Compound Logics for Modification Problems

Fedor V. Fomin, Petr A. Golovach, Ignasi Sau, Giannos Stamoulis, Dimitrios M. Thilikos

Compound Logics for Modification Problems

Fedor V. Fomin
Department of Informatics, University of Bergen, Norway

Petr A. Golovach
Department of Informatics, University of Bergen, Norway

Ignasi Sau
LIRMM, Université de Montpellier, CNRS, Montpellier, France

Giannos Stamoulis
LIRMM, Université de Montpellier, CNRS, Montpellier, France

Dimitrios M. Thilikos
LIRMM, Université de Montpellier, CNRS, Montpellier, France

Abstract

We introduce a novel model-theoretic framework inspired from graph modification and based on the interplay between model theory and algorithmic graph minors. The core of our framework is a new *compound logic* operating with two types of sentences, expressing graph modification: the *modulator sentence*, defining some property of the modified part of the graph, and the *target sentence*, defining some property of the resulting graph. In our framework, modulator sentences are in counting monadic second-order logic (CMSOL) and have models of bounded treewidth, while target sentences express first-order logic (FOL) properties along with minor-exclusion. Our logic captures problems that are not definable in first-order logic and, moreover, may have instances of unbounded treewidth. Also, it permits the modeling of wide families of problems involving vertex/edge removals, alternative modulator measures (such as elimination distance or $\mathcal{G}$-treewidth), multistage modifications, and various cut problems. Our main result is that, for this compound logic, model-checking can be done in quadratic time. All derived algorithms are constructive and this, as a byproduct, extends the constructibility horizon of the algorithmic applications of the Graph Minors theorem of Robertson and Seymour. The proposed logic can be seen as a general framework to capitalize on the potential of the *irrelevant vertex technique*. It gives a way to deal with problem instances of unbounded treewidth, for which Courcelle's theorem does not apply.

2012 ACM Subject Classification Theory of computation → Logic; Theory of computation → Parameterized complexity and exact algorithms; Mathematics of computing → Graph algorithms

Keywords and phrases Algorithmic meta-theorems, Graph modification problems, Model-checking, Graph minors, First-order logic, Monadic second-order logic, Flat Wall theorem, Irrelevant vertex technique.

Digital Object Identifier 10.4230/LIPIcs.ICALP.2023.51

Related Version *Full version*: http://arxiv.org/abs/2111.02755 [31]

Funding *Fedor V. Fomin*: Supported by the Research Council of Norway via the project BWCA (grant no. 314528).
Petr A. Golovach: Supported by the Research Council of Norway via the project BWCA (grant no. 314528).
Ignasi Sau: Supported by the ANR projects ELIT (ANR-20-CE48-0008-01) and ESIGMA (ANR-17-CE23-0010) and the French-German Collaboration ANR/DFG Project UTMA (ANR-20-CE92-0027).
Giannos Stamoulis: Supported by the ANR project ESIGMA (ANR-17-CE23-0010) and the French-German Collaboration ANR/DFG Project UTMA (ANR-20-CE92-0027).
Dimitrios M. Thilikos: Supported by the ANR project ESIGMA (ANR-17-CE23-0010) and the French-German Collaboration ANR/DFG Project UTMA (ANR-20-CE92-0027).

Acknowledgements We wish to thank the anonymous reviewers for their valuable remarks.

1 Introduction

Our work is kindled by the current algorithmic advances in graph modification. The core of our approach is a novel model-theoretic framework that is based on the interplay between model theory and algorithmic graph minors. Departing from this new perspective, we obtain algorithmic meta-theorems that encompass, unify, and extend all known meta-algorithmic results on minor-closed graph classes.

1.1 State of the art and our contribution

Modification problems. A *graph modification problem* asks whether it is possible to apply a series of modifications to a graph in order to transform it to a graph with some desired target property. Such problems have been the driving force of Parameterized Complexity where parameterization quantifies the concept of "distance from triviality" [48] and measures the amount of the applied modification. Classically, modification operations may be vertex or edge deletions, edge additions/contractions, or combinations of them like taking a minor. In their generality, such problems are NP-complete [60] and much research in Parameterized Complexity is on the design of algorithms in time $f(k) \cdot n^{\mathcal{O}(1)}$, where the parameter k is some measure of the modification operation [20]. The target property may express desired structural properties that respond to certain algorithmic or combinatorial demands. A widely studied family of target properties are minor-closed graph classes such as edgeless graphs [14], forests [13], bounded treewidth graphs [35, 54], planar graphs [50, 62], bounded genus graphs [55], or, most generally, minor-excluding graphs [72, 73]. However, other families of target properties have also been considered, such as those that exclude an odd cycle [29], a topological minor [36], an (induced) subgraph [22, 69], an immersion [40], or an induced minor [41]. A broad class of graph modification problems concerns cuts. In a typical cut problem, one wants to find a minimum-size set of edges or vertices X in a graph G such that in the new graph $G \setminus X$, obtained by deleting X from G, some terminal-connectivity conditions are satisfied. For example, the condition can be that a set of specific terminals becomes separated or that at least one connected component in the new graph is of a specific size. The development of parameterized algorithms for cut problems is a popular trend in parameterized algorithms [21, 61]. More involved modification measures of vertex set removals, related to treewidth or treedepth, have been considered very recently [1, 12, 27, 49].

Algorithmic meta-theorems. A vibrant line of research in Logic and Algorithms is the development of *algorithmic meta-theorems*. According to Grohe and Kreutzer [45], algorithmic meta-theorems state that certain families of algorithmic problems, typically defined by some logical and some combinatorial condition, can be solved "efficiently", under some suitable definition of this term. Algorithmic meta-theorems play an important role in the theory of algorithms as they reveal deep interplays between Algorithms, Logic, and Combinatorics. One of the most celebrated meta-theorems is Courcelle's theorem asserting that graph properties definable in CMSOL (counting monadic second-order logic) are decidable in linear time on graphs of bounded treewidth [15]; see also [2, 10]. Another stream of research concerns identifying wide combinatorial structures where model-checking for FOL (first-order logic) can be done in polynomial time. This includes graph classes of bounded degree [76], graph classes of bounded local treewidth [37], *minor-closed graph classes* [30], graph classes locally excluding a minor [23], and more powerful concepts of sparsity, such as having bounded expansion [26, 63], nowhere denseness [46], or having bounded twin-width [9]. (See [44, 58] for surveys. Also for results on the combinatorial horizon of FOL and CMSOL (and its variants) see [8, 9, 46] and [57] respectively.)

Another line of research, already mentioned in [44], is to prove algorithmic meta-theorems for extensions of FOL of greater expressibility. Two such extensions have been recently presented. The first one consists in enhancing FOL with predicates that can express k-connectivity for every $k \geq 1$. This extension of FOL was introduced independently by Schirrmacher, Siebertz, and Vigny in [75] (under the name FOL+conn) and by Bojańczyk in [6] (under the name *separator logic*). The second and more expressive extension, also introduced by Schirrmacher, Siebertz, and Vigny in [75], is FOL+DP, that enhances FOL with predicates expressing the existence of disjoint paths between certain pairs of vertices. For FOL+conn, an algorithmic meta-theorem for model-checking on graphs excluding a topological minor has been very recently given by Pilipczuk, Schirrmacher, Siebertz, Torunczyk, and Vigny [66]. For the more expressive FOL+DP, an algorithmic meta-theorem for model-checking on graphs excluding a minor has been very recently given by Golovach, Stamoulis, and Thilikos in [43] (see [42] for the full version).

Research on the meta-algorithmics of FOL is quite active and has moved to several directions such as the study of FOL-interpretability [7, 39, 64, 65] or the enhancement of FOL with counting/numerical predicates [25, 47, 59].

In this paper, we initiate an alternative approach consisting in combining the expressive power of FOL and CMSOL. A typical family of problems where such an approach becomes relevant is the one of modification problems. Courcelle's theorem implies that if the target property corresponds to a class of bounded treewidth and the modification conditions are definable in CMSOL, then such modification problems are fixed-parameter tractable when parameterized by the length of the sentence and the treewidth of the graph. However, when the target class graph is of unbounded treewidth, none of the aforementioned algorithmic meta-theorems encompasses broad families of modification problems. As an illustrative example, consider the Planarization problem, which consists in deciding whether at most k vertices can be removed from an input graph to make it planar (or equivalently, minor-excluding K_5 and $K_{3,3}$). While this problem is definable in CMSOL, Courcelle's theorem cannot be applied as we cannot assume that yes-instances are of bounded treewidth. On the other hand, we can easily assume that yes-instances minor-exclude K_{k+6}. However, all known meta-theorems whose combinatorial condition encompasses the minor-exclusion are about FOL, and FOL cannot express the Planarization problem. On the positive side, an algorithm in time $f(k) \cdot n^2$ for Planarization is an algorithmic consequence of Robertson-Seymour's theorem [68] (combined with [51, 67]). This automatic implication follows directly (albeit non-constructively) for a wide family of modification problems whose yes-instances are minor-closed. There is a long line of research in parameterized algorithms towards providing constructive and reasonable estimations of $f(k)$ [50, 62, 72, 73]. Note that Robertson-Seymour's theorem, besides not being constructive in general, automatically offers results only for problems whose yes-instances are minor-closed.

Our contribution. We introduce a *compound logic* that models computational problems through the lens of the "modulator vs target" duality of graph modification problems. Each sentence of this logic is a composition of two types of sentences. The first one, called the *modulator sentence*, models a modification operation, while the second one, called the *target sentence*, models a target property. Informally, our result, in its simplest form, asserts that if some appropriate version of the modulator sentence meets the meta-algorithmic assumptions of Courcelle's theorem [15] (i.e., CMSOL-definability and bounded treewidth) and the target sentence meets the meta-algorithmic assumptions of the theorem of Flum and Grohe [30] (i.e., FOL-definability and minor-exclusion), then model-checking for the composed compound sentence can be done, constructively, in quadratic time. Our main result (Theorem 5) can be seen as a "two-dimensional product" of the two aforementioned meta-algorithmic results, contains both of them as special cases, and automatically implies the tractability of wide families of problems that *neither* are FOL-definable *nor* have instances of bounded treewidth.

1.2 Our results

In this subsection we give formal statements of our results. We need first some definitions.

