First-Order Logic with Connectivity Operators

Nicole Schirrmacher
University of Bremen, Germany

Sebastian Siebertz
University of Bremen, Germany

Alexandre Vigny
University of Bremen, Germany

Abstract

First-order logic (FO) can express many algorithmic problems on graphs, such as the independent set and dominating set problem parameterized by solution size. On the other hand, FO cannot express the very simple algorithmic question whether two vertices are connected. We enrich FO with connectivity predicates that are tailored to express algorithmic graph properties that are commonly studied in parameterized algorithmics. By adding the atomic predicates $\mathrm{conn}_k(x, y, z_1, \ldots, z_k)$ that hold true in a graph if there exists a path between (the valuations of) x and y after (the valuations of) $z_1, \ldots, z_k$ have been deleted, we obtain *separator logic* $\mathrm{FO} + \mathrm{conn}$. We show that separator logic can express many interesting problems such as the feedback vertex set problem and elimination distance problems to first-order definable classes. Denote by $\mathrm{FO} + \mathrm{conn}_k$ the fragment of separator logic that is restricted to connectivity predicates with at most $k + 2$ variables (that is, at most k deletions). We show that $\mathrm{FO} + \mathrm{conn}_{k+1}$ is strictly more expressive than $\mathrm{FO} + \mathrm{conn}_k$ for all $k \geq 0$. We then study the limitations of separator logic and prove that it cannot express planarity, and, in particular, not the disjoint paths problem. We obtain the stronger *disjoint-paths logic* $\mathrm{FO} + \mathrm{DP}$ by adding the atomic predicates $\text{disjoint-paths}_k[(x_1, y_1), \ldots, (x_k, y_k)]$ that evaluate to true if there are internally vertex-disjoint paths between (the valuations of) x_i and y_i for all $1 \leq i \leq k$. Disjoint-paths logic can express the disjoint paths problem, the problem of (topological) minor containment, the problem of hitting (topological) minors, and many more. Again we show that the fragments $\mathrm{FO} + \mathrm{DP}_k$ that use predicates for at most k disjoint paths form a strict hierarchy of expressiveness. Finally, we compare the expressive power of the new logics with that of transitive-closure logics and monadic second-order logic.

2012 ACM Subject Classification Theory of computation → Finite Model Theory; Mathematics of computing → Combinatorics

Keywords and phrases First-order logic, graph theory, connectivity

Digital Object Identifier 10.4230/LIPIcs.CSL.2022.34

Funding This paper is a part of the ANR-DFG project *Unifying Theories for Multivariate Algorithms* (UTMA), which has received funding from the German Research Foundation (DFG) with grant agreement No 446200270.

Acknowledgements We thank Mikołaj Bojańczyk for fruitful discussions. He independently studied $\mathrm{FO} + \mathrm{conn}$ and suggested the name *separator logic* for this logic. We also thank Michał Pilipczuk and Szymon Toruńczyk for fruitful discussions.

1 Introduction

Logic provides a very elegant way of formally describing computational problems. Fagin's celebrated result in 1974 [11] established that existential second-order logic captures the complexity class NP. Fagin thereby provided a machine-independent characterization of a complexity class and initiated the field of descriptive complexity theory. Many other complexity classes were later characterized by logics in this theory. Today it remains one of the major open problems whether there exists a logic capturing PTime.

In 1990 Courcelle proved that every graph property definable in monadic second-order logic (MSO) can be decided in linear time on graphs of bounded treewidth [7]. This theorem has a much more algorithmic (rather than a complexity-theoretic) flavor, in the sense that, from a logical description of a problem, it derives an algorithmic approach on how to solve it on certain graph classes. Grohe in his seminal survey coined the term *algorithmic meta-theorem* for such theorems that provide general conditions on a problem and on the input instances that, when satisfied, imply the existence of an efficient algorithm for the problem [17]. Courcelle's theorem for MSO was extended to graph classes with bounded cliquewidth [8] and it is known that these are essentially the most general graph classes on which efficient MSO model-checking [15, 21] is possible. MSO is a powerful logic that can express many important algorithmic properties on graphs. With quantification over edges, we can for example express the existence of a Hamiltonian path, the existence of a fixed minor or topological minor, the disjoint paths problem, and many deletion problems. For a property Π, the task in the Π-deletion problem is to find in a given graph G a minimum-size subset S of $V(G)$ such that the graph $G - S$ obtained from G by removing S has the property Π. Important examples of Π-deletion problems are the feedback vertex set problem, the odd cycle transversal problem, or the problem of hitting all minors or topological minors from a given list $\mathcal{F}$. Also, many elimination distance problems recently studied [5] in parameterized algorithmics can be expressed in MSO. However, as we have seen, this expressiveness comes at the price of algorithmic intractability already on very restricted graph classes. This cannot be a surprise as e.g. the Hamiltonian path problem is NP-complete already on planar graphs of maximum degree 3 [6].

First-order logic (FO) is much weaker than MSO and consequently, the model-checking problem can be solved efficiently on much more general graph classes. FO model-checking is fixed-parameter tractable on a subgraph-closed class $\mathscr{C}$ if and only if $\mathscr{C}$ is nowhere dense [18] and a recent breakthrough result showed that it is fixed-parameter tractable on a class $\mathscr{C}$ of ordered graphs if and only if $\mathscr{C}$ has bounded twin-width [3]. FO is weaker than MSO but it can still express many important problems such as the independent set problem and dominating set problem parameterized by solution size, the Steiner tree problem parameterized by the number of Steiner vertices, and many more problems. On the other hand, first-order logic cannot even express the algorithmically extremely simple problem of whether a graph is connected. Also, the other algorithmic problems mentioned before are not expressible in FO, even though some of them are fixed-parameter tractable on general graphs. For example, we can efficiently test for a fixed minor or topological minor and solve the disjoint paths problem [26]. Many Π-deletion problems are fixed-parameter tractable, see e.g. [9, 14, 25], as well as many elimination distance problems [1, 12].

The fact that first-order logic can only express local properties is classically addressed by adding transitive-closure or fixed-point operators, see e.g. [10, 16, 22]. Unfortunately, this again comes at the price of intractable model-checking for very restricted graph classes. For example, even the model-checking problem for the very restricted monadic transitive-closure logic TC^1 studied by Grohe [17], is AW[$\star$]-hard on planar graphs of maximum degree at most 3 [17, Theorem 7.3]. Also, these logics fall short of being able to express all of the above mentioned algorithmic graph problems studied in recent parameterized algorithmics.

This motivates our present work in which we enrich first-order logic with basic connectivity predicates. The extensions are tailored to express algorithmic graph properties that are studied in recent parameterized algorithmics. We can add the atomic predicate $\mathrm{conn}_0(x, y)$

that evaluates to true on a graph G if (the valuations of) x and y are connected in G. This predicate easily generalizes to directed graphs but for simplicity, we work with undirected graphs only. Of course, with this predicate we can express connectivity of graphs, however, it falls short of expressing other interesting properties, e.g. it cannot express that a graph is acyclic. We hence introduce more general predicates $\mathrm{conn}_k(x, y, z_1, \ldots, z_k)$, parameterized by a number k, that evaluate to true on a graph G if (the valuations of) x and y are connected in G once (the valuations of) $z_1, \ldots, z_k$ have been deleted. The interplay of these predicates with the usual nesting of first-order quantification makes the new logic FO + conn already quite powerful. For example, we can express simple properties such as 2-connectivity by $\forall z \forall x \forall y \big(x \neq z \wedge y \neq z \rightarrow \mathrm{conn}_1(x, y, z)\big)$. We can also express many deletion problems, such as the feedback vertex set problem, and the elimination distance to bounded degree, and more generally, elimination distance to any fixed first-order property.

We also point to the work of Mikołaj Bojańczyk [2], who independently introduced FO + conn and proposed the name *separator logic*. He studied a variant of star-free expressions for graphs and showed that these expressions exactly correspond to separator logic. We follow his suggestion and thank Mikołaj for the discussion on separator logic.

In Section 3 we study the expressive power of separator logic. We give examples on properties expressible with separator logic as well as proofs that certain properties, such as planarity and in particular the disjoint paths problem, are not expressible in separator logic. We show that $(k + 2)$-connectivity of a graph cannot be expressed with only conn_k predicates and conclude that the restricted use of these predicates induces a natural hierarchy of expressiveness.

Using the notion of *block decompositions* together with known model-checking results, one can show that model-checking for formulas using only conn_1 predicates is fixed-parameter tractable on nowhere dense classes of graphs. Hence, we can evaluate very simple connectivity queries in formulas without an increase in the complexity of the model-checking problem on subgraph-closed graph classes. On the other hand, when we allow conn_2 predicates, there are some simple graph classes that do not exclude a topological minor, and on which model-checking becomes AW[$\star$]-hard. In this paper, we do not go into the details of model-checking, but in a companion paper [24], we prove that in fact model-checking for FO + conn is fixed-parameter tractable on graph classes that exclude a topological minor.

The fact that planarity and the disjoint paths problem cannot be expressed in separator logic motivates us to define an even stronger logic that can express these properties. The atomic predicate $\text{disjoint-paths}_k[(x_1, y_1), \ldots, (x_k, y_k)]$ evaluates to true if and only if there are internally vertex-disjoint paths between (the valuations of) x_i and y_i for all $1 \leq i \leq k$. Connectivity of x and y can be tested by $\text{disjoint-paths}_1[(x, y)]$. More generally, the so obtained *disjoint-paths logic* FO + DP strictly extends separator logic. With this more powerful logic, we can test if a graph contains a fixed minor or topological minor, and in particular, test for planarity. In combination with first-order quantification, we can also express many Π-deletion problems such as the problem of hitting all minors or topological minors from a given list $\mathcal{F}$. On the other hand, we cannot express the odd cycle transversal problem, as we cannot even express bipartiteness of a graph. We study the expressive power of FO + DP in Section 4. Among other results, we prove that again an increase in the number of disjoint paths in the predicates leads to an increase in expressive power.

