the strict weak order on $\sim$'s equivalence classes induced by the strict weak order on the classes' respective lexicographic (υ, ϵ)-leader. Using Lemma 2.8, it is easy to define $<$, using elements from $V^k \times [|V|]^\ell$ to identify equivalence classes. Using the fixed point-operator, define ω inductively starting with the $<$-least elements, saving those elements $\vec{v}$ from equivalence classes that have already been considered in a relation R. In each step, find the $<$-least elements L in $V^k \times [|V|]^\ell$ which are not in R, calculate the number n of equivalence classes contained in L, and then expand ω by n copies of $\lambda[\vec{v}, \cdot]$ (which is the same for any $\vec{v} \in L$). □