Preliminaries on graphs. Given a graph G, we denote by $\mathsf{cc}(G)$ the set of all connected components of G. For a graph G and a set $X \subseteq V(G)$, the *stellation* of X in G is the graph $\mathsf{stell}(G,X)$ obtained from G if, for every $C \in \mathsf{cc}(G \setminus X)$, we contract all edges of C to a single vertex v_C. The *torso* of X in G is the graph $\mathsf{torso}(G,X)$ obtained from $\mathsf{stell}(G,X)$ if, for every v_C where $C \in \mathsf{cc}(G \setminus X)$, we add all edges between neighbors of v_C and finally remove all v_C's from the resulting graph. Given a family of graphs $\mathcal{H}$, we define $\mathsf{excl}(\mathcal{H})$ as the class of all graphs minor-excluding the graphs in $\mathcal{H}$ and note that $\mathsf{excl}(\mathcal{H})$ is minor-closed. The *Hadwiger number* of a graph G, denoted by $\mathbf{hw}(G)$, is the minimum k where $G \in \mathsf{excl}(\{K_k\})$ and K_k is the complete graph on k vertices. We also use the well-known parameter of *treewidth* of a graph G, denoted by $\mathbf{tw}(G)$. Given a graph class $\mathcal{G}$, we define $\mathbf{tw}(\mathcal{G}) = \max\{\mathbf{tw}(G) \mid G \in \mathcal{G}\}$. We define $\mathbf{hw}(\mathcal{G})$ analogously. We use $\mathcal{G}_{\text{all}}$ for the set of all graphs.

Preliminaries on logic. We use CMSOL (resp. FOL) for the set of sentences in counting monadic second-order logic (resp. first-order logic). Given some vocabulary τ and a sentence $\varphi \in \mathsf{CMSOL}[\tau]$, we denote by $\mathrm{Mod}(\varphi)$ the set of all finite models of φ, i.e., all structures that are models of φ. In this introduction, in order to simplify our presentation, all structures that we consider are either graphs or annotated graphs, i.e., pairs (G,X) where G is a graph and $X \subseteq V(G)$. In the first case $\tau = \{\mathsf{E}\}$, and in the second $\tau = \{\mathsf{E},\mathsf{X}\}$.

Given a $\varphi \in \mathsf{CMSOL}[\{\mathsf{E}\}]$, we define the *connectivity extension* $\varphi^{(\mathsf{c})}$ of φ so that $G \models \varphi^{(\mathsf{c})}$ if $\forall C \in \mathsf{cc}(G), C \models \varphi$. Similarly, for every $\mathcal{L} \subseteq \mathsf{CMSOL}[\{\mathsf{E}\}]$, we define $\mathcal{L}^{(\mathsf{c})} = \mathcal{L} \cup \{\varphi^{(\mathsf{c})} \mid \varphi \in \mathcal{L}\}$. Notice that $\{\varphi\}^{(\mathsf{c})} = \{\varphi, \varphi^{(\mathsf{c})}\}$. Also by $\mathbf{PB}(\mathcal{L})$ we denote the set of all positive Boolean combinations (i.e., using only the Boolean connectives $\vee$ and $\wedge$) of sentences in $\mathcal{L}$. We next define the following sets of sentences:

- The set $\mathsf{CMSOL}^{\mathsf{tw}}[\{\mathsf{E},\mathsf{X}\}]$ contains every sentence $\beta \in \mathsf{CMSOL}[\{\mathsf{E},\mathsf{X}\}]$ for which there exists some c_β such that the torsos of all the models of β have treewidth at most c_β. Formally,
  $\mathsf{CMSOL}^{\mathsf{tw}}[\{\mathsf{E},\mathsf{X}\}] = \{\beta \in \mathsf{CMSOL}[\{\mathsf{E},\mathsf{X}\}] \mid \exists c_\beta : \mathbf{tw}\{\mathsf{torso}(G,X) \mid (G,X) \models \beta\} \leq c_\beta\}$.
- The set $\mathsf{EM}[\{\mathsf{E}\}]$ is the set of all sentences in $\mathsf{CMSOL}[\{\mathsf{E}\}]$ that express the minor-exclusion of a non-empty set of graphs. Formally,
  $\mathsf{EM}[\{\mathsf{E}\}] = \{\mu \in \mathsf{CMSOL}[\{\mathsf{E}\}] \mid \exists \mathcal{H} \subseteq \mathcal{G}_{\text{all}}, \mathcal{H} \neq \emptyset : \mathrm{Mod}(\mu) = \mathsf{excl}(\mathcal{H})\}$.
- $\Theta_0[\{\mathsf{E}\}]$ contains every sentence $\sigma \wedge \mu$ where $\sigma \in \mathsf{FOL}[\{\mathsf{E}\}]$ and $\mu \in \mathsf{EM}[\{\mathsf{E}\}]$.

For simplicity, we use $\mathsf{CMSOL}^{\mathsf{tw}}$, EM, and Θ_0 as shortcuts for $\mathsf{CMSOL}^{\mathsf{tw}}[\{\mathsf{E},\mathsf{X}\}]$, $\mathsf{EM}[\{\mathsf{E}\}]$, and $\Theta_0[\{\mathsf{E}\}]$, respectively. Note that both $\mathsf{CMSOL}^{\mathsf{tw}}$ and Θ_0 are undecidable.

Algorithmic meta-theorems. We are now in position to restate three major meta-algorithmic results that were mentioned in the previous subsection.

▶ **Proposition 1** (Courcelle [15]). *For every $\beta \in \mathsf{CMSOL}^{\mathsf{tw}}$, there is an algorithm deciding $\mathrm{Mod}(\beta)$ in linear time.*

▶ **Proposition 2** (Robertson and Seymour [67,68] and Kawarabayashi, Kobayashi, and Reed [51]). *For every minor-closed graph class $\mathcal{G}$, deciding membership in $\mathcal{G}$ can be done in quadratic time.*

▶ **Proposition 3** (Flum and Grohe [30]). *For every $\gamma \in \Theta_0$, there is an algorithm deciding* $\mathrm{Mod}(\gamma)$ *in quadratic time.*

Some comments are in order. The statements of Proposition 1 and Proposition 3 have been adapted so to incorporate the combinatorial demands in the logical condition. While they can both be stated for structures, we state Proposition 1 for annotated graphs and Proposition 3 for graphs in order to facilitate our presentation. In the classic formulation of Courcelle's theorem, we are given a sentence $\beta \in \mathsf{CMSOL}$ and a tree decomposition of bounded treewidth. As such a decomposition can be found in linear time, using e.g., [4, 56], the linearity in the running time of Courcelle's theorem is preserved when it is stated in the form of Proposition 1. For the theorem of Flum and Grohe, the situation is different as the combinatorial demand is minor-exclusion of a clique, which is not definable is FOL. For this reason we state Proposition 3 using the logic Θ_0 that contains *compound* sentences of the form $\sigma \wedge \mu$, where $\sigma \in \mathsf{FOL}$ and μ expresses minor-exclusion. For the running time of the algorithm of Proposition 3, we also need to take into account Proposition 2. As we already mentioned, Proposition 1 and Proposition 3 cannot deal, in general, with modification problems to properties of unbounded treewidth. Moreover, recall that Proposition 2 applies only to problems whose yes-instances are minor-closed.

We stress that Proposition 1, Proposition 2, and Proposition 3 are non-constructive. In order to construct the algorithms promised by Proposition 1, one should also know the bound c_β on the treewidth of the models of $\beta \in \mathsf{CMSOL}^{\mathsf{tw}}$ (note that bounded treewidth is also CMSOL-definable since it is characterized by a finite set of forbidden minors) and this appears in the hidden constants in the running time in Proposition 1. Similarly, for Proposition 2 (resp. Proposition 3), one should have an upper bound on the Hadwiger number of the graphs in $\mathcal{G}$ (resp. the models of γ).

A logic for modification problems. As a key ingredient of our result, we define the following operation between sentences. Let $\beta \in \mathsf{CMSOL}[\{\mathsf{E}, \mathsf{X}\}]$ and $\gamma \in \mathsf{CMSOL}[\{\mathsf{E}\}]$. We refer to β as the *modulator* sentence on annotated graphs and to γ as the *target* sentence on graphs. We define $\beta \triangleright \gamma$ so that

$$G \models \beta \triangleright \gamma \text{ if there is } X \subseteq V(G) \text{ such that } (\mathsf{stell}(G, X), X) \models \beta \text{ and } G \setminus X \models \gamma. \tag{1}$$

In other words, $G \models \beta \triangleright \gamma$ means that the stellation of X in G, along with X, is a model of the modulator sentence β and the $G \setminus X$ is a model of the target sentence γ. That way, β implies the modification operation and γ expresses the target graph property. It is easy to see that $\beta \triangleright \gamma \in \mathsf{CMSOL}[\{\mathsf{E}\}]$. This will allow us to apply the operation $\triangleright$ iteratively.

As an example, the problem of removing a set X of k vertices so that $G \setminus X$ is a triangle-free planar graph can be expressed by $\beta \triangleright \gamma$ if β asks that X has k vertices and $\gamma = \sigma \wedge \mu$, where σ expresses triangle-freeness and μ expresses planarity by the exclusion of $K_{3,3}$ and K_5.

Before we present our result in full generality, we give first the following indicative special case, which already expresses the conditions of Proposition 1 and Proposition 3.

▶ **Theorem 4.** *For every $\beta \in \mathsf{CMSOL}^{\mathsf{tw}}$ and every $\gamma \in \Theta_0$, there is an algorithm deciding* $\mathrm{Mod}(\beta \triangleright \gamma)$ *in quadratic time.*

Indeed, Proposition 1 follows if β expresses that $X = V(G)$ and γ demands that $G \setminus X$ is the empty graph (in particular, Theorem 4 contains Proposition 1 as a linear-time black-box procedure for deciding models of bounded treewidth) and Proposition 3 follows if β demands that $X = \emptyset$. In other words, Proposition 1 follows if the target sentence becomes void while Proposition 3 follows if the modulator sentence is void.

As a first step towards a more general statement, Theorem 4 also holds if we replace $\gamma \in \Theta_0$ by $\gamma \in \Theta_0^{(\mathsf{c})}$ or even by positive Boolean combinations of sentences in $\Theta_0^{(\mathsf{c})}$, i.e., $\gamma \in \mathbf{PB}(\Theta_0^{(\mathsf{c})})$. Moreover, in order to present our result in full generality, we recursively define, for every $i \geq 1$,

$$\Theta_i \quad = \quad \{\beta \triangleright \gamma \mid \beta \in \mathsf{CMSOL}^{\mathsf{tw}} \text{ and } \gamma \in \mathbf{PB}(\Theta_{i-1}^{(\mathsf{c})})\}. \tag{2}$$

Notice that the sentences of Theorem 4 (hence also of Proposition 1 and Proposition 3) are already contained in Θ_1. We set $\Theta = \bigcup_{i\geq 1} \Theta_i$. The full strength of our results, stated in the vocabulary of graphs, is given by our main theorem.