Note that while it would be desirable to be able to express bipartiteness, which is equivalent to 2-colorability, it is not desirable to express general colorability problems, as we aim for logics that are tractable on planar graphs and beyond, while the 3-colorability problem is NP-complete on planar graphs. This example shows again that it is a delicate balance between expressiveness and tractability and it will be a challenging and highly interesting problem in future work to find the right set of predicates to express even more algorithmic graph properties while at the same time having tractable model-checking. Until now the complexity of the model-checking problem for FO + DP has remained elusive and will be a very interesting problem in future work.

We conclude the paper in Section 5 with a comparison between the newly introduced logics and more established ones, like MSO and transitive-closure logics.

2 Preliminaries

Graphs. In this paper we deal with finite and simple undirected graphs. Let G be a graph. We write $V(G)$ for the vertex set of G and $E(G)$ for its edge set. For a set $X \subseteq V(G)$ we write $G[X]$ for the subgraph of G induced by X and $G - X$ for the subgraph induced by $V(G) \setminus X$. For a singleton set $\{v\}$ we write $G - v$ instead of $G - \{v\}$. A *path* P in G is a subgraph on distinct vertices $v_1, \ldots, v_t$ with $\{v_i, v_{i+1}\} \in E(P)$ for all $1 \leq i < t$ and a path P is said to *connect* its endpoints v_1 and v_t. Two paths are *internally vertex-disjoint* if and only if every vertex that appears in both paths is an end point of both paths. The graph G is *connected* if every two of its vertices are connected by a path. It is *k-connected* if G has more than k vertices and $G - X$ is connected for every subset $X \subseteq V(G)$ of size strictly smaller than k. A *cycle* C in G is a subgraph on distinct vertices $v_1, \ldots, v_t$, $t \geq 3$, with $\{v_t, v_1\} \in E(C)$ and $\{v_i, v_{i+1}\} \in E(C)$ for all $1 \leq i < t$. An acyclic graph is a *forest* and a connected acyclic graph is a *tree*.

A graph H is a *minor* of G, denoted $H \preccurlyeq G$, if for all $v \in V(H)$ there are pairwise vertex-disjoint connected subgraphs G_v of G such that whenever $\{u, v\} \in E(H)$, then there are $x \in V(G_u)$ and $y \in V(G_v)$ with $\{x, y\} \in E(G)$. The graph H is a *topological minor* of G, denoted $H \preccurlyeq^{top} G$, if for all $v \in V(H)$ there is a distinct vertex x_v in G and for all $\{u, v\} \in E(H)$ there are internally vertex-disjoint paths P_{uv} in G with endpoints x_u and x_v. A graph is *planar* if and only if it does not contain K_5, the complete graph on 5 vertices, and $K_{3,3}$, the complete bipartite graph with two partitions of size 3, as a minor.

Logic. In this work we deal with structures over purely relational *signatures*. A (purely relational) signature is a collection of relation symbols, each with an associated arity. Let σ be a signature. A *σ-structure* $\mathfrak{A}$ consists of a non-empty set A, the universe of $\mathfrak{A}$, together with an interpretation of each k-ary relation symbol $R \in \sigma$ as a k-ary relation $R^{\mathfrak{A}} \subseteq A^k$. For a subset $X \subseteq A$ we write $\mathfrak{A}[X]$ for the substructure induced by X. A *partial isomorphism* between σ-structures $\mathfrak{A}$ and $\mathfrak{B}$ is an isomorphism between $\mathfrak{A}[X]$ and $\mathfrak{B}[Y]$ for some subset $X \subseteq A$ of the universe A of $\mathfrak{A}$ and some subset $Y \subseteq B$ of the universe B of $\mathfrak{B}$.

We assume an infinite supply $\textsc{Var}$ of variables. First-order formulas are built from the atomic formulas $x = y$, where x and y are variables, and $R(x_1, \ldots, x_k)$, where $R \in \sigma$ is a k-ary relation symbol and $x_1, \ldots, x_k$ are variables, by closing under the Boolean connectives $\neg$, $\wedge$ and $\vee$, and by existential and universal quantification $\exists x$ and $\forall x$. A variable x not in the scope of a quantifier is a *free variable*. A formula without free variables is a *sentence*. The *quantifier rank* $\mathrm{qr}(\varphi)$ of a formula φ is the maximum nesting depth of quantifiers in φ. We write $\mathrm{FO}_\sigma[q]$ for the set of all FO σ-formulas of quantifier rank at most q, or simply $\mathrm{FO}[q]$ if σ is clear from the context. A formula without quantifiers is called *quantifier-free*.

If $\mathfrak{A}$ is a σ-structure with universe A, then an *assignment* of the variables in $\mathfrak{A}$ is a mapping $\bar{a} : \textsc{Var} \to A$. We use the standard notation $(\mathfrak{A}, \bar{a}) \models \varphi(\bar{x})$ or $\mathfrak{A} \models \varphi(\bar{a})$ to indicate that φ is satisfied in $\mathfrak{A}$ when the free variables $\bar{x}$ of φ have been assigned by $\bar{a}$. We refer e.g. to the textbook [22] for more background on first-order logic.

3 Separator logic

In this section, we study the expressive power of separator logic FO + conn. Formally, we assume that σ is a signature that does not contain any of the relation symbols conn_k for all $k \geq 0$, and that it does contain a binary relation symbol E, representing an edge relation. We assume that E is always interpreted as an irreflexive and symmetric relation and connectivity will always refer to this relation. We let $\sigma + \mathrm{conn} \coloneqq \sigma \cup \{\mathrm{conn}_k : k \geq 0\}$, where each conn_k is a $(k + 2)$-ary relation symbol.

▶ **Definition 3.1.** *The formulas of* $(\mathrm{FO} + \mathrm{conn})[\sigma]$ *are the formulas of* $\mathrm{FO}[\sigma + \mathrm{conn}]$*. We usually simply write* FO + conn*, when* σ *is understood from the context.*

For a σ*-structure* $\mathfrak{A}$*, an assignment* $\bar{a}$ *and an* FO + conn *formula* $\varphi(\bar{x})$*, we define the satisfaction relation* $(\mathfrak{A}, \bar{a}) \models \varphi(\bar{x})$ *as for first-order logic, where an atomic predicate* $\mathrm{conn}_k(x, y, z_1, \ldots, z_k)$ *is evaluated as follows. Assume that the universe of* $\mathfrak{A}$ *is* A *and let* $G = (A, E^{\mathfrak{A}})$ *be the graph on vertex set* A *and edge set* $E^{\mathfrak{A}}$*. Then* $(\mathfrak{A}, \bar{a})$ *models* $\mathrm{conn}_k(x, y, z_1, \ldots, z_k)$ *if and only if* $\bar{a}(x)$ *and* $\bar{a}(y)$ *are connected in* $G - \{\bar{a}(z_1), \ldots, \bar{a}(z_k)\}$*.*

Note in particular that if $\bar{a}(x) = \bar{a}(z_i)$ or $\bar{a}(y) = \bar{a}(z_i)$ for some $i \leq k$, then $(\mathfrak{A}, \bar{a}) \not\models \mathrm{conn}_k(x, y, z_1, \ldots, z_k)$.

We write $\mathrm{FO} + \mathrm{conn}_k$ for the fragment of FO + conn that uses only conn_ℓ predicates for $\ell \leq k$. The quantifier rank of an FO + conn formula is defined as for plain first-order logic. For structures $\mathfrak{A}$ with universe A and $\bar{a} \in A^m$ and $\mathfrak{B}$ with universe B and $\bar{b} \in B^m$, we write $(\mathfrak{A}, \bar{a}) \equiv_{\mathrm{conn}} (\mathfrak{B}, \bar{b})$ if $(\mathfrak{A}, \bar{a})$ and $(\mathfrak{B}, \bar{b})$ satisfy the same FO + conn formulas, that is, for all $\varphi(\bar{x})$ we have $\mathfrak{A} \models \varphi(\bar{a}) \Leftrightarrow \mathfrak{B} \models \varphi(\bar{b})$. Similarly, we write $(\mathfrak{A}, \bar{a}) \equiv_{\mathrm{conn}_k} (\mathfrak{B}, \bar{b})$ and $(\mathfrak{A}, \bar{a}) \equiv_{\mathrm{conn}_{k,q}} (\mathfrak{B}, \bar{b})$ if $(\mathfrak{A}, \bar{a})$ and $(\mathfrak{B}, \bar{b})$ satisfy the same $\mathrm{FO} + \mathrm{conn}_k$ formulas and the same $\mathrm{FO} + \mathrm{conn}_k$ formulas of quantifier rank at most q, respectively.

3.1 Expressive power of separator logic

We now give examples of properties that are expressible with separator logic.