▶ **Theorem 5.** *For every $\theta \in \Theta$, model-checking for θ can be done in quadratic time.*

An alternative statement. Our results can also be seen under the typical meta-algorithmic framework where a logical and a combinatorial condition are given. For this, consider an alternative of Θ, called $\tilde{\Theta}$, that is defined as in (2) by taking $\tilde{\Theta}_0 = \mathsf{FOL}$ as the base case, i.e., by discarding the minor-exclusion from the definition of Θ_0. Notice that $\tilde{\Theta}$ contains FOL and can be seen as a natural extension of it. A direct consequence of Theorem 5 is the following.

▶ **Theorem 6.** *For every $\tilde{\theta} \in \tilde{\Theta}$, model-checking for $\tilde{\theta}$ can be done in quadratic time on every graph class of bounded Hadwiger number.*

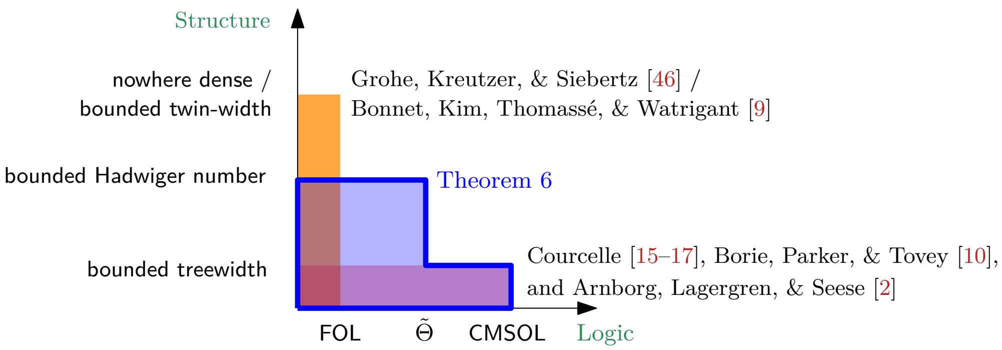

Figure 1 Theorem 6 in the current meta-algorithmic landscape. The vertical axis is the combinatorial one and is marked by four different types of (structural) sparsity, while the horizontal one is the logical one and is marked with FOL, $\tilde{\Theta}$, and CMSOL.

Theorem 6 is a corollary of Theorem 5 and provides an alternative meta-algorithmic set up between the logical and the combinatorial condition (see Figure 1): for each sentence θ in Θ, one may consider a sentence $\tilde{\theta}$ in $\tilde{\Theta}$ where we discard minor-exclusion from all its target sentences and then consider the problem of deciding $\mathrm{Mod}(\theta)$ on some minor-excluding graph class. This correspondence is many-to-one, as many different $\theta \in \Theta$ correspond to the same $\tilde{\theta} \in \tilde{\Theta}$. We opted for presenting and proving our results in the form of Theorem 5, as it is more general and more versatile in expressing modification problems. In the full version of the paper [31], we define we define Θ on general structures.

Compound logics based on FOL+DP. In the full version of the paper [31], by combining our proofs with the meta-algorithmic results of [42, 43], we extend Theorem 5 (resp. Theorem 6) in the cases of the logic Θ^{DP} (resp. $\tilde{\Theta}^{\mathsf{DP}}$) that are obtained if in the definition of Θ (resp. $\tilde{\Theta}$) we now consider the (more expressive) logic FOL+DP instead of FOL in the target sentences. That way, the derived extensions of Theorem 5 and Theorem 6 (that is, Theorem 8 and Theorem 9) encompass, as special cases, all results and applications in [42, 43] (see Figure 3 for a visualization of the overall state-of-the-art on the related algorithmic meta-theorems on subgraph-closed graph classes). While presenting our results and techniques, for the sake of simplicity, we chose to focus on the statement and the proof of our meta-theorems for Θ (Theorem 5) and $\tilde{\Theta}$ (Theorem 6) and then, in the full version of the paper [31], present the modifications that should be applied in order to extend them for Θ^{DP} and $\tilde{\Theta}^{\mathsf{DP}}$.

A parametric variant of our results. A *graph parameter* is a function $\mathbf{p} : \mathcal{G}_{\text{all}} \to \mathbb{N}$. We say that $\mathbf{p}$ is *treewidth-bounded* if there is a function $f : \mathbb{N} \to \mathbb{N}$ such that for each $G \in \mathcal{G}_{\text{all}}$, $\mathbf{p}(G) \leq f(\mathsf{tw}(G))$. We say that $\mathbf{p}$ is CMSOL-*definable* if for every $k \in \mathbb{N}$ there is a CMSOL-sentence (on graphs) β_k such that the set of all models of β_k is $\text{Mod}(\beta_k) = \{G \mid \mathbf{p}(G) \leq k\}$. Clearly, if $\mathbf{p}$ is treewidth-bounded then we can also assume that each β_k is a sentence in $\mathsf{CMSOL}^{\mathsf{tw}}$ and in this case we say that $\mathbf{p}$ is $\mathsf{CMSOL}^{\mathsf{tw}}$-definable. There are several known graph parameters that are $\mathsf{CMSOL}^{\mathsf{tw}}$-definable, such as treewidth, pathwidth, tree-depth, bridge-depth, block tree-depth, vertex cover, feedback vertex set, branch-width, carving-width, or cutwidth.

For a graph parameter $\mathbf{p}$ and a graph class $\mathcal{G}$, we define the new graph parameter $\mathbf{p}_{\mathcal{G}} : \mathcal{G}_{\text{all}} \to \mathbb{N}$ such that

$$\mathbf{p}_{\mathcal{G}}(G) \quad = \quad \min\{k \mid \exists X \subseteq V(G) \mid \mathbf{p}(\mathsf{torso}(G, X)) \leq k \ \wedge \ G \setminus X \in \mathcal{G}\}.$$

Thus $\mathbf{p}_{\mathcal{G}}$ measures by $\mathbf{p}$ the quality of a modulator X to property $\mathcal{G}$. For example, when $\mathbf{p}$ is the size of the modulator, then this is just the *vertex deletion distance to* $\mathcal{G}$, that is, the minimum number of vertices X such that $G \setminus X \in \mathcal{G}$. When $\mathbf{p}$ is the tree-depth of a graph, then $\mathbf{p}_{\mathcal{G}}$ is the elimination distance to $\mathcal{G}$. Or when $\mathbf{p}$ is the treewidth of a graph, then $\mathbf{p}_{\mathcal{G}}$ corresponds to $\mathcal{G}$-treewidth. We consider the general setting where $\mathbf{p}$ is a $\mathsf{CMSOL}^{\mathsf{tw}}$-definable graph parameter and $\mathcal{G}$ is a Θ-definable graph class, that is, $\text{Mod}(\theta) = \mathcal{G}$ for some $\theta \in \Theta$. By setting $\theta_k = \beta_k \triangleright \theta \in \Theta$, we have that $\text{Mod}(\theta_k) = \{G \mid \mathbf{p}_{\mathcal{G}}(G) \leq k\}$. Then the following theorem is a direct consequence of Theorem 5 and Theorem 6.

▶ **Theorem 7.** *Let $\mathbf{p}$ be a $\mathsf{CMSOL}^{\mathsf{tw}}$-definable graph parameter and $\mathcal{G} = \text{Mod}(\theta)$ for some $\theta \in \Theta$. Then there is an algorithm that, with input a graph G and $k \in \mathbb{N}$, checks whether $\mathbf{p}_{\mathcal{G}}(G) \leq k$ in time $\mathcal{O}_{k,|\theta|}(n^2)$. Moreover, if $\mathcal{G} = \text{Mod}(\tilde{\theta})$ for some $\tilde{\theta} \in \tilde{\Theta}$, then there is an algorithm that, with the same input, checks whether $\mathbf{p}_{\mathcal{G}}(G) \leq k$ in time $\mathcal{O}_{k,|\theta|,\mathsf{hw}(G)}(n^2)$.*

All the results mentioned in this subsection, in what concerns minor-excluded graphs, are subsumed by Theorem 7. Moreover, by allowing FOL-definability in the target sentence and $\mathsf{CMSOL}^{\mathsf{tw}}$-definability in the modulator sentence, we vastly extend Proposition 2 to graph classes and parameters that are not necessarily minor-closed or hereditary. We stress that none of the results in [43, 66] is able to deal with the problems captured by Theorem 7 in their full generality.

Constructibility. While Robertson-Seymour's theorem (Proposition 2) implies the existence of an algorithm, its proof is not constructive and cannot be used to construct such an algorithm [28]. An extra feature of the proof of Theorem 5 (as well as of its corollary Theorem 6) is that it is *constructive*, in the sense that the implied algorithms can be constructed if we are given some bound on the Hadwiger number of the models of θ. This considerably extends the constructibility horizon of Proposition 2 for graph classes that are not necessarily minor-closed or even hereditary. See the full version of the paper [31] for more details.

Techniques. The algorithm and the proofs of Theorem 5 use as departure point core techniques from the proofs of Propositions 1, 3, and 2 such as Courcelle's theorem for dealing with CMSOL-sentences, the use of Gaifman's theorem for dealing with FOL-sentences, and an extended version of the *irrelevant vertex technique*, introduced by Robertson and Seymour in [67], along with some suitable version of the Flat Wall theorem which appeared recently in [53, 71] (see also [3, 70, 72, 73]). The algorithm produces equivalent and gradually "strictly simpler" instances of an annotated version of the problem. Each equivalent instance is produced in linear time and this simplification is repeated until the graph has bounded treewidth (here we may apply Courcelle's theorem, that is Proposition 1). This yields a (constructive) quadratic-time algorithm. We stress that our approach avoids techniques that have been recently used for this type of problems such as *recursive understanding* (in [1]) or the use of *important separators* (in [49]) that give worst running times in n.

Natural limitations We wish to comment on why the three basic ingredients of the definition of our logic Θ are necessary for the statement and the proof of our meta-algorithmic results.