▶ **Example 3.2.** Connectivity is expressible in $\mathrm{FO} + \mathrm{conn}_0$ by the formula

$$\forall x \forall y \big(\mathrm{conn}_0(x, y)\big).$$

More generally, for every non-negative integer k, $(k+1)$-connectivity can be expressed by the formula

$$\forall x \forall y \forall z_1 \ldots \forall z_k \Big(\bigwedge_{1 \leq i \leq k} (x \neq z_i \wedge y \neq z_i) \to \mathrm{conn}_k(x, y, z_1, \ldots, z_k)\Big).$$

▶ **Example 3.3.** We can express that there exists a cycle by

$$\exists x \exists y \big(E(x,y) \wedge \exists z \big(\mathrm{conn}_1(z,x,y) \wedge \mathrm{conn}_1(z,y,x)\big)\big),$$

hence, that a graph is acyclic by the negation of that formula. We write $\psi_{acyclic}$ for that formula. We can express that a graph is a tree by stating that it is connected and acyclic.

We can conveniently express deletion problems by relativizing formulas as follows. For a formula φ that does not contain z as a free variable write $\mathrm{del}(z)[\varphi]$ for the formula obtained from φ by recursively replacing every subformula $\exists x\psi$ by $\exists x(x \neq z \wedge \psi)$, every subformula $\forall x\psi$ by $\forall x(x \neq z \rightarrow \psi)$ and every atomic formula $\mathrm{conn}_k(x,y,z_1,\ldots,z_k)$ by $\mathrm{conn}_{k+1}(x,y,z_1,\ldots,z_k,z)$. Then $(\mathfrak{A},\bar{a}) \models \mathrm{del}(z)[\varphi]$ if and only if $(\mathfrak{A} - \bar{a}(z),\bar{a}) \models \varphi$, where $\mathfrak{A} - \bar{a}(z)$ denotes the substructure induced on the universe of $\mathfrak{A}$ without $\bar{a}(z)$.

▶ **Example 3.4.** We can state the existence of a feedback vertex set of size k by

$$\exists z_1 \mathrm{del}(z_1)[\cdots[\exists z_k \mathrm{del}(z_k)[\psi_{acyclic}]\ldots].$$

We can of course use the same principle to express any Π-deletion problem that is $\mathrm{FO}+\mathrm{conn}$ expressible.

We can also, much more generally, express many elimination distance problems.

▶ **Example 3.5.** The *elimination distance* to a class $\mathscr{C}$ of graphs measures the number of recursive deletions of vertices needed for a graph G to become a member of $\mathscr{C}$. More precisely, a graph G has elimination distance 0 to $\mathscr{C}$ if $G \in \mathscr{C}$, and otherwise elimination distance at most $k+1$ if in every connected component of G we can delete a vertex such that the resulting graph has elimination distance at most k to $\mathscr{C}$. Elimination distance was introduced by Bulian and Dawar [5] in their study of the parameterized complexity of the graph isomorphism problem and has recently obtained much attention in the literature, see e.g. [1, 4, 13, 19, 20, 23].

Again, we define auxiliary notation. We write $\mathrm{comp}(x)$ for the connected component of (the valuation of) x. For a formula φ we write $\varphi^{[\mathrm{comp}(x)]}$ for the formula obtained from φ by recursively replacing all subformulas $\exists y\psi$ by $\exists y(\mathrm{conn}_0(x,y) \wedge \psi)$ and all subformulas $\forall y\psi$ by $\forall y(\mathrm{conn}_0(x,y) \rightarrow \psi)$. Then $(\mathfrak{A},\bar{a}) \models \varphi^{[\mathrm{comp}(x)]}$ if and only if $(\mathfrak{A}[\mathrm{comp}(\bar{a}(x))],\bar{a}) \models \varphi$, where $\mathfrak{A}[\mathrm{comp}(\bar{a}(x))]$ denotes the substructure induced on the connected component of $\bar{a}(x)$.

Now assume $\mathscr{C}$ is a first-order definable class, say defined by a formula $\psi_{\mathscr{C}}$. Then elimination distance 0 to $\mathscr{C}$ is defined by $\mathrm{ed}_0 = \psi_{\mathscr{C}}$. If ed_k has been defined, then we can express elimination distance $k+1$ to $\mathscr{C}$ by the formula

$$\mathrm{ed}_{k+1} \coloneqq \mathrm{ed}_k \vee \forall x \big(\exists y\ \mathrm{del}(y)[\mathrm{ed}_k]\big)^{[\mathrm{comp}(x)]}.$$

Our final example concerns the expressive power of separator logic on finite words and finite trees. By the classical result of Büchi, a language on words is regular if and only if it is definable in MSO. Here, words are represented as finite structures over the vocabulary of the successor relation and unary predicates representing the letters of the alphabet. When considering first-order logic on strings, it makes a big difference whether one considers word structures over the successor relation or over its transitive closure, the order relation. Languages definable by FO over the order relation are exactly the star-free languages (see e.g. [22, Theorem 7.26]), while languages definable by FO over the successor relation are exactly the locally threshold testable languages [27, Theorem 4.8]. Similarly, MSO on trees can define exactly the tree regular languages (defined via tree automata, see [22, Theorem 7.30]), while FO can only define a proper subclass of the regular tree languages when the ancestor-descendant or even only the parent-child relation is present. This background was also the motivation of Bojańczyk, who studied a variant of star-free expressions for graphs and showed that these expressions exactly correspond to separator logic [2]. In our example, we show that separator logic on rooted trees has exactly the same expressive power as first-order logic in the presence of the ancestor-descendant relation. Let us write FO[<] for the latter logic. On the other hand, we treat a rooted tree as a graph-theoretic tree with an additional unary predicate marking the root. In the degenerate case, we treat a word as a path, where one of the endpoints is marked by a unary predicate as the smallest vertex (the beginning of the word).

▶ **Example 3.6.** On rooted trees (and similarly on words) $\mathrm{FO} + \mathrm{conn}$ collapses to $\mathrm{FO} + \mathrm{conn}_1$ and has exactly the same expressive power as FO[<] over trees with the ancestor-descendant relation. We show first that $\mathrm{conn}_k(x, y, z_1, \ldots, z_k)$ can be expressed in FO[<]. For this, we need to ensure that x and y are not equal to any z_i and that no z_i lies on the unique path between x and y in the tree. We can define the vertices on the unique path between x and y by first defining the least common ancestor of x and y by the formula $\mathrm{lca}(x, y, z) = z \leq x \wedge z \leq y \wedge \neg\exists z'(z < z' \wedge z' \leq x \wedge z' \leq y)$. If z is the least common ancestor of x and y, it remains to state that none of the z_i lies either between x and z or between y and z, which is done by the formula $\exists z\big(\mathrm{lca}(x, y, z) \wedge \bigwedge_{1 \leq i \leq k} \neg(z \leq z_i \leq x \vee z \leq z_i \leq y)\big)$.

Conversely, we show that we can define with $\mathrm{FO} + \mathrm{conn}_1$ the ancestor-descendant relation in rooted trees. Assume the root is marked by the unary symbol R. Then $x < y$ is equivalent to $\exists r\big(R(r) \wedge \mathrm{conn}_1(x, r, y) \wedge \neg\mathrm{conn}_1(y, r, x)\big)$.

3.2 The limits of separator logic

We now study the limits of separator logic and show that planarity cannot be expressed in $\mathrm{FO} + \mathrm{conn}$. Slightly abusing notation let us also write $\mathrm{FO} + \mathrm{conn}_k$ for the properties that are expressible in $\mathrm{FO} + \mathrm{conn}_k$. We show that there is a strict hierarchy of expressiveness: $\mathrm{FO} + \mathrm{conn}_0 \subsetneq \mathrm{FO} + \mathrm{conn}_1 \subsetneq \mathrm{FO} + \mathrm{conn}_2 \subsetneq \ldots$ These results are based on an adaptation of the standard Ehrenfeucht-Fraïssé game (EF game), which is commonly used in the study of the expressive power of first-order logic.

Ehrenfeucht-Fraïssé Games. The Ehrenfeucht-Fraïssé game is played by two players called *Spoiler* and *Duplicator*. Given two structures $\mathfrak{A}$ and $\mathfrak{B}$, Spoiler's aim is to show that the structures can be distinguished by first-order logic (with formulas of a given quantifier rank), while Duplicator wants to prove the opposite. The q-round EF game proceeds in q rounds, where each round consists of the following two steps.

1. Spoiler picks an element $a \in \mathfrak{A}$ or an element $b \in \mathfrak{B}$.
2. Duplicator responds by picking an element of the other structure, that is, she picks a $b \in \mathfrak{B}$ if Spoiler chose $a \in \mathfrak{A}$, and she picks an $a \in \mathfrak{A}$ if Spoiler chose $b \in \mathfrak{B}$.

After q rounds, the game stops. Assume the players have chosen $\bar{a} = a_1, \ldots, a_q$ and $\bar{b} = b_1, \ldots, b_q$. Then Duplicator *wins* if the mapping $a_i \mapsto b_i$ for all $1 \leq i \leq q$ is a partial isomorphism of $\mathfrak{A}$ and $\mathfrak{B}$. We write for short $\bar{a} \mapsto \bar{b}$ for this mapping. Otherwise, Spoiler wins. We say that Duplicator *wins the q-round EF game* on $\mathfrak{A}$ and $\mathfrak{B}$ if she can force a win no matter how Spoiler plays. We then write $\mathfrak{A} \simeq_q \mathfrak{B}$.