The first ingredient of Θ is that the modulator sentences belong in $\mathsf{CMSOL}^{\mathsf{tw}}[\{\mathsf{E},\mathsf{X}\}]$ which is defined so that the treewidth of $\mathsf{torso}(G, X)$ is bounded. While it is known that bounding the treewidth is necessary for CMSOL-model-checking [19, 58], one may ask why it is not enough to just bound the treewidth of $G[X]$. To see why this unavoidable, consider a graph G and let G' be the graph obtained from G by subdividing each edge once. Then, asking whether G is Hamiltonian, which is a well-known NP-complete problem, is equivalent to asking whether G' has a vertex set S' such that $G'[S']$ is a cycle and such that $G' \setminus S'$ is an edgeless graph, that is, a K_2-minor-free graph. Notice that, while $\mathbf{tw}(G'[S']) = 2$, $\mathsf{torso}(G', S') = G$ has unbounded treewidth.

The second ingredient of Θ is minor-exclusion, that is materialized by the conjunction with μ in the definition of Θ_0. Notice first that expressing whether a graph G contains a clique on k vertices can be done by a FOL-sentence, while the k-Clique problem is W[1]-hard [20]. Therefore, the minor-exclusion condition cannot be dropped. Moreover, even if we consider a *fixed* target FOL-sentence, it was proved in [33] that there exists a FOL-sentence σ such that checking whether a graph G has a set $S \subseteq V(G)$ with $|S| = k$ such that $G \setminus S \models \sigma$ is a W[1]-hard problem, when parameterized by k. This implies that, even for this restricted problem where the FOL-sentence σ is fixed, an algorithm running in time $f(k) \cdot n^{\mathcal{O}(1)}$ cannot be expected.

The third ingredient of Θ is the FOL demand, that is materialized by the conjunction with σ in the definition of Θ_0. This is also necessary, as otherwise we may choose some property σ not definable in FOL, such as Hamiltonicity, which is CMSOL-definable and NP-complete on planar graphs. Without the restriction that σ needs to be FOL-definable, a void modulator and a sentence μ expressing planarity would be able to model this NP-complete problem. Nevertheless, we may consider extensions of FOL in the target sentence, as done in Section 3.

2 Overview of the proof

In this section we summarize some of the main ideas involved in the proof of Theorem 5, while keeping the description at an intuitive level. We would like to stress that some of the informal definitions given in this section are deliberately *imprecise*, since providing the precise ones would result in a huge overload of technicalities that would hinder the flow of the proof. Our algorithms consider as input a general structure $\mathfrak{A}$ (not necessarily a graph), and most of the arguments in the proofs concern its Gaifman graph $G_{\mathfrak{A}}$. Dealing with general structures, besides making our results more versatile, turns out to be useful in the proofs, in particular for using tools such as the *Backwards Translation Theorem* [18, Theorem 1.40], or for extending our results to other modification operations beyond vertex removal (see the full version of the paper [31]). Since the Gaifman graph of a graph is the graph itself, in this overview we will assume for simplicity that the input of our algorithms is a graph G, instead of a general structure $\mathfrak{A}$. In Subsection 2.1 we present the general scheme of the algorithm. In Subsection 2.2 we present a simplified and illustrative setting, where the input sentence θ belongs to the fragment $\bar{\Theta}_1$. This (very) particular case of Theorem 5 is helpful to illustrate our main conceptual ideas. For a more detailed proof-overview and formal proofs (up to the general compound logic Θ considered in Theorem 5), we refer the reader to the full version of the paper [31].

2.1 General scheme of the algorithm

We use the *irrelevant vertex technique* introduced by Robertson and Seymour [67]. Our overall strategy is the "typical" one when using this technique: if the treewidth of the input graph G is bounded by an appropriately chosen function, depending only on the sentence $\theta \in \Theta$, then we use Courcelle's theorem [15] and solve the problem in linear time, using the fact that our compound logic Θ is a fragment of counting monadic second-order logic. Otherwise, we identify an irrelevant vertex in linear time, that is, a vertex whose removal produces an equivalent instance. Naturally, the latter case concentrates all our efforts and, in what follows, we sketch the main ingredients that we use in order to identify such an irrelevant vertex. In a nutshell, our approach is based on introducing a robust combinatorial framework for finding irrelevant vertices. In fact, what we find is *annotation-irrelevant flat territories*, building on our previous recent work [3, 3, 32, 70–73], which is formulated with enough generality so as to allow for the application of powerful tools such as Gaifman's locality theorem [38] or a variant of Courcelle's theorem on boundaried graphs, intuitively saying that the dynamic programming tables constructed by the proof of Courcelle's theorem are also definable in CMSOL (see [5, Lemma 3.2]).

Flat walls. An essential tool of our approach is the notion of *flat wall*, originating in the work of Robertson and Seymour [67]. Informally speaking, a flat wall W is a structure made up of (non-necessarily planar) pieces, called *flaps*, that are glued together in a bidimensional grid-like way defining the so-called *bricks* of the wall. While such a structure may not be planar, it enjoys topological properties similar to those of planar graphs, in the sense that two paths that are not routed entirely inside a flap cannot "cross", except at a constant-sized vertex set A whose vertices are called *apices*. Hence, flat walls are only "locally non-planar", and after removing apices we can apply useful locality arguments, in the sense that two vertices that are in "distant" flaps should also be "distant" in the whole graph without the apices. One of the most celebrated results in the theory of Graph Minors by Robertson and Seymour [67, 68], known as the Flat Wall theorem (see also [53, 71] for recently proved variants), informally states that graphs of large treewidth contain either a large clique minor or a large flat wall. In this article we use the framework recently introduced in [71] that provides a more accurate view of some previously defined notions concerning flat walls, particularly in [53]. Precise definitions of the concepts of *flatness pair*, *homogeneity*, *regularity*, *tilt*, and *influence* can be found in the full version of this article [31] and we stress that they are *not* critical in order to understand the main technical contributions of the current article (however, they are critical for their formal correctness). In what follows, when considering a flat wall W with an apex set A in a graph G, for simplicity we refer to W by using indistinguishably the terms "wall" and "compass of a wall", which can be roughly described as the component containing W in the graph obtained from G by removing A and the "boundary" of W.

Working with an annotated version of the problem. We start by defining a convenient equivalent version of the problem, by replacing our sentence $\theta \in \Theta$ with an equivalent *enhanced* sentence $\theta_{\mathsf{R},\mathbf{c}}$. This is done in two steps, as we explain in the following two paragraphs.

Assuming the existence of a flat wall and an apex set in our input graph G, we first transform the question θ on G to a question on a structure obtained from G by "neutralizing" the apex set (see the full version of the paper [31]). The goal of this step is to ask the final FOL-sentences σ of our sentence θ in a "flattened" structure, where apices can no longer "bring close" any distant parts of the wall. This transformation of the problem, which we call *apex-projection*, will allow for the application of the locality-based strategy discussed in the definition of the *in-signature* of a wall in Subsection 2.2. To do this, we introduce some additional constant symbols $\mathbf{c}$ to our vocabulary that will be interpreted as the apex vertices.

The second step consists in defining an equivalent *annotated* version of the problem in order to deal with the FOL-sentences of θ, inspired by the approach of [32]. To do so, we introduce a vertex set $R \subseteq V(G)$, and require, for each FOL-sentence σ of θ, that the vertices interpreting the variables of (the equivalent Gaifman sentence of) σ belong to the annotated set R. We prove that the initial sentence θ and the obtained sentence, denoted by $\theta_{\mathsf{R},\mathbf{c}}$ and called an *enhanced sentence*, are equivalent for *any* choice of the apex set interpreting $\mathbf{c}$ and when R is interpreted as the whole vertex set of the graph. This independence of the choice of the apex set is strongly used in the proofs since, as discussed below, we will consider a number of different flat walls, each of which associated with a different apex set.

Our algorithms will work with the enhanced sentence $\theta_{\mathsf{R},\mathbf{c}}$. Starting with the input graph G with $V(G)$ as the annotated set R, we will create successive equivalent annotated instances, in which vertices from G are removed and such that the annotated set R is only reduced.

Zooming inside a flat wall. Our next step is to find, in G, a large flat wall W_0 to work with. The definition of our logic Θ implies that models of θ exclude a fixed complete graph K_c as a minor, where c depends only on θ. Therefore, we can apply the algorithmic version of the Flat Wall Theorem [72, Proposition 10] (see also [53, 67, 71]) to the input graph G and, assuming that the treewidth of G is large enough, we can find in linear time a flat wall W_0 and an apex set A in G such that the height of W_0 is a sufficiently large function of θ. Moreover, another crucial property guaranteed by this algorithm is that the treewidth of W_0 is bounded from above by a function of θ. This will be exploited in Subsection 2.2 in order to compute the so-called *θ-characteristic* of a wall. We will now apply a series of "zooming" arguments to the wall W_0, which are illustrated in Figure 2.

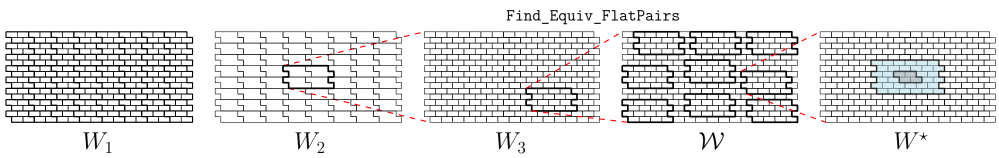

Figure 2 Sequence of walls in the general scheme of our algorithm. The first wall is obtained by applying the algorithm of [72, Proposition 14] for the wall W_0 in the input graph G.

Starting from W_0 and its associated apex set A, we apply the algorithm of [72, Proposition 14] and find, in linear time, a large (again, as a function of θ) subwall W_1 that is *λ-homogeneous*, where λ depends only on θ. The definition of a homogenous flat wall can be found in the full version of the paper [31], and roughly means that each of its bricks can route the same set of partial minors of the graphs corresponding to the minor-exclusion part of the sentence θ. We now apply the algorithm of [73, Lemma 16] to W_1, and obtain in linear time a large subwall W_2 that is *irrelevant* with respect to the minor-exclusion part of θ after the removal of a vertex set $X \subseteq V(G)$ of small enough *bidimensionality* (see Subsection 2.2). Intuitively, working "inside" W_2 allows us to "forget" the minor-exclusion part of θ in what follows. As our next step, we obtain in linear time a still large subwall W_3 of W_2 such that its associated apex set A_3 is "tightly tied" to W_3, in the sense that the neighbors in W_3 of every vertex in A_3 are spread in a "bidimensional" way.