▶ **Theorem 3.7** (Ehrenfeucht-Fraïssé, see e.g. [22, Theorem 3.18]). *Let $\mathfrak{A}$ and $\mathfrak{B}$ be two σ-structures where σ is purely relational. Then $\mathfrak{A} \equiv_q \mathfrak{B}$ if and only if $\mathfrak{A} \simeq_q \mathfrak{B}$.*

The EF game for FO naturally extends to separator logic. The *($\mathrm{conn}_{k,q}$)-game* is played just as the q-round EF game, but the winning condition is changed as follows. If in q rounds the players have chosen $\bar{a} = a_1, \dots, a_q$ and $\bar{b} = b_1, \dots, b_q$, then Duplicator wins if

1. the mapping $\bar{a} \mapsto \bar{b}$ is a partial isomorphism of $\mathfrak{A}$ and $\mathfrak{B}$, and
2. for every $\ell \le k$ and every sequence $(i_1, \dots, i_{\ell+2})$ of numbers in $\{1, \dots, q\}$ we have

$$\mathfrak{A} \models \mathrm{conn}_\ell(a_{i_1}, \dots, a_{i_{\ell+2}}) \iff \mathfrak{B} \models \mathrm{conn}_\ell(b_{i_1}, \dots, b_{i_{\ell+2}}).$$

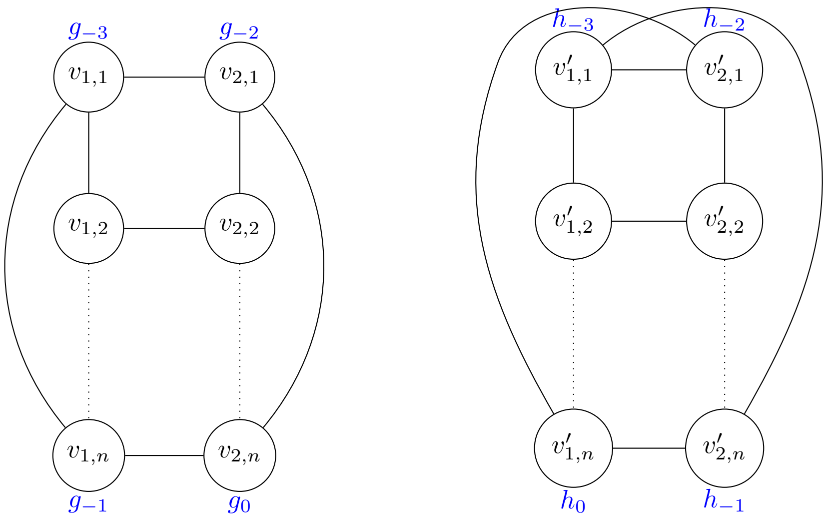
Otherwise, Spoiler wins. We say that Duplicator wins the ($\mathrm{conn}_{k,q}$)-game on $\mathfrak{A}$ and $\mathfrak{B}$ if she can force a win no matter how Spoiler plays. We then write $\mathfrak{A} \simeq_{\mathrm{conn}_{k,q}} \mathfrak{B}$.

By following the lines of the proof of the classical Ehrenfeucht-Fraïssé Theorem we can prove the following theorem.

▶ **Theorem 3.8.** *Let $\mathfrak{A}$ and $\mathfrak{B}$ be two σ-structures where σ is purely rational (and contains a binary relation symbol E that is interpreted on both structures as an irreflexive and symmetric relation). Then $\mathfrak{A} \equiv_{\mathrm{conn}_{k,q}} \mathfrak{B}$ if and only if $\mathfrak{A} \simeq_{\mathrm{conn}_{k,q}} \mathfrak{B}$.*

The next theorem exemplifies the use of the ($\mathrm{conn}_{k,q}$)-game.

▶ **Theorem 3.9.** *Planarity is not expressible in* $\mathrm{FO} + \mathrm{conn}$.



(a) G_q **(b)** H_q

Figure 1 Planarity is not expressible in $\mathrm{FO} + \mathrm{conn}$.

Proof. Assume planarity is expressible by a sentence φ of $\mathrm{FO} + \mathrm{conn}_k$ of quantifier rank q. Without loss of generality, we may assume that $k \le q$, as otherwise, we have repetitions in the conn_k predicates that can be avoided by using conn_ℓ predicates for $\ell < k$. Let G_q and H_q be defined as shown in Figure 1, where $n = 2^{q+1}$. Then, G_q is planar but H_q embeds only in a surface of genus one (into the Möbius strip, which cannot be embedded into the plane). We show that $G_q \simeq_{\mathrm{conn}_{k,q}} H_q$, contradicting the assumption that φ must distinguish G_q and H_q. In fact, we prove an even stronger statement by giving Spoiler four free moves $g_{-3} = v_{1,1}$, $g_{-2} = v_{2,1}$, $g_{-1} = v_{1,n}$ and $g_0 = v_{2,n}$ in G_q and forcing Duplicator to respond with the vertices $h_{-3} = v'_{1,1}$, $h_{-2} = v'_{2,1}$, $h_{-1} = v'_{2,n}$ and $h_0 = v'_{1,n}$ in H_q. Note the twist in the last two vertices. These extra moves are helpful to define Duplicator's winning strategy.

We define the x-distance of two nodes $v_{i,j}$ and $v_{k,\ell}$ as $d_x(v_{i,j}, v_{k,\ell}) = |i - k|$ and the y-distance as $d_y(v_{i,j}, v_{k,\ell}) = |j - \ell|$. Note that the y-distance is not the distance in the graphs, e.g. $d_y(g_{-3}, g_{-1}) = 2^{q+1} - 1$, even though g_{-3} and g_{-1} are adjacent in G_q.

Assume now that the first i moves have been made in the game and the players have selected the vertices $\bar{g} = (g_{-3}, \ldots, g_0, g_1, \ldots, g_i)$ in G_q (where $g_1, \ldots, g_i$ were freely chosen by the players), and $\bar{h} = (h_{-3}, \ldots, h_0, h_1, \ldots, h_i)$ in H_q (where $h_1, \ldots, h_i$ were freely chosen by the players). We prove by induction that Duplicator can play in such a way that after round i of the $(\text{conn}_{k,q})$-game the following conditions hold for all $-3 \leq j, \ell \leq i$:

1. if $g_j = v_{x,y}$, then $h_j = v'_{x',y}$, that is, corresponding pebbles are in the same row, and in particular $d_y(g_j, g_\ell) = d_y(h_j, h_\ell)$, and
2. if $d_y(g_j, g_\ell) \leq 2^{q-i}$, then $d_x(g_j, g_\ell) = d_x(h_j, h_\ell)$.

These conditions together with the first four extra moves imply that the mapping $\bar{g} \mapsto \bar{h}$ is a partial isomorphism of G_q and H_q. Let us show that also for every $0 \leq \ell \leq k$ and every sequence $(i_1, \ldots, i_{\ell+2})$ of numbers in $\{-3, \ldots, i\}$ we have $G_q \models \text{conn}_\ell(g_{i_1}, \ldots, g_{i_{\ell+2}})$ if and only if $H_q \models \text{conn}_\ell(h_{i_1}, \ldots, h_{i_{\ell+2}})$. Assume $G_q \models \text{conn}_\ell(g_{i_1}, \ldots, g_{i_{\ell+2}})$, that is, g_{i_1} and g_{i_2} are connected after the deletion of $g_{i_3}, \ldots, g_{i_{\ell+2}}$, say by a path $P = v_{x_1,y_1} \ldots v_{x_m,y_m}$, where $v_{x_1,y_1} = g_{i_1}$ and $v_{x_m,y_m} = g_{i_2}$. Then there are no $g_{i_{j_1}} = v_{x,y}$ and $g_{i_{j_2}} = v_{x',y'}$ (for $j_1, j_2 \geq 3$) with $y = y' = y_i$ and $x \neq x'$ for some $2 \leq i \leq m-1$ (this would block a row along which the path goes, which is not possible) and no $g_{i_{j_1}} = v_{x,y}$ and $g_{i_{j_2}} = v_{x',y'}$ (for $j_1, j_2 \geq 3$) with $y_i = y = y' - 1 = y_{i+1} - 1$ and $x \neq x'$ for some $2 \leq i \leq m-1$ (this would block a "diagonal" of which the path contains at least one vertex, which is not possible). By the first condition of the invariant there are no $h_{i_{j_1}} = v_{x,y}$ and $h_{i_{j_2}} = v_{x',y'}$ (for $j_1, j_2 \geq 3$) with $y = y' = y_i$ and $x \neq x'$ for some $2 \leq i \leq m-1$ and by the second condition of the invariant there are no $h_{i_{j_1}} = v_{x,y}$ and $h_{i_{j_2}} = v_{x',y'}$ (for $j_1, j_2 \geq 3$) with $y_i = y = y' - 1 = y_{i+1} - 1$ and $x \neq x'$ for some $2 \leq i \leq m-1$. Now, if $P' = v'_{x_1,y_1} \ldots v'_{x_m,y_m}$ is not a path from h_{i_1} to h_{i_2} after the deletion of $h_{i_3}, \ldots, g_{i_{\ell+2}}$, it is possible to reroute the path by switching the row appropriately, as the h_{i_j} never block a complete row or a diagonal, as shown above. The case $H_q \models \text{conn}_\ell(h_{i_1}, \ldots, h_{i_{\ell+2}})$ is symmetrical.

We now show that Duplicator can maintain this invariant throughout the game. For the initial configuration $i = 0$, the conditions are obviously fulfilled for $-3 \leq j, \ell \leq 0$. Corresponding pebbles are in the same row and note that $d_y(g_j, g_\ell) = 2^{q+1} - 1$, for $j \in \{-3, -2\}$ and $\ell \in \{-1, 0\}$ and analogously for h_j and h_ℓ.