Finding an irrelevant subwall. So far, we have found a large wall W_3 that satisfies the conditions of the above paragraph. Now, in order to identify an irrelevant vertex inside W_3, we find, inside the wall W_3, a collection $\mathcal{W}$ of pairwise disjoint subwalls, and to associate each of these subwalls with an appropriately defined *θ-characteristic* that captures its behavior with respect to the partial satisfaction of the sentence θ. Then the idea is that, if there are sufficiently many subwalls in $\mathcal{W}$ with the same θ-characteristic (called *θ-equivalent*), then some subwall in the interior of one of them can be declared *annotation-irrelevant* and this implies some progress in simplifying the current problem instance.

The above strategy allows to identify a subwall $W^\star$ inside $\mathcal{W}$ such that its central part can be removed from the annotated set R, and such that a smaller central part can be removed from G (the blue and grey subwalls in the rightmost wall of Figure 2, respectively). This is done by an algorithm, called `Find_Equiv_FlatPairs`, that is based on an appropriate definition of the *θ-characteristic* of a wall. In what follows we sketch the main ingredients and key ideas.

2.2 A simplified and illustrative setting

In order to provide some intuition, in this subsection we focus on formulas $\theta \in \Theta$ of a particular form, i.e., belonging to $\bar{\Theta}_1$, a set of formulas which we proceed to define informally in a semantical level: Given a general graph G as input, we seek for a vertex set $X \subseteq V(G)$, called *modulator*, such that, using the notation defined in the introduction, $\mathsf{stell}(G, X)$ satisfies the so-called *modulator sentence* β, and either every connected component C of $G \setminus X$, or the whole graph $G \setminus X$, satisfies the so-called *target sentence* γ, where $\gamma = \sigma \wedge \mu$ with σ being an arbitrary FOL-sentence and μ expressing the property of belonging to a proper minor-closed graph class.

Note that when $\theta \in \bar{\Theta}_1$, the target sentence γ needs to be satisfied either by each of the resulting connected components separately, or jointly by their union. We deal with this easily, by introducing a $\circ/\bullet$ -flag into the corresponding sentences that distinguishes both cases. The latter case is simpler, but in this description, in order to better illustrate our techniques, we assume the former.

Identifying the privileged component. A very useful tool in our algorithms is to identify, for every given X, a *unique* connected component among those of $G \setminus X$, which we call the *privileged component*, that contains "most" of the wall W_3. Let us formalize a bit this idea. For a positive integer q, a *pseudogrid* $\mathbf{W}_q$, is a collection of q "vertical" and q "horizontal" paths that intersect in a "grid-like" way. Note that the considered wall W_3 naturally defines a (large, as a function of θ) pseudogrid. A connected component C of a graph G is *privileged* with respect to a set $X \subseteq V(G)$ and a pseudogrid $\mathbf{W}_q$ if C is a connected component of $G \setminus X$ that contains entirely at least one vertical and one horizontal path of $\mathbf{W}_q$. It is easy to see that such a privileged component, if it exists, is unique.

Moreover, when X is a modulator, the fact that $\mathsf{torso}(G, X)$ has bounded treewidth implies that every connected component of $G \setminus X$ has a "small interface" to X and thus the flat wall W_0 (and any large subwall of it) is not significantly "damaged" by X, which we formalize via the notion of having small *bidimensionality.* Intuitively for the definition), this means that X intersects a small number of so-called "bags" of the wall. Informally, the *bags* of a wall W in a graph G with apex set A define a partition of $G \setminus A$ into connected sets, such that each bag, except the external one, contains the part of the wall W between two neighboring degree-3 vertices of the wall. This property is used extensively in the proofs and, in particular, it defines, assuming the existence of a large flat wall W_0 and a modulator X, a *unique* privileged component C in $G \setminus X$ (regardless of the $\circ/\bullet$-flag). In our sentences, in order to identify such a component, we need to integrate the "recognition" of a pseudogrid $\mathbf{W}_q$ and its associated privileged component with respect to a modulator X: it is easy to see that these properties can be defined in CMSOL.

Splitting the sentence $\theta_{\mathsf{R},\mathbf{c}}$. The existence of a privileged component C allows us to see the sentence $\theta_{\mathsf{R},\mathbf{c}}$ as a conjunction of two subsentences: one that concerns the privileged component C (where we will find the irrelevant vertex) and another one concerning the modulator X and the other (non-privileged) components of $G \setminus X$. Namely, we define a sentence $\tilde{\theta}_q$, called the *split version* of $\theta_{\mathsf{R},\mathbf{c}}$, that allows us to "break" θ into two questions: one denoted by θ_q^{out} that is the conjunction of the modulator sentence β and the target sentence γ in the non-privileged components of $G \setminus X$ and another one that concerns the target sentence γ in the privileged component C. This latter question is composed of two subsentences, namely one about the satisfaction of the FOL-sentence σ and another one about the minor-exclusion given by μ. Given this decomposition of θ into three questions (one "external" and two "internal" ones), our "irrelevancy" arguments also decompose into three parts. Concerning the "irrelevancy" for minor-exclusion, as discussed above, the fact that the whole wall W_2 is irrelevant with respect to μ allows us to focus on the other two questions. For this, we need to define the *characteristic* of a wall with respect to θ, denoted by θ-char. This characteristic is composed of two parts: the *out-signature* corresponding to the satisfiability of the sentence θ_q^{out}, and the *in-signature* corresponding to the FOL-sentence σ. Let us now explain how we define the out-signature and the in-signature, and sketch why we can eventually declare a subwall irrelevant.

Defining the out-signature of a wall. Dealing with the irrelevancy with respect to the "external" sentence θ_q^{out} turns out to be the most interesting part of the proof and we introduce several ideas which are, in our opinion, one of the main conceptual contributions of this article. The goal is, for each wall W in the collection $\mathcal{W}$, to encode all the necessary information that concerns the satisfiability of θ_q^{out} in the "non-privileged" part of the graph and the modulator X. To do this, for each $W \in \mathcal{W}$ with apex set A, we define a set of *ℓ-boundaried graphs* (i.e., graphs in which ℓ "boundary" vertices are equipped with labels), constructed as we describe below, and where ℓ depends only on θ. The boundary corresponds to where the sentence has been "split" and we need to "guess" how to complement this boundary by the part of the modulator that is not inside the wall. Note that, since θ_q^{out} is a CMSOL-sentence, by a variant of Courcelle's theorem for boundaried graphs [15], there exists a *finite* collection $\mathsf{rep}^{(\ell)}(\theta_q^{\mathsf{out}})$ of sentences on ℓ-boundaried graphs that are "representatives" of the sentence θ_q^{out} and that can be effectively constructed. We next described how these ℓ-boundaried graphs are constructed.

We observe that, using the bounded-treewidth property of the modulator sentence β, there exists a "buffer" I in W, consisting of a set of consecutive layers of the wall, which is disjoint from a hypothetical modulator X. We guess with an integer d where this "buffer" I is placed in the wall and we denote its inner part by $I^{(d)}$. This naturally induces a partition of X into X_{in} and X_{out}, with X_{in} being the part of X that is inside $I^{(d)}$. We also guess which subset of the apex set A will belong to the modulator X and we denote it by $V_L(\mathbf{a})$, where L is the set containing the indices of the corresponding apex vertices. Since parts of the "non-privileged" vertex set of the graph may lie outside the considered wall, we need to guess the part of the modulator (namely, its boundary towards the component) that lies outside the wall. More precisely, we need to guess as well which subset F' of X_{out}, other than $V_L(\mathbf{a})$, will belong to the neighborhood of the privileged component. This is achieved by guessing all ways an (abstract) graph F' with a bounded number of vertices can extend the boundary. We let F be the graph obtained from the union of $V_L(\mathbf{a})$ and F'. Finally, we also need to consider a set Z that corresponds to X_{in} together with the part inside $I^{(d)}$ that has been "chopped off" by the modulator X, that is, the part of W inside $I^{(d)}$ that will not belong to the privileged component after the removal of the modulator X. We denote by $\partial(Z)$ the set of vertices in Z that have a neighbor in $I^{(d)}$. Altogether, these guesses result in the ℓ-boundaried graph $K^{(d,Z,L,F)}$ obtained from the graph induced by $I^{(d)}$ and the set F, whose boundary is the set $\partial(Z) \cup F$.

With each such a guess (R, d, L, Z) we associate the out-signature defined as follows and denoted by $\mathsf{out\text{-}sig}$. Its elements are pairs $(\mathbf{H}, \bar{\theta})$, where $\mathbf{H}$ encodes how the set $V_L(\mathbf{a})$ in the boundary has been extended by the "abstract" graph F', and $\bar{\theta} \in \mathsf{rep}^{(\ell)}(\theta_q^{\mathsf{out}})$ prescribes the equivalence class, within the set of Courcelle's representatives mentioned above, of the considered ℓ-boundaried graph. This concludes the description of the out-signature.

While this out-signature indeed encodes the behavior of the considered wall with respect to the "external" sentence θ_q^{out}, a crucial issue has been overlooked so far: in order to be able to identify an irrelevant subwall inside the collection $\mathcal{W}$ within the claimed running time, we need to be able to *compute* the (in- and out-)signature of a wall in linear time. To do this using Courcelle's theorem, we need to consider a graph that has treewidth bounded by a function of θ. Recall that θ_q^{out} is the conjunction of the modulator sentence β (which is evaluated in the graph $\mathsf{stell}(G, X)$) and the target sentence γ in the "non-privileged" components of $G \setminus X$. It follows that the treewidth of W is bounded by a function of θ, hence the treewidth of the ℓ-boundaried "subwall" $K^{(d,Z,L,F)}$, for which we want to compute the out-signature, is also bounded by a function of θ. However, the graph $K^{(d,Z,L,F)} \setminus V(F)$ "lives" inside the *whole* privileged component C, and we cannot guarantee that the treewidth of C is bounded by a function of θ. We overcome this problem with the following trick. We observe that the satisfaction of θ_q^{out} is preserved if, instead of the whole privileged component C, we consider the graph $K^{(d,Z,L,F)}$, which is obtained by "shrinking" C to the subwall $I^{(d)}$, and which has bounded treewidth as we need. Indeed, this modification does not change any of the non-privileged components in which the target sentence γ is evaluated and, by adding edges from the "guessed extended boundary" F' to $I^{(d)}$ in order to preserve connectivity, the resulting graph $\mathsf{stell}(G, X)$ remains unchanged with this transformation, and therefore the satisfaction of the modulator sentence β is also preserved.

Defining the in-signature of a wall. To deal with the irrelevancy with respect to the FOL-sentence σ, we use arguments strongly inspired by those of [32]. The core tool here is Gaifman's locality theorem, which states that every FOL-sentence σ is a Boolean combination of basic local sentences $\sigma_1, \ldots, \sigma_p$, in the sense that the satisfaction of each σ_i depends only on the satisfaction of a set of sentences $\psi_1, \ldots, \psi_{\ell_i}$ evaluated on single vertices that can be assumed to be pairwise far apart. As discussed before, taking care of the domain of these vertices is the main reason why we consider a annotated version of the problem, corresponding to the enhanced sentence $\theta_{\mathsf{R},\mathbf{c}}$. Extending the approach of [32] (which does not deal with apices), the in-signature of a wall, denoted by $\mathsf{in\text{-}sig}$, encodes all (partial) sets of variables, one set for each basic local sentence of the so-called Gaifman sentence $\breve{\sigma}$, such that these variables lie inside an "inner part" of the wall, they are scattered in the "apex-projection" of this inner part, and they satisfy the local sentences ψ_i.

Declaring a subwall irrelevant. We now sketch the remaining of the proof for sentences in $\bar{\Theta}_1$. As mentioned above, suppose that we have already found, inside the collection $\mathcal{W}$, a large (as a function of θ) subcollection $\mathcal{W}' \subseteq \mathcal{W}$ of walls all having the same θ-characteristic.

We pick one of these walls, say $W^\star \in \mathcal{W}'$, and we declare its central part irrelevant (see Figure 2). We need to prove that, if the input graph G satisfies θ, then the graph G' obtained from G by removing the central part of $W^\star$, also satisfies θ. That is, given a modulator X in the original instance G, we need to construct another set $X' \subseteq V(G)$ that is disjoint from $W^\star$ and that is a modulator in G'. For this, we proceed as follows.

The cardinality of $\mathcal{W}'$ and the fact that X intersects few bags of the wall W_3 imply that there is a large (again, as a function of θ) subcollection $\mathcal{W}'' \subseteq \mathcal{W}'$ of walls that are disjoint from X. We take such a wall $\hat{W} \in \mathcal{W}''$ and, using the fact that $W^\star$ and $\hat{W}$ have the same θ-characteristic, we show that we can "replace" the part of the modulator X that intersects $W^\star$ with another part in $\hat{W}$, together with an alternative assignment of variables that satisfies the corresponding sentences. This results in another set X' that is a modulator in G', hence yielding the annotation-irrelevancy of (the central part of) $W^\star$.

Showing these facts is far from being easy and we need a number of technical details dealing with the irrelevancy with respect to θ_q^{out} (which incorporates β), σ, and μ. In particular, an important idea is that, changing from X to X', we obtain a new boundaried graph, which is in fact the same graph but with a new boundary. The replacement arguments for the in-signature work because of the aforementioned distance-preservation property of the apex-projection. See the full version of the paper [31] for more details.

3 From FOL to FOL+DP: the compound logic Θ^{DP}

In the definition of Θ_0, the base case of Θ, we consider compound sentences $\sigma \wedge \mu$, where $\sigma \in \mathsf{FOL}$ and μ expresses minor-exclusion. However, one can consider extensions of FOL in the compound sentences. A possible candidate is first-order logic with disjoint-paths predicates defined in [75] (see the paragraph below for a formal definition). This way we can define a more general logic Θ^{DP} and prove an algorithmic meta-theorem that encompasses also the results in [42, 43]. To ease reading, in this subsection we deal only with graphs and not with general structures. However, our results can be straightforwardly be extended to general structures. All proofs of the results of this section can be found in Section 3.

The disjoint-paths logic. We define the $2k$-ary predicate $\mathsf{dp}_k(\mathsf{x}_1, \mathsf{y}_1, \ldots, \mathsf{x}_k, \mathsf{y}_k)$, which evaluates true in a graph G if and only if there are paths $P_1, \ldots, P_k$ of G of length at least two between (the interpretations of) x_i and y_i for all $i \in [k]$ such that for every $i, j \in [k]$, $i \neq j$, $V(P_i) \cap V(P_j) = \emptyset$. We let FOL+DP be the logic obtained from FOL after allowing $\mathsf{dp}_k(\mathsf{x}_1, \mathsf{y}_1, \ldots, \mathsf{x}_k, \mathsf{y}_k), k \geq 1$ as atomic predicates.

The compound logic Θ^{DP}. We define an extension Θ^{DP} of Θ by considering, as the base case, instead of Θ_0, the logic $\Theta_0^{\mathsf{DP}} = \{\sigma \wedge \mu \mid \sigma \in \mathsf{FOL{+}DP} \text{ and } \mu \in \mathsf{EM}[\{\mathsf{E}\}]\}$.

▶ **Theorem 8.** *For every $\theta \in \Theta^{\mathsf{DP}}$, there exists an algorithm that, given a graph G, outputs whether $G \models \theta$ in time $\mathcal{O}_{|\theta|}(n^2)$.*

As we define the alternative $\tilde{\Theta}$ of Θ, we can also define $\tilde{\Theta}^{\mathsf{DP}}$ by taking $\tilde{\Theta}_0^{\mathsf{DP}} = \mathsf{FOL{+}DP}$ as the base case, i.e., by discarding the minor-exclusion from the definition of Θ_0^{DP}. Notice that $\tilde{\Theta}^{\mathsf{DP}}$ contains FOL+DP and can be seen as a natural extension of it. As a corollary of Theorem 8, we get the following analogue of Theorem 6.

▶ **Theorem 9.** *For every $\tilde{\theta} \in \tilde{\Theta}^{\mathsf{DP}}$, there exists an algorithm that, given a graph G, outputs whether $G \models \theta$ in time $\mathcal{O}_{|\theta|, \mathbf{hw}(G)}(n^2)$.*

Theorem 9 contains all results and applications of [42, 43] as a (very) special case. For a visualization of the current meta-algorithmic landscape on subgraph-closed classes, see Figure 3.

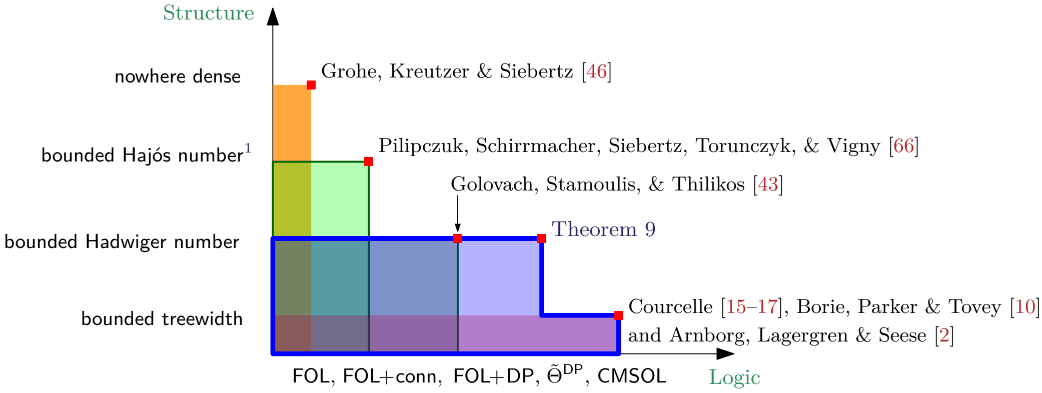

Figure 3 The current meta-algorithmic landscape on subgraph-closed classes and the position of Theorem 9 in it.

4 Further research

The minor-exclusion framework. The graph-structural horizon in both Theorem 5 and Theorem 6 is delimited by minor-exclusion. In the case of Theorem 5, this restriction is applied to the target property defined by μ in the logic Θ, while in Theorem 6 this is the promise combinatorial restriction that yields efficient model-checking for $\tilde{\Theta}$. This restriction is hard-wired in our proof in the way it combines the Flat Wall theorem with Gaifman's theorem. Recently, several efficient algorithms appeared for modification problems targeting or assuming topological minor-freeness (see [1, 36, 49] and the meta-algorithmic results in [66, 74]). For such classes, to achieve efficient model-checking for Θ, or some fragment of it, is an interesting open challenge.

Quadratic time. The proof of Theorem 5 can be seen as a possible "meta-algorithmization" of the *irrelevant vertex technique* introduced by Robertson and Seymour [67], going further than the two known recent attempts in this direction [32, 43]. The main routine of the algorithm transforms the input of the problem to a simpler graph by detecting territories in it that can be safely discarded, therefore producing a simpler instance. This routine is applied repetitively until the graph has "small" treewidth, so that the problem can be solved in linear time by using Courcelle's theorem. This approach gives an algorithm running in quadratic time. Any improvement of this quadratic running time should rely on techniques escaping the above scheme of gradual simplification. The only results in this direction are the cases of making a graph planar by deleting at most k vertices (resp. edges) in [50] (resp. [52]) that run in time $\mathcal{O}_k(n)$.

Further than connectivity closure. One of the key operations defining Θ is the connectivity extension operation, that is, given a sentence φ, to consider the (conjunctive) sentence $\varphi^{(\mathsf{c})}$. We incorporated this operation to our logic in order to express elimination distance modifications (such as those of tree-depth [12] and bridge-depth [11]) where, at each step, we remove some tree-like structure and then we apply the current target sentence to the connected components of the remaining graph. In [24], the notion of *block elimination distance* has been introduced, where the target property is applied to the biconnected components of the remaining graph (instead of the connected components). We are confident that our results can be adapted so to include the biconnectivity extension – or even the 3-connectivity

[1] The *Hajós number* of a graph G is the maximum k for which G contains K_k as a topological minor.

extension, as defined by Tutte's decomposition. However, we prefer to avoid this here as it would add undesirable burden to the statement of our results (and to the proofs as well). Another direction is to consider different versions of $\varphi^{(\mathsf{c})}$. One of them might be a *disjunctive* version, namely $\varphi^{\vee(\mathsf{c})}$, where $G \models \varphi^{\vee(\mathsf{c})}$ if *at least one* of the connected components of G is a model of φ. Another one is a *selective* version, namely $\varphi^{\exists(\mathsf{c})}$, where $G \models \varphi^{\wedge(\mathsf{c})}$ if there is some subset of the connected components of G whose union is a model of φ. Our proof fails if we wish to incorporate any of these two variants of $\varphi^{(\mathsf{c})}$ in Θ. However, it can be easily adapted so to incorporate $\varphi^{\vee(\mathsf{c})}$ in $\tilde{\Theta}$.