For the induction step, suppose that the conditions are fulfilled so far and that Spoiler is making his $(i+1)$-move in G_q (the case of H_q is symmetrical). We may assume that Spoiler does not choose a vertex that was chosen before, say Spoiler picks $g_{i+1} = v_{\_,a}$. Duplicator must choose $h_{i+1} = v'_{\_,a}$ with the same y-coordinate. We have to make sure that she can choose the vertex with that y-coordinate satisfying the second condition. Let $g_j = v_{\_,b}$ and $g_\ell = v_{\_,c}$ with $-3 \leq j, \ell \leq i$ be such that $b \leq a \leq c$ and there is no other $g_k = v_{\_,d}$ with $b < d < c$. Intuitively, g_j is the lowest pebble that was placed above (or in the same row as) g_{i+1}, while g_k is the highest pebble that was placed below (or in the same row as) g_{i+1}. There are two cases:

1. $d_y(g_j, g_\ell) \leq 2^{q-i}$: Then by hypothesis, $d_x(h_j, h_\ell) = d_x(g_j, g_\ell)$ and $d_y(h_j, h_\ell) = d_x(g_j, g_\ell)$. Here, Duplicator chooses the unique $h_{i+1} = v'_{\_,a}$ such that $d_x(h_j, h_{i+1}) = d_x(g_j, g_{i+1})$, and we have $d_x(h_\ell, h_{i+1}) = d_x(g_\ell, g_{i+1})$.
2. $d_y(g_j, g_\ell) > 2^{q-i}$: Then $d_y(h_j, h_\ell) > 2^{q-i}$ and there are three possibilities:
   - $d_y(g_j, g_{i+1}) \leq 2^{q-(i+1)}$: Then $d_y(g_\ell, g_{i+1}) > 2^{q-(i+1)}$, and Duplicator chooses $h_{i+1} = v'_{\_,a}$ such that $d_x(h_j, h_{i+1}) = d_x(g_j, g_{i+1})$. Hence, $d_y(h_\ell, h_{i+1}) > 2^{q-(i+1)}$.

- $d_y(g_\ell, g_{i+1}) \leq 2^{q-(i+1)}$: Then $d_y(g_j, g_{i+1}) > 2^{q-(i+1)}$. Similarly to the previous case, Duplicator chooses $h_{i+1} = v'_{\_,a}$ such that $d_x(h_\ell, h_{i+1}) = d_x(g_\ell, g_{i+1})$. Consequently, $d_y(h_j, h_{i+1}) > 2^{q-(i+1)}$.
- $d_y(g_j, g_{i+1}) > 2^{q-(i+1)}$ and $d_y(g_\ell, g_{i+1}) > 2^{q-(i+1)}$: Here, Duplicator can choose $h_{i+1} = v'_{1,a}$ or $h_{i+1} = v'_{2,a}$ as she wants. We get that $d_y(h_j, h_{i+1}) \geq 2^{q-(i+1)}$ and $d_y(h_\ell, h_{i+1}) \geq 2^{q-(i+1)}$.

Thus, in all cases, the conditions are fulfilled and Duplicator wins the $(\text{conn}_{k,q})$-game on G_q and H_q. Hence, planarity is not definable in $\text{FO} + \text{conn}$. ◀

As a graph is planar if and only if it excludes K_5 and $K_{3,3}$ as (topological) minors and we will show that this can be expressed using disjoint paths predicates, we conclude that the disjoint paths predicate cannot be expressed with $\text{FO} + \text{conn}$.

▶ **Corollary 3.10.** *The disjoint paths problem cannot be expressed in* $\text{FO} + \text{conn}$*.*

The proof of the next theorem is deferred to the next section, as it is a consequence of the fact that the even stronger logic $\text{FO} + \text{DP}$ cannot express bipartiteness (Theorem 4.7).

▶ **Theorem 3.11.** *Bipartiteness cannot be expressed in* $\text{FO} + \text{conn}$*.*

Finally, we show that the $\text{FO}+\text{conn}_k$ hierarchy is strict by proving that $(k+2)$-connectivity cannot be expressed by $\text{FO}+\text{conn}_k$. On the other hand, $(k+2)$-connectivity can be expressed by $\text{FO} + \text{conn}_{k+1}$ (Example 3.2).

▶ **Theorem 3.12.** $(k + 2)$*-connectivity cannot be expressed by* $\text{FO} + \text{conn}_k$*. In particular, the* $\text{FO} + \text{conn}_k$ *hierarchy is strict, that is,* $\text{FO} + \text{conn}_0 \subsetneq \text{FO} + \text{conn}_1 \subsetneq \dots$

Proof. Let k be an integer. For every integer q, we choose two graphs G_q and H_q such that:
- G_q is connected,
- H_q is not connected, and
- $G_q \simeq_q H_q$.

This is possible, as connectivity is not first-order definable and $\simeq_q$ has only finitely many equivalence classes.

Then, we define the graph G_q^k (resp. H_q^k) as the disjoint union of G_q (resp. H_q) and K_{k+1}, a clique of size $k + 1$, and connect the vertices of the clique with all vertices of G_q (resp. H_q), that is, we add the additional edges such that $(x, y) \in E(G_q^k)$ (resp. $(x, y) \in E(H_q^k)$) if $x \in G_q$ (resp. $x \in H_q$) and $y \in K_{k+1}$. Obviously, G_q^k is $(k + 2)$-connected (the deletion of any $k + 1$ vertices cannot disconnect G_q^k), while H_q^k is not $(k + 2)$-connected (the deletion of the copy of K_{k+1} disconnects H_q^k).

The same argument shows that every $\text{conn}_k(x, y, z_1, \dots, z_k)$ can be expressed by an atomic plain first-order formula: in both graphs (the valuations of) x and y are not connected after the deletion of (the valuations of) $z_1, \dots, z_k$ if and only if x or y is equal to one of the z_i. Hence, to prove $G_q^k \simeq_{\text{conn}_{k,q}} H_q^k$ it suffices to prove $G_q^k \simeq_q H_q^k$, and this finishes the proof.

▷ Claim 3.13. For all integers q, k we have $G_q^k \simeq_q H_q^k$.

Proof. The following is obviously a winning strategy for Duplicator in the q-round EF game on G_q^k and H_q^k. If Spoiler plays a pebble in the subgraph G_q or H_q, Duplicator can respond by a pebble in the subgraph H_q or G_q according to the winning strategy of Duplicator in the EF game on G_q and H_q. Otherwise, if Spoiler picks a pebble in the subgraph K_{k+1} of G_q^k or H_q^k, Duplicator can respond by a pebble in the subgraph K_{k+1} of the other graph H_q^k or G_q^k. ◁

This concludes the proof of Theorem 3.12. ◀

4 Disjoint-paths logic

In this section, we study the expressive power of disjoint-paths logic $\mathrm{FO}+\mathrm{DP}$. We again fix a signature σ that does not contain the symbol disjoint-paths$_k$ for any $k \geq 1$ and that does contain a binary (edge) relation symbol E. The disjoint paths predicates will always refer to this relation. We let $\sigma + \text{disjoint-paths} := \sigma \cup \{\text{disjoint-paths}_k : k \geq 1\}$, where each disjoint-paths$_k$ is a $2k$-ary relation symbol.

▶ **Definition 4.1.** *The formulas of* $(\mathrm{FO}+\mathrm{DP})[\sigma]$ *are the formulas of* $\mathrm{FO}[\sigma + \text{disjoint-paths}]$. *We usually simply write* $\mathrm{FO}+\mathrm{DP}$, *when* σ *is understood from the context.*

For a σ*-structure* $\mathfrak{A}$, *an assignment* $\bar{a}$ *and an* $\mathrm{FO}+\mathrm{DP}$ *formula* $\varphi(\bar{x})$, *we define the satisfaction relation* $(\mathfrak{A}, \bar{a}) \models \varphi(\bar{x})$ *as for first-order logic, where an atomic predicate* $\text{disjoint-paths}_k[(x_1, y_1), \ldots (x_k, y_k)]$ *is evaluated as follows. Assume that the universe of* $\mathfrak{A}$ *is* A *and let* $G = (A, E^{\mathfrak{A}})$ *be the graph on vertex set* A *and edge set* $E^{\mathfrak{A}}$. *Then* $(\mathfrak{A}, \bar{a})$ *models* $\text{disjoint-paths}_k[(x_1, y_1), \ldots, (x_k, y_k)]$ *if and only if in* G *there exist* k *internally vertex-disjoint paths* $P_1, \ldots, P_k$, *where* P_i *connects* $\bar{a}(x_i)$ *and* $\bar{a}(y_i)$.

As previously mentioned, it is natural to consider these predicates for both undirected and directed graphs. We will, however, in this work only study the undirected case.

We write $\mathrm{FO}+\mathrm{DP}_k$ for the fragment of $\mathrm{FO}+\mathrm{DP}$ that uses only disjoint-paths$_\ell$ predicates for $\ell \leq k$. The quantifier rank of an $\mathrm{FO}+\mathrm{DP}$ formula is defined as for plain first-order logic. For structures $\mathfrak{A}$ with universe A and $\bar{a} \in A^m$ and $\mathfrak{B}$ with universe B and $\bar{b} \in B^m$, we write $(\mathfrak{A}, \bar{a}) \equiv_{\mathrm{DP}} (\mathfrak{B}, \bar{b})$ if $(\mathfrak{A}, \bar{a})$ and $(\mathfrak{B}, \bar{b})$ satisfy the same $\mathrm{FO}+\mathrm{DP}$ formulas, that is, for all $\varphi(\bar{x})$ we have $\mathfrak{A} \models \varphi(\bar{a}) \Leftrightarrow \mathfrak{B} \models \varphi(\bar{b})$. Similarly, we write $(\mathfrak{A}, \bar{a}) \equiv_{\mathrm{DP}_k} (\mathfrak{B}, \bar{b})$ and $(\mathfrak{A}, \bar{a}) \equiv_{\mathrm{DP}_{k,q}} (\mathfrak{B}, \bar{b})$ if $(\mathfrak{A}, \bar{a})$ and $(\mathfrak{B}, \bar{b})$ satisfy the same $\mathrm{FO}+\mathrm{DP}_k$ formulas and the same $\mathrm{FO}+\mathrm{DP}_k$ formulas of quantifier rank at most q, respectively.