Descriptive complexity and the Θ-hierarchy. Recall that $\Theta = \bigcup_{i\in\mathbb{N}} \Theta_i$, where each level of the sentence set Θ_i is defined by adding an extra modulator sentence, followed by some positive Boolean combination of the connectivity closure of the lower level. We extended our result from Θ_1 to every Θ_i because Θ is quite versatile and makes it easier to express more complex hierarchical modification problems. However, it is an open problem whether this hierarchy is proper with respect to the descriptive complexity of the problems that it defines in each of its levels. In simple cases where the modulator sentence asks for a set of bounded size, and under the absence of positive Boolean combinations, it is possible to express any Θ-definable problem using Θ_1. For instance, elimination ordering to some Θ_0-definable class can be straightforwardly expressed in Θ, however with a more technical proof one can also express it in Θ_1 (see [34]). Is this collapse maintained when we consider the full expressive power of Θ? We conjecture a negative answer to this question for both Θ and Θ^{DP}.

References

1 Akanksha Agrawal, Lawqueen Kanesh, Daniel Lokshtanov, Fahad Panolan, M. S. Ramanujan, Saket Saurabh, and Meirav Zehavi. Deleting, Eliminating and Decomposing to Hereditary Classes Are All FPT-Equivalent. In *Proc. of the 32st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1976–2004, 2022. doi:10.1137/1.9781611977073.79.

2 Stefan Arnborg, Jens Lagergren, and Detlef Seese. Easy problems for tree-decomposable graphs. *Journal of Algorithms*, 12:308–340, 1991. doi:10.1016/0196-6774(91)90006-K.

3 Julien Baste, Ignasi Sau, and Dimitrios M. Thilikos. A complexity dichotomy for hitting connected minors on bounded treewidth graphs: the chair and the banner draw the boundary. In *Proc. of the 31st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 951–970, 2020. doi:10.1137/1.9781611975994.57.

4 Hans L. Bodlaender. A linear time algorithm for finding tree-decompositions of small treewidth. In *Proc. of the 25th Annual ACM Symposium on Theory of Computing (STOC)*, pages 226–234, 1993. doi:10.1145/167088.167161.

5 Hans L. Bodlaender, Fedor V. Fomin, Daniel Lokshtanov, Eelko Penninkx, Saket Saurabh, and Dimitrios M. Thilikos. (Meta) Kernelization. *Journal of the ACM*, 63(5):44:1–44:69, 2016. doi:10.1145/2973749.

6 Mikołaj Bojańczyk. Separator logic and star-free expressions for graphs, 2021. arXiv:2107.13953.

7 Édouard Bonnet, Jan Dreier, Jakub Gajarský, Stephan Kreutzer, Nikolas Mählmann, Pierre Simon, and Szymon Torunczyk. Model checking on interpretations of classes of bounded local cliquewidth. In *Proc. of the 37th Annual ACM/IEEE Symposium on Logic in Computer (LICS)*, pages 54:1–54:13. ACM, 2022. doi:10.1145/3531130.3533367.

8 Édouard Bonnet, Ugo Giocanti, Patrice Ossona de Mendez, Pierre Simon, Stéphan Thomassé, and Szymon Torunczyk. Twin-width IV: ordered graphs and matrices. In *Proc. of the 54th Annual ACM Symposium on Theory of Computing (STOC)*. ACM, 2022. doi:10.1145/3519935.3520037.

9 Édouard Bonnet, Eun Jung Kim, Stéphan Thomassé, and Rémi Watrigant. Twin-width I: tractable FO model checking. In *Proc. of the 61st IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, pages 601–612, 2020. doi:10.1109/FOCS46700.2020.00062.

10 Richard B. Borie, R. Gary Parker, and Craig A. Tovey. Automatic generation of linear-time algorithms from predicate calculus descriptions of problems on recursively constructed graph families. *Algorithmica*, 7(5-6):555–581, 1992. doi:10.1007/BF01758777.

11 Marin Bougeret, Bart M. P. Jansen, and Ignasi Sau. Bridge-depth characterizes which structural parameterizations of vertex cover admit a polynomial kernel. In *Proc. of the 47th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 168 of *LIPIcs*, pages 16:1–16:19, 2020. doi:10.4230/LIPIcs.ICALP.2020.16.

12 Jannis Bulian and Anuj Dawar. Fixed-parameter tractable distances to sparse graph classes. *Algorithmica*, 79(1):139–158, 2017. doi:10.1007/s00453-016-0235-7.

13 Jianer Chen, Fedor V. Fomin, Yang Liu, Songjian Lu, and Yngve Villanger. Improved algorithms for feedback vertex set problems. *Journal of Computer and System Sciences*, 74(7):1188–1198, 2008. doi:10.1016/j.jcss.2008.05.002.

14 Jianer Chen, Iyad A. Kanj, and Ge Xia. Improved parameterized upper bounds for vertex cover. In *Proc. of the 31st International Symposium on Mathematical Foundations of Computer Science (MFCS)*, volume 4162 of *LNCS*, pages 238–249, 2006. doi:10.1007/11821069_21.

15 Bruno Courcelle. The monadic second-order logic of graphs. I. recognizable sets of finite graphs. *Information and Computation*, 85(1):12–75, 1990. doi:10.1016/0890-5401(90)90043-H.

16 Bruno Courcelle. The monadic second-order logic of graphs III: tree-decompositions, minor and complexity issues. *RAIRO - Theoretical Informatics and Applications*, 26:257–286, 1992. doi:10.1051/ita/1992260302571.

17 Bruno Courcelle. The expression of graph properties and graph transformations in monadic second-order logic. In *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 1: Foundations*, pages 313–400. World Scientific, 1997.

18 Bruno Courcelle and Joost Engelfriet. *Graph Structure and Monadic Second-Order Logic - A Language-Theoretic Approach*, volume 138 of *Encyclopedia of mathematics and its applications*. Cambridge University Press, 2012.

19 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. doi:10.1007/s002249910009.

20 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.

21 Marek Cygan, Daniel Lokshtanov, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. Minimum bisection is fixed-parameter tractable. *SIAM Journal on Computing*, 48(2):417–450, 2019. doi:10.1137/140988553.

22 Marek Cygan, Dániel Marx, Marcin Pilipczuk, and Michal Pilipczuk. Hitting forbidden subgraphs in graphs of bounded treewidth. *Information and Computation*, 256:62–82, 2017. doi:10.1016/j.ic.2017.04.009.

23 Anuj Dawar, Martin Grohe, and Stephan Kreutzer. Locally excluding a minor. In *Proc. of the 21st IEEE Symposium on Logic in Computer Science (LICS)*, pages 270–279, 2007. doi:10.1109/LICS.2007.31.

24 Oznur Y. Diner, Archontia C. Giannopoulou, Giannos Stamoulis, and Dimitrios M. Thilikos. Block elimination distance. In *Proc. of the 50th International Workshop on Graph Theoretic Concepts in Computer Science (WG)*, volume 12911 of *LNCS*, 2002. doi:10.1007/978-3-030-86838-3_3.

25 Jan Dreier and Peter Rossmanith. Approximate evaluation of first-order counting queries. In Dániel Marx, editor, *Proc. of the 32nd ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1720–1739. SIAM, 2021. doi:10.1137/1.9781611976465.104.

26 Zdeněk Dvořák, Daniel Král, and Robin Thomas. Testing first-order properties for subclasses of sparse graphs. *Journal of the ACM*, 60(5):36:1–36:24, 2013. doi:10.1145/2499483.

27 Eduard Eiben, Robert Ganian, Thekla Hamm, and O-joung Kwon. Measuring what matters: A hybrid approach to dynamic programming with treewidth. *Journal of Computer and System Sciences*, 121:57–75, 2021. doi:10.1016/j.jcss.2021.04.005.

28 Michael R. Fellows and Michael A. Langston. Nonconstructive tools for proving polynomial-time decidability. *Journal of the ACM*, 35(3):727–739, 1988. doi:10.1145/44483.44491.

29 Samuel Fiorini, Nadia Hardy, Bruce A. Reed, and Adrian Vetta. Planar graph bipartization in linear time. *Discrete Applied Mathematics*, 156(7):1175–1180, 2008. doi:10.1016/j.dam.2007.08.013.

30 Jörg Flum and Martin Grohe. Fixed-parameter tractability, definability, and model-checking. *SIAM Journal on Computing*, 31(1):113–145, 2001. doi:10.1137/S0097539799360768.

31 Fedor V. Fomin, Petr A. Golovach, Ignasi Sau, Giannos Stamoulis, and Dimitrios M. Thilikos. Compound logics for modification problems, 2021. arXiv:2111.02755.

32 Fedor V. Fomin, Petr A. Golovach, Giannos Stamoulis, and Dimitrios M. Thilikos. An algorithmic meta-theorem for graph modification to planarity and FOL. In *Proc. of the 28th Annual European Symposium on Algorithms (ESA)*, volume 173 of *LIPIcs*, pages 51:1–51:17, 2020. doi:10.4230/LIPIcs.ESA.2020.51.

33 Fedor V. Fomin, Petr A. Golovach, and Dimitrios M. Thilikos. On the parameterized complexity of graph modification to first-order logic properties. *Theory of Computing Systems*, 64(2):251–271, 2020. doi:10.1007/s00224-019-09938-8.

34 Fedor V. Fomin, Petr A. Golovach, and Dimitrios M. Thilikos. Parameterized complexity of elimination distance to first-order logic properties. *ACM Transactions on Computational Logic*, 23(3):17:1–17:35, 2022. doi:10.1145/3517129.

35 Fedor V. Fomin, Daniel Lokshtanov, Neeldhara Misra, and Saket Saurabh. Planar $\mathcal{F}$-Deletion: Approximation, Kernelization and Optimal FPT Algorithms. In *Proc. of the 53rd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 470–479, 2012. doi:10.1109/FOCS.2012.62.

36 Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, Saket Saurabh, and Meirav Zehavi. Hitting topological minors is FPT. In *Proc. of the 52nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 1317–1326, 2020. doi:10.1145/3357713.3384318.

37 Markus Frick and Martin Grohe. Deciding first-order properties of locally tree-decomposable structures. *Journal of the ACM*, 48(6):1184–1206, 2001. doi:10.1145/504794.504798.