4.1 Expressive power of disjoint-paths logic

We now study the expressive power of disjoint-paths logic.

▶ **Observation 4.2.** $\mathrm{FO}+\mathrm{conn} \subseteq \mathrm{FO}+\mathrm{DP}$ *because* $\mathrm{conn}_k(x, y, z_1, \ldots, z_k)$ *is equivalent to* $\text{disjoint-paths}_{k+1}[(x, y), (z_1, z_1), \ldots, (z_k, z_k)] \wedge \bigwedge_{i \leq k} (z_i \neq x \wedge z_i \neq y)$.

Moreover, the inclusion is strict because planarity is not expressible in $\mathrm{FO}+\mathrm{conn}$ as seen in Corollary 3.10. We show that planarity and in fact the property that a graph contains a fixed (topological) minor can be expressed in $\mathrm{FO}+\mathrm{DP}$.

▶ **Example 4.3.** For every fixed graph H, there is an $\mathrm{FO}+\mathrm{DP}$ formula φ_H^{top} such that $G \models \varphi_H^{top}$ if and only if $H \preccurlyeq^{top} G$.

Let n, m, ℓ respectively be the number of vertices, edges, and isolated vertices in H. Let $x_1, \ldots x_n$ be n variables. Let $e_1, \ldots, e_m$ be the list of edges of H, and let v_{j_s} and v_{j_t} be the two endpoints of e_j. Finally, let $v_{i_1}, \ldots, v_{i_\ell}$ be the isolated vertices of H. Then,

$$\varphi_H^{top} := \exists x_1, \ldots x_n \Big(\bigwedge_{i \neq j} x_i \neq x_j \quad \wedge \\ \text{disjoint-paths}[(x_{e_{1_s}}, x_{e_{1_t}}), \ldots (x_{e_{m_s}}, x_{e_{m_t}}), (x_{i_1}, x_{i_1}), \ldots (x_{i_\ell}, x_{i_\ell})] \Big).$$

▶ **Example 4.4.** For every fixed graph H, there is an $\mathrm{FO}+\mathrm{DP}$ formula φ_H such that $G \models \varphi_H$ if and only if $H \preccurlyeq G$. This is because, for every graph H, there exists a finite family of graphs $H_1, \ldots, H_\ell$ such that $H \preccurlyeq G$ if and only if there is an $i \leq \ell$ such that $H_i \preccurlyeq^{top} G$.

This family can be obtained by considering all possibilities of replacing every branch set representing a vertex of H of degree $d \geq 3$ with a tree with at most d leaves and hardcoding their shapes by disjoint paths.

▶ **Example 4.5.** Planarity can be expressed in $\mathrm{FO} + \mathrm{DP}$. This is a corollary of the previous example, using the formula $\varphi_{planar} := \neg\varphi_{K_5} \wedge \neg\varphi_{K_{3,3}}$.

4.2 The limits of disjoint-paths logic

We now study the limits of disjoint-paths logic and show that bipartiteness cannot be expressed in $\mathrm{FO} + \mathrm{DP}$. We also show that the hierarchy on $(\mathrm{FO} + \mathrm{DP}_k)_{k\geq 1}$ is strict. These results are based again on an adaptation of the standard Ehrenfeucht-Fraïssé game.

The *($\mathrm{DP}_{k,q}$)-game* is played just as the q-round EF game, but the winning condition is changed as follows. If in q rounds the players have chosen $\bar{a} = a_1, \ldots, a_q$ and $\bar{b} = b_1, \ldots, b_q$, then Duplicator wins if

1. the mapping $\bar{a} \mapsto \bar{b}$ is a partial isomorphism of $\mathfrak{A}$ and $\mathfrak{B}$, and
2. for every $\ell \leq k$ and every sequence $(i_1, \ldots, i_{2\ell})$ of numbers in $\{1, \ldots, q\}$ we have

$$\begin{aligned} &\mathfrak{A} \models \text{disjoint-paths}[(a_{i_1}, a_{i_2}), \ldots, (a_{i_{2\ell-1}}, a_{i_{2\ell}})] \\ \iff\ &\mathfrak{B} \models \text{disjoint-paths}[(b_{i_1}, b_{i_2}), \ldots, (b_{i_{2\ell-1}}, b_{i_{2\ell}})]. \end{aligned}$$

Otherwise, Spoiler wins. We say that Duplicator *wins the ($\mathrm{DP}_{k,q}$)-game* on $\mathfrak{A}$ and $\mathfrak{B}$ if she can force a win no matter how Spoiler plays. We then write $\mathfrak{A} \simeq_{\mathrm{DP}_{k,q}} \mathfrak{B}$.

By following the lines of the proof of the classical Ehrenfeucht-Fraïssé Theorem we can prove the following theorem.

▶ **Theorem 4.6.** *Let $\mathfrak{A}$ and $\mathfrak{B}$ be two σ-structures where σ is purely rational (and contains a binary relation symbol E that is interpreted on both structures as an irreflexive and symmetric relation). Then $\mathfrak{A} \equiv_{\mathrm{DP}_{k,q}} \mathfrak{B}$ if and only if $\mathfrak{A} \simeq_{\mathrm{DP}_{k,q}} \mathfrak{B}$.*

▶ **Theorem 4.7.** *Bipartiteness is not definable in* $\mathrm{FO} + \mathrm{DP}$.

Proof. Let q be an integer, and let G be a cycle graph with 2^q vertices and H a cycle graph with $2^q + 1$ vertices. Then, G is bipartite because it has an even number of vertices, and H is not bipartite because it has an odd number of vertices. We want to show that $G \simeq_{\mathrm{DP}_{k,q}} H$ by induction over q.

We define the distance $d(x, y)$ of two vertices x and y as the length of the shortest path between x and y.

Let $\bar{g} = (g_1, \ldots, g_i)$ be the first i moves in G and similarly $\bar{h} = (h_1, \ldots, h_i)$ the first i moves in H. We can prove by induction that Duplicator can play in such a way that after round i of the ($\mathrm{DP}_{k,q}$)-game the following conditions hold for all $j, \ell \leq i$:

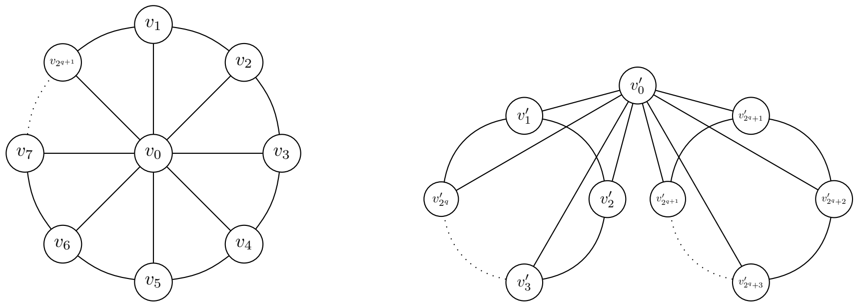
1. If $d(g_j, g_\ell) < 2^{q-i+1}$, then $d(g_j, g_\ell) = d(h_j, h_\ell)$.
2. If $d(g_j, g_\ell) \geq 2^{q-i+1}$, then $d(h_j, h_\ell) \geq 2^{q-i+1}$.
3. The pebbles are placed in G and H with the same "circular order".

By the first two conditions, the partial isomorphism $\bar{g} \mapsto \bar{h}$ can be ensured. Furthermore, the third condition implies that the second condition for Duplicator's win is also satisfied.

The base case $i = 1$ of the induction is trivial because $d(g_1, g_1) = d(h_1, h_1) = 0$.

For the induction step, suppose that $G \simeq_{\mathrm{DP}_{k,i}} H$ holds and Spoiler is making his $(i+1)$-st move in G. The case of H is equivalent.

If Spoiler picks g_j for some $j \leq i$, a pebble that was already played before, Duplicator can choose h_j, and the conditions are fulfilled by the induction hypothesis. Otherwise, Spoiler picks a pebble g_{i+1} that wasn't played before. Now we have to differentiate two cases:



(a) G_q (b) H_q

Figure 2 FO + DP hierarchy is strict.

1. There is only one other pebble that was already played, $g_j = g_1, j \le i$. Then, we can find h_{i+1} such that $d(h_1, h_{i+1}) = d(g_1, g_{i+1})$.
2. g_{i+1} lies on the shortest path of g_j and g_ℓ with $j, \ell \le i$ such that there is no other $g_n, n \le i$ that lies on this path. Then, there are two possibilities:
   - $d(g_j, g_\ell) < 2^{q-i+1}$: Then $d(h_j, h_\ell) < 2^{q-i+1}$ and we can find h_{i+1} on the shortest path of h_j and h_ℓ such that $d(h_j, h_{i+1}) = d(g_j, g_{i+1})$ and $d(h_{i+1}, h_\ell) = d(g_{i+1}, g_\ell)$.
   - $d(g_j, g_\ell) \ge 2^{q-i+1}$: Then $d(h_j, h_\ell) \ge 2^{q-i+1}$ and there are three cases:
     a. $d(g_j, g_{i+1}) < 2^{q-i}$: Then $d(g_{i+1}, g_\ell) \ge 2^{q-i}$ and we can choose h_{i+1} on the shortest path of h_j and h_ℓ such that $d(h_j, h_{i+1}) = d(g_j, g_{i+1})$ and $d(h_{i+1}, h_\ell) \ge 2^{q-i}$.
     b. $d(g_{i+1}, g_\ell) < 2^{q-i}$: This case is similar to the previous one.
     c. $d(g_j, g_{i+1}) \ge 2^{q-i}$ and $d(g_{i+1}, g_\ell) \ge 2^{q-i}$: Since $d(h_j, h_\ell) \ge 2^{q-i+1}$, we can find h_{i+1} with $d(h_j, h_{i+1}) \ge 2^{q-i}$ and $d(h_{i+1}, h_\ell) \ge 2^{q-i}$ in the middle of the shortest path of h_j, and h_ℓ.