38 Haim Gaifman. On local and non-local properties. In *Proc. of the Herbrand Symposium*, volume 107 of *Studies in Logic and the Foundations of Mathematics*, pages 105–135. Elsevier, 1982. doi:10.1016/S0049-237X(08)71879-2.

39 Jakub Gajarský, Stephan Kreutzer, Jaroslav Nesetril, Patrice Ossona de Mendez, Michal Pilipczuk, Sebastian Siebertz, and Szymon Torunczyk. First-order interpretations of bounded expansion classes. *ACM Transactions on Computational Logic*, 21(4):29:1–29:41, 2020. doi:10.1145/3382093.

40 Archontia C. Giannopoulou, Michal Pilipczuk, Jean-Florent Raymond, Dimitrios M. Thilikos, and Marcin Wrochna. Linear kernels for edge deletion problems to immersion-closed graph classes. *SIAM Journal on Discrete Mathematics*, 35(1):105–151, 2021. doi:10.1137/18M1228839.

41 Petr A. Golovach, Dieter Kratsch, and Daniël Paulusma. Detecting induced minors in AT-free graphs. *Theoretical Computer Science*, 482:20–32, 2013. doi:10.1016/j.tcs.2013.02.029.

42 Petr A. Golovach, Giannos Stamoulis, and Dimitrios M. Thilikos. Model-checking for first-order logic with disjoint paths predicates in proper minor-closed graph classes, 2022. arXiv:2211.01723.

43 Petr A. Golovach, Giannos Stamoulis, and Dimitrios M. Thilikos. Model-checking for first-order logic with disjoint paths predicates in proper minor-closed graph classes. In *Proc. of the*

2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 3684–3699, 2023. doi:10.1137/1.9781611977554.ch141.

44 Martin Grohe. Logic, graphs, and algorithms. In *Logic and Automata: History and Perspectives, in Honor of Wolfgang Thomas*, volume 2 of *Texts in Logic and Games*, pages 357–422. Amsterdam University Press, 2008. URL: https://eccc.weizmann.ac.il/report/2007/091/.

45 Martin Grohe and Stephan Kreutzer. Methods for algorithmic meta theorems. In *Model Theoretic Methods in Finite Combinatorics - AMS-ASL Joint Special Session*, volume 558, pages 181–206. AMS, 2009. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.395.8282&rep=rep1&type=pdf.

46 Martin Grohe, Stephan Kreutzer, and Sebastian Siebertz. Deciding first-order properties of nowhere dense graphs. *Journal of the ACM*, 64(3):17:1–17:32, 2017. doi:10.1145/3051095.

47 Martin Grohe and Nicole Schweikardt. First-order query evaluation with cardinality conditions. In *Proc. of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems (PODS)*, pages 253–266. ACM, 2018. doi:10.1145/3196959.3196970.

48 Jiong Guo, Falk Hüffner, and Rolf Niedermeier. A structural view on parameterizing problems: Distance from triviality. In *Proc. of the 1st International Workshop on Parameterized and Exact Computation (IWPEC)*, volume 3162 of *LNCS*, pages 162–173, 2004. doi:10.1007/978-3-540-28639-4_15.

49 Bart M. P. Jansen, Jari J. H. de Kroon, and Michal Włodarczyk. Vertex deletion parameterized by elimination distance and even less. In *Proc. of the 53rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 1757–1769, 2021. doi:10.1145/3406325.3451068.

50 Bart M. P. Jansen, Daniel Lokshtanov, and Saket Saurabh. A near-optimal planarization algorithm. In *Proc. of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1802–1811, 2014. doi:10.1137/1.9781611973402.130.

51 Ken-ichi Kawarabayashi, Yusuke Kobayashi, and Bruce A. Reed. The disjoint paths problem in quadratic time. *Journal of Combinatorial Theory, Series B*, 102(2):424–435, 2012. doi:10.1016/j.jctb.2011.07.004.

52 Ken-ichi Kawarabayashi and Bruce A. Reed. Computing crossing number in linear time. In *Proc. of the 39th annual ACM symposium on Theory of computing (STOC)*, pages 382–390, 2007. doi:10.1145/1250790.1250848.

53 Ken-ichi Kawarabayashi, Robin Thomas, and Paul Wollan. A new proof of the flat wall theorem. *Journal of Combinatorial Theory, Series B*, 129:204–238, 2018. doi:10.1016/j.jctb.2017.09.006.

54 Eun Jung Kim, Alexander Langer, Christophe Paul, Felix Reidl, Peter Rossmanith, Ignasi Sau, and Somnath Sikdar. Linear kernels and single-exponential algorithms via protrusion decompositions. *ACM Transactions on Algorithms*, 12(2):21:1–21:41, 2016. doi:10.1145/2797140.

55 Tomasz Kociumaka and Marcin Pilipczuk. Deleting vertices to graphs of bounded genus. *Algorithmica*, 81(9):3655–3691, 2019. doi:10.1007/s00453-019-00592-7.

56 Tuukka Korhonen. A single-exponential time 2-approximation algorithm for treewidth. In *Proc. of the 62nd IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, pages 184–192, 2021. doi:10.1109/FOCS52979.2021.00026.

57 Stephan Kreutzer. On the parameterised intractability of monadic second-order logic. In *Proc. of the 18th EACSL Annual Conference on Computer Science Logic (CSL)*, pages 348–363, 2009. doi:10.1007/978-3-642-04027-6_26.

58 Stephan Kreutzer. Algorithmic meta-theorems. In *Finite and Algorithmic Model Theory*, volume 379 of *London Mathematical Society Lecture Note Series*, pages 177–270. Cambridge University Press, 2011. URL: http://www.cs.ox.ac.uk/people/stephan.kreutzer/Publications/amt-survey.pdf.

59 Dietrich Kuske and Nicole Schweikardt. First-order logic with counting. In *Proc. of the 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–12. IEEE Computer Society, 2017. doi:10.1109/LICS.2017.8005133.

60 John M. Lewis and Mihalis Yannakakis. The Node-Deletion Problem for Hereditary Properties is NP-Complete. *Journal of Computer and System Sciences*, 20(2):219–230, 1980. doi:10.1016/0022-0000(80)90060-4.

61 Dániel Marx and Igor Razgon. Fixed-Parameter Tractability of Multicut Parameterized by the Size of the Cutset. *SIAM Journal on Computing*, 43(2):355–388, 2014. doi:10.1137/110855247.

62 Dániel Marx and Ildikó Schlotter. Obtaining a planar graph by vertex deletion. *Algorithmica*, 62(3-4):807–822, 2012. doi:10.1007/s00453-010-9484-z.

63 Jaroslav Nesetril and Patrice Ossona de Mendez. *Sparsity - Graphs, Structures, and Algorithms*, volume 28 of *Algorithms and combinatorics*. Springer, 2012. doi:10.1007/978-3-642-27875-4.

64 Jaroslav Nesetril, Patrice Ossona de Mendez, Michal Pilipczuk, Roman Rabinovich, and Sebastian Siebertz. Rankwidth meets stability. In *Proc. of the 32nd ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2014–2033. SIAM, 2021. doi:10.1137/1.9781611976465.120.

65 Jaroslav Nesetril, Patrice Ossona de Mendez, and Sebastian Siebertz. Structural properties of the first-order transduction quasiorder. In *Proc. of the 30th EACSL Annual Conference on Computer Science Logic (CSL)*, volume 216 of *LIPIcs*, pages 31:1–31:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.CSL.2022.31.

66 Michal Pilipczuk, Nicole Schirrmacher, Sebastian Siebertz, Szymon Torunczyk, and Alexandre Vigny. Algorithms and data structures for first-order logic with connectivity under vertex failures. In *Proc. of the 49th International Colloquium on Automata, Languages, and Programming, (ICALP)*, volume 229 of *LIPIcs*, pages 102:1–102:18, 2022. doi:10.4230/LIPIcs.ICALP.2022.102.

67 Neil Robertson and Paul D. Seymour. Graph minors. XIII. The disjoint paths problem. *Journal of Combinatorial Theory, Series B*, 63(1):65–110, 1995. doi:10.1006/jctb.1995.1006.

68 Neil Robertson and Paul D. Seymour. Graph minors. XX. Wagner's conjecture. *Journal of Combinatorial Theory, Series B*, 92(2):325–357, 2004. doi:10.1016/j.jctb.2004.08.001.

69 Ignasi Sau and Uéverton dos Santos Souza. Hitting forbidden induced subgraphs on bounded treewidth graphs. *Information and Computation*, 281:104812, 2021. doi:10.1016/j.ic.2021.104812.

70 Ignasi Sau, Giannos Stamoulis, and Dimitrios M. Thilikos. An FPT-Algorithm for Recognizing k-Apices of Minor-Closed Graph Classes. In *Proc. of the 47th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 168 of *LIPIcs*, pages 95:1–95:20, 2020. doi:10.4230/LIPIcs.ICALP.2020.95.

71 Ignasi Sau, Giannos Stamoulis, and Dimitrios M. Thilikos. A more accurate view of the Flat Wall Theorem, 2021. arXiv:2102.06463.

72 Ignasi Sau, Giannos Stamoulis, and Dimitrios M. Thilikos. k-apices of minor-closed graph classes. II. Parameterized algorithms. *ACM Transactions on Algorithms*, 18(3), 2022. doi:10.1145/3519028.

73 Ignasi Sau, Giannos Stamoulis, and Dimitrios M. Thilikos. k-apices of minor-closed graph classes. I. Bounding the obstructions. *Journal of Combinatorial Theory, Series B*, 161:180–227, 2023. doi:10.1016/j.jctb.2023.02.012.

74 Nicole Schirrmacher, Sebastian Siebertz, Giannos Stamoulis, Dimitrios M. Thilikos, and Alexandre Vigny. Model checking disjoint-paths logic on topological-minor-free graph classes, 2023. arXiv:2302.07033.

75 Nicole Schirrmacher, Sebastian Siebertz, and Alexandre Vigny. First-Order Logic with Connectivity Operators. In *Proc. of the 30th EACSL Annual Conference on Computer Science Logic (CSL)*, volume 216 of *LIPIcs*, pages 34:1–34:17, 2022. doi:10.4230/LIPIcs.CSL.2022.34.

76 Detlef Seese. Linear time computable problems and first-order descriptions. *Mathematical Structures in Computer Science*, 6(6):505–526, 1996. doi:10.1017/S0960129500070079.