Thus, in all cases, the conditions are fulfilled. This completes the inductive proof. ◀

We now show that the hierarchy on $(\mathrm{FO} + \mathrm{DP}_k)_{k \ge 1}$ is strict.

▶ **Lemma 4.8.** *For all integers $k \ge 1$, $2k$-connectivity is not expressible in* $\mathrm{FO} + \mathrm{DP}_k$.

Proof. Let k be an integer. For every integer q, we define two graphs G_q and H_q such that:
- G_q is 2-connected,
- H_q is 1-connected but not 2-connected, and
- $G_q \simeq_q H_q$

For example, take G_q the cycle with 2^{q+1} many elements, together with an apex vertex, while H_q is the disjoint union of two cycles with 2^q many elements each, together with an apex vertex (see Figure 2).

We then define G_q^k (resp. H_q^k) as the lexicographical product of G_q (resp. H_q) with K_{2k}, the clique with $2k$ elements. More precisely, if $G_q = (V, E)$, where $V = \{1, \dots, n\}$, then $G_q^k := (V', E')$ where:
- $V' := \{v_{1,1}, \dots, v_{1,2k}, \dots, v_{n,1}, \dots, v_{n,2k}\}$
- $E' := \{\{v_{i,j}, v_{i',j'}\} \;:\; i = i' \vee (i, i') \in E\}$.

One can view G_q^k as $2k$ copies of G_q on top of each other. Vertices are replaced by $2k$-cliques, and edges are replaced by $(2k, 2k)$-bicliques. A direct consequence of the definition is the following equivalence.

▷ Claim 4.9. For all integers q, k, we have that $G_q^k \simeq_q H_q^k$.

Proof. Duplicator's strategy follows the one derived from $G_q \simeq_q H_q$. If Spoiler picks a vertex $v_{i,j} \in G_q^k$, then Duplicator can respond by choosing the vertex $v_{i',j} \in H_q^k$ where $v_{i'} \in H_q$ is Duplicator's respond to $v_i \in G_q$. ◁

We then show that over G_q^k and H_q^k, the predicate disjoint-paths$_k$[ ] is always true and therefore that, for these structures, $(\mathrm{FO} + \mathrm{DP}_k)[q]$ collapses to $\mathrm{FO}[q]$.

▷ Claim 4.10. For every integers q, k, for every k-tuples $\bar{a}, \bar{b}$, we have that G_q^k and H_q^k both model disjoint-paths$_k[(a_1, b_1), \ldots, (a_k, b_k)]$.

Proof. The proofs for G_q^k and H_q^k are identical, so we only do it for G_q^k. Remember that n is the number of vertices in G_q. The idea is that each of the k paths uses at most two "copies" of each vertex of G_q, hence $2k$ "copies" is enough for all paths to exists. For every $i \leq n$, let $B_i := \{v_{i,j} \ : \ j \leq 2k\}$, and $F_i := \{v_{i,j} \ : \ j \leq 2k \wedge v_{i,j} \notin \bar{a} \wedge v_{i,j} \notin \bar{b}\}$. We call B_i the set of vertices in *position* i, and F_i the *free vertices* in position i. We then compute each path, starting with (a_1, b_1).

Let i, j, i', j' such that $a_1 = v_{i,j}$ and $b_1 = v_{i',j'}$. If $i = i'$, then there is nothing to do as a_1 and b_1 are neighbors. Otherwise, note that for every $i'' \leq n$, $F_{i''} \neq \emptyset$, because there are only $2k - 2$ elements among $a_2, \ldots, a_k, b_2, \ldots, b_k$. Since G_q is a connected graph, there is a path from i to i'. For every inner node i'' of this path, we can select a vertex $v \in F_{i''}$. We can therefore create a path in G_q^k from a_1 to b_1 where all inner vertices are free vertices. We then remove these vertices from the sets of free vertices.

Let now $1 < \ell \leq k$, and let i, j, i', j' such that $a_\ell = v_{i,j}$ and $b_\ell = v_{i',j'}$. We assume that the first $\ell - 1$ paths have already been computed. Observe that here again, if $i = i'$ there is nothing to do. Otherwise, we again have that for every i'', $F_{i''}$ is not empty. This is because for every $s \leq k$, the path from a_s to b_s intersects $B_{i''}$ at most twice *(at most once for the inner vertices, and twice when the two endpoints are both in position i'')*. Therefore, we can select a path in G_q from i to i' and for each i'' in this path, pick a vertex $v \in F_{i''}$. ◁

With Claim 4.10, we can replace formulas of $(\mathrm{FO} + \mathrm{DP}_k)[q]$ by formulas of $\mathrm{FO}[q]$. Thanks to Claim 4.9, $G_q^k \simeq_q H_q^k$, we conclude that $G_q^k \simeq_{\mathrm{DP}_{k,q}} H_q^k$. So $\mathrm{FO} + \mathrm{DP}_k$ cannot express $2k$-connectivity. Note that this bound is tight for these structures i.e. $G_q^k \not\simeq_{\mathrm{DP}_{k+1,q}} H_q^k$. ◀

▶ **Lemma 4.11.** *The* $\mathrm{FO} + \mathrm{DP}_k$ *hierarchy is strict, that is,* $\mathrm{FO} + \mathrm{DP}_1 \subsetneq \mathrm{FO} + \mathrm{DP}_2 \subsetneq \ldots$

Proof. Consider the structures in the proof of Lemma 4.8, which are indistinguishable in $\mathrm{FO} + \mathrm{DP}_k$. The following sentence of $\mathrm{FO} + \mathrm{DP}_{k+1}$ distinguishes G_q^k and H_q^k:

$$\exists a_1 \ldots \exists b_{k+1} \neg \text{disjoint-paths}_{k+1}[(a_1, b_1), \ldots, (a_{k+1}, b_{k+1})]$$

In H_q^k, pick i such that $H_q \setminus i$ is not connected (i' and i'' two disconnected vertices). Then pick $a_j = v_{i,j}$ if $j \leq k$, $b_j = v_{i,k+j}$ if $j \leq k$, and finally $a_{k+1} = v_{i',1}$, $b_{k+1} = v_{i'',1}$. Intuitively, this means that the vertices $v_{i,j}$ are "blocked" for every $j \leq 2k$ by the first k paths and can therefore not be used for the $(k+1)$-st path such that this disjoint path does not exist.

G_q^k does not satisfy the formula because even if we "block" such a clique, there is still a disjoint path connecting every pair of vertices because G_q is 2-connected. ◀

5 Connection to other logics

In this section, we compare the expressive power of the separator logic and the disjoint-paths logic with monadic second-order logic and transitive-closure logic. Figure 3 depicts the connections between these logics.

5.1 Monadic second-order logic

Monadic second-order logic (MSO_1) allows quantification over sets of vertices in addition to the first-order quantifiers. It has a higher expressive power than first-order logic because for example connectivity is expressible in MSO_1 and every first-order formula can be expressed with the first-order quantifiers. Connectivity is expressible by

$$\forall R\Big(\big(\exists x R(x) \wedge \exists x \neg R(x)\big) \rightarrow \exists x \exists y\big(R(x) \wedge \neg R(y) \wedge E(x,y)\big)\Big)$$

By an extension of this formula, we can say that a given set S is connected:

$$\text{conn-set}(S) := \forall R\Big(\big(R \subseteq S \wedge \exists x\ R(x) \wedge \exists x\ (S(x) \wedge \neg R(x))\big) \\ \rightarrow \exists x \exists y\big(R(x) \wedge \neg R(y) \wedge S(y) \wedge E(x,y)\big)\Big)$$

Furthermore, we can express the connectivity operators in MSO_1. The connectivity operator $\mathrm{conn}_0(x,y)$ can be expressed by:

$$\mathrm{conn}_0(x,y) := \forall R\Big(R(x) \wedge \forall v \forall w\big((R(v) \wedge E(v,w)) \rightarrow R(w)\big) \rightarrow R(y)\Big)$$

and $\mathrm{conn}_k(x,y,z_1,\ldots,z_k)$ using conn-set(S) by:

$$\mathrm{conn}_k(x,y,z_1,\ldots,z_k) := \exists S\ \big(\text{conn-set}(S) \wedge S(x) \wedge S(y) \wedge \bigwedge_{i \leq k} \neg S(z_i)\big).$$

We can express the disjoint paths predicates disjoint-paths$_k[(x_1,y_1),\ldots,(x_k,y_k)]$ by:

$$\exists S_1,\ldots,S_k\Bigg(\bigwedge_{i\leq k}\Big(S_i(x_i) \wedge S_i(y_i) \wedge \text{conn-set}(S_i)\Big) \\ \wedge \bigwedge_{i<j\leq k} \forall z\Big(\big(S_i(z) \wedge S_j(z)\big) \rightarrow \big((z = x_i \vee z = y_i) \wedge (z = x_j \vee z = y_j)\big)\Big)\Bigg)$$

Since the disjoint paths operators are expressible in MSO_1, $\mathrm{FO}+\mathrm{DP}$ is included in MSO_1. This inclusion is strict because it is well-known that bipartiteness is expressible in MSO_1:

$$\exists R_1 \exists R_2\Big(\forall x\big(R_1(x) \leftrightarrow \neg R_2(x)\big) \wedge \bigwedge_{i\leq 2} \forall x \forall y\big((R_i(x) \wedge R_i(y)) \rightarrow \neg E(x,y)\big)\Big)$$

but we showed in Theorem 4.7 that bipartiteness is not expressible in $\mathrm{FO}+\mathrm{DP}$.

5.2 Transitive-closure logic

Transitive-closure logic TC^i_j is the enrichment of first-order logic with the transitive-closure operator $[\mathrm{TC}_{\bar{x},\bar{y}}\varphi(\bar{x},\bar{y})]$ where $\bar{x}$ and $\bar{y}$ are tuples of length i and φ is a formula with at most j free variables other than $\bar{x}$ and $\bar{y}$.

Every $\mathrm{FO}+\mathrm{conn}_k$ formula can be expressed in TC^1_k because the conn_k operator can be expressed with the help of the transitive-closure operator:

$$\mathrm{conn}_k(x,y,z_1,\ldots,z_k) = [\mathrm{TC}_{v,w} E(v,w) \wedge v \neq z_1 \wedge \ldots \wedge v \neq z_k \wedge w \neq z_1 \wedge \ldots \wedge w \neq z_k](x,y)$$

In fact, TC^1_k is more expressible than $\mathrm{FO}+\mathrm{conn}_k$, as it can express bipartiteness [17, Example 7.2]. On the other hand, 2-connectivity can naturally be expressed in $\mathrm{FO}+\mathrm{conn}_1$, but presumably not in TC^1_0.

▶ **Conjecture 5.1.** *2-connectivity cannot be expressed in* TC^1_0.

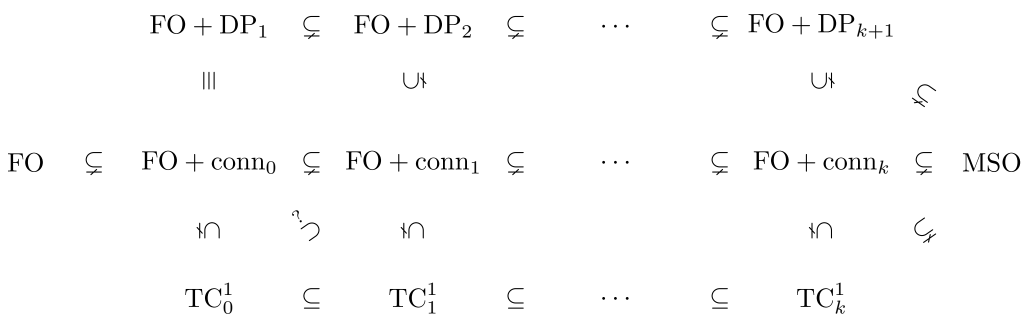


Figure 3 Connections between the logics.

6 Conclusion

We studied first-order logic enriched with connectivity predicates tailored to express algorithmic graph properties that are commonly studied in contemporary parameterized algorithmics. This yielded separator logic, which can query connectivity after the deletion of a bounded number of elements, and disjoint-paths logic, which can express the disjoint-paths problem. We demonstrated a rich expressiveness that arises from the interplay of these predicates with the nested quantification of first-order logic. We also studied the limits of expressiveness of these new logics.

In a companion paper, we studied the model-checking problem for separator logic and proved that it is fixed-parameter tractable parameterized by formula size on classes of graphs that exclude a fixed topological minor [24]. This yields a powerful algorithmic meta-theorem for separator logic. On the other hand, while the disjoint-paths problem is fixed-parameter tractable on general graphs [26], it is not clear that the model-checking problem for disjoint-paths logic is fixed-parameter tractable beyond graphs of bounded treewidth. This remains a challenging question for future work.

It will also be interesting to study other extensions of first-order logic that can express further interesting algorithmic graph problems, such as reachability with regular paths queries. This would, in the simplest case, allow to express bipartiteness and the odd cycle transversal problem. On the other hand, it is very likely that with general regular paths queries, we will get intractability beyond bounded treewidth graphs.

References

1 Akanksha Agrawal, Lawqueen Kanesh, Fahad Panolan, M. S. Ramanujan, and Saket Saurabh. An FPT algorithm for elimination distance to bounded degree graphs. In *38th International Symposium on Theoretical Aspects of Computer Science (STACS 2021)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2021.

2 Mikołaj Bojańczyk. Separator logic and star-free expressions for graphs. *arXiv preprint*, 2021. `arXiv:2107.13953`.

3 Édouard Bonnet, Ugo Giocanti, Patrice Ossona de Mendez, Pierre Simon, Stéphan Thomassé, and Szymon Toruńczyk. Twin-width IV: ordered graphs and matrices. *arXiv preprint*, 2021. `arXiv:2102.03117`.

4 Jannis Bulian. Parameterized complexity of distances to sparse graph classes. Technical report, University of Cambridge, Computer Laboratory, 2017.

5 Jannis Bulian and Anuj Dawar. Graph isomorphism parameterized by elimination distance to bounded degree. *Algorithmica*, 75(2):363–382, 2016.

6 Michael Buro. Simple amazons endgames and their connection to Hamilton circuits in cubic subgrid graphs. In *International Conference on Computers and Games*, pages 250–261. Springer, 2000.

7 Bruno Courcelle. The monadic second-order logic of graphs. I. recognizable sets of finite graphs. *Information and computation*, 85(1):12–75, 1990.

8 Bruno Courcelle, Johann A. Makowsky, and Udi Rotics. Linear time solvable optimization problems on graphs of bounded clique-width. *Theory of Computing Systems*, 33(2):125–150, 2000.

9 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.

10 Heinz-Dieter Ebbinghaus and Jörg Flum. *Finite model theory*. Springer Science & Business Media, 2005.

11 Ronald Fagin. Generalized first-order spectra and polynomial-time recognizable sets. *Complexity of computation*, 7:43–73, 1974.

12 Fedor V. Fomin, Petr A. Golovach, Giannos Stamoulis, and Dimitrios M. Thilikos. An algorithmic meta-theorem for graph modification to planarity and FOL. In *28th Annual European Symposium on Algorithms, ESA 2020*, pages 51:1–51:17, 2020.

13 Fedor V. Fomin, Petr A. Golovach, and Dimitrios M. Thilikos. Parameterized complexity of elimination distance to first-order logic properties. *arXiv preprint*, 2021. `arXiv:2104.02998`.

14 Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, and Meirav Zehavi. Hitting topological minors is FPT. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1317–1326, 2020.

15 Robert Ganian, Petr Hliněnỳ, Alexander Langer, Jan Obdržálek, Peter Rossmanith, and Somnath Sikdar. Lower bounds on the complexity of MSO1 model-checking. *Journal of Computer and System Sciences*, 80(1):180–194, 2014.

16 Erich Grädel, Phokion G. Kolaitis, Leonid Libkin, Maarten Marx, Joel Spencer, Moshe Y. Vardi, Yde Venema, and Scott Weinstein. *Finite Model Theory and its applications*. Springer Science & Business Media, 2007.

17 Martin Grohe. Logic, graphs, and algorithms. *Logic and automata*, 2:357–422, 2008.

18 Martin Grohe, Stephan Kreutzer, and Sebastian Siebertz. Deciding first-order properties of nowhere dense graphs. *Journal of the ACM (JACM)*, 64(3):17, 2017.

19 Eva-Maria C. Hols, Stefan Kratsch, and Astrid Pieterse. Elimination distances, blocking sets, and kernels for vertex cover. In *STACS*, 2020.

20 Bart M. P. Jansen, Jari J. H. de Kroon, and Michał Włodarczyk. Vertex deletion parameterized by elimination distance and even less. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1757–1769, 2021.

21 Stephan Kreutzer and Siamak Tazari. Lower bounds for the complexity of monadic second-order logic. In *2010 25th Annual IEEE Symposium on Logic in Computer Science*, pages 189–198. IEEE, 2010.

22 Leonid Libkin. *Elements of finite model theory*. Springer Science & Business Media, 2013.

23 Alexander Lindermayr, Sebastian Siebertz, and Alexandre Vigny. Elimination distance to bounded degree on planar graphs. In *45th International Symposium on Mathematical Foundations of Computer Science, MFCS 2020, August 24-28, 2020, Prague, Czech Republic*, pages 65:1–65:12, 2020.

24 Michał Pilipczuk, Nicole Schirrmacher, Sebastian Siebertz, Szymon Toruńczyk, and Alexandre Vigny. Algorithms and data structures for first-order logic with connectivity under vertex failures. *arXiv preprint*, 2021. `arXiv:2111.03725`.

25 Bruce Reed, Kaleigh Smith, and Adrian Vetta. Finding odd cycle transversals. *Operations Research Letters*, 32(4):299–301, 2004.

26 Neil Robertson and P. D. Seymour. Graph minors. XIII. the disjoint paths problem. *J. Combin. Theory Ser. B*, 63:65–110, 1995.

27 Wolfgang Thomas. Languages, automata, and logic. In Grzegorz Rozenberg and Arto Salomaa, editors, *Handbook of Formal Languages, Volume 3: Beyond Words*, pages 389–455. Springer, 1997. `doi:10.1007/978-3-642-59126-6_7`.