Optimality Program for Algorithms and Logics

Summary of the project. This project develops a coherent research program around two foundational questions on the limits of efficient computation, at the interface of algorithms, logic, and graph structure. It explores the limits of efficient algorithms, and asks how expressive a logic can be while retaining tractability.

Our first goal is to achieve a quantitative understanding of the provably optimal algorithms for fundamental computational problems, and especially for problem classes definable in **logical frameworks**. This is the **optimality program for algorithms**. Our focus is on graph algorithms, through the lens of **parameterized and approximation algorithms**, while not limited to these paradigms. This mobilizes, on the one hand, classic and emerging algorithmic tools, and on the other hand, recent developments in fine-grained complexity.

The second goal is to establish how general a logic can be while retaining efficient algorithms for the class of problems it defines. This is the **optimality program for logics**. The methodological ingredients are largely the same, except that logic plays a more central role. As we explore various classes of graphs, our goals go hand-in-hand with a deeper understanding of **structural graph theory**.

Both goals cut across our three tasks, each subdivided into two subtasks. Task 1 develops **algorithmic meta-theorems**, such as Courcelle's theorem, in a fine-grained setting, with subtask 1.1 focusing on the second goal (optimality program for logics) and subtask 1.2 on the first (optimality program for algorithms). Tasks 2 and 3 apply our program to the two main directions beyond the celebrated graph minor theory (relevant to subtask 1.1). Task 2 concerns **classes excluding an induced minor**, which have recently gained prominence but remain poorly understood. Task 3 addresses **new width parameters**, such as twin-width and merge-width, and **monadically dependent classes**, which form the broadest family conjectured to admit efficient first-order model checking, and beyond which the problem is known to become intractable.

The tasks jointly lay the foundations for a **modern logic-based theory of provably tight graph algorithms**. The consortium brings together the right range of expertise to tackle this ambitious challenge, and is already at the forefront of the relevant areas: algorithmic model theory, approximation and parameterized algorithms, fine-grained complexity, and new width parameters.

Organization. As a slight modification of the proposal template, we merge "I. a. Objective" and "I. b. Position" into Section 1, and we move "I. c. Methodology" to a separate **??**. This is because, due to the nature of our research, the objectives and state of the art are inseparable, and because we wish to avoid overly deep nesting of sections and subsections. The rest of the structure follows the template.

1 Context, positioning and objectives of the project

Tracing the **limits of efficient computation** is one of the core foundational goals of theoretical computer science. This project proposes to advance our understanding of this key topic by using tools from **logic**, **graph structure**, **approximation**, and **fine-grained** and **parameterized complexity**. The interplay between efficient computation, logic, and graph structure is a well-developed area, exemplified by landmark results such as Courcelle's theorem. A standard approach is to identify a **class of problems** definable in a given logic, a **class of inputs** characterized by parameters such as treewidth, and an algorithm that solves **all** problems in the former class on every input in the latter. Results of this kind, known as **algorithmic meta-theorems**, unify problem-specific techniques, yield broad tractability results, and help delineate the boundaries of efficient computation.

This project aims to open new directions in this area by (i) employing tools that have so far seen limited use in this context, notably **fine-grained complexity** and **approximation**; (ii) investigating directions motivated by recent developments, notably the theory of **induced minors** and new **graph width measures**; and (*iii*) leveraging the leading and complementary **expertise** of the partners across these topics. We organize our investigation around a series of basic questions:

1. Can we refine and extend the theory behind classic algorithmic meta-theorems? The current landscape is framed in coarse dichotomies (FPT versus intractability, or FO versus MSO), but these hide important tradeoffs between algorithmic efficiency, expressive power, and generality of the input classes. We ask two main questions in this direction. First, can we pinpoint the **most general** logics that still admit tractable algorithms on given classes of inputs? Second, can we **precisely** characterize the **best possible** complexity guarantees achievable for problems definable in these logics? We pursue these two directions in subtask 1.1 and subtask 1.2, respectively. Together, they form Task 1, whose goal is to initiate an **optimality program** for algorithms and logic that we further develop in tasks 2 and 3.

2. How can we best leverage **induced-minor theory** algorithmically? Our focus on **induced** minors is motivated both by the historical role of graph minors in the study of sparse classes and by recent progress suggesting that induced minors may hold a deep and fruitful structural and algorithmic theory. We devote Task 2 to the investigation of meta-theorem techniques for the induced-minor relation. Task 3 explores the other main (far-reaching) generalization of minor-free classes: monadically dependent classes.

3. What are the limits of efficient computation for **new graph widths** such as twin-width and merge-width, which lie at the frontier of tractability for FO logic? In Task 3, we investigate **fine-grained complexity** and **approximation** questions for these recent but highly promising structural parameters, and more generally for **monadically dependent classes** (and beyond). Our guiding perspective is meta-theoretic: such questions are tightly connected to the advancement of algorithmic meta-theorems for FO and its variants, and should be studied in tandem with them.

All in all, the tasks outlined above draw the broad contours of an optimality program: we are looking to move far beyond the state of the art by looking for meta-theorems that are as general as possible in the problems they cover (subtask 1.1) and in the types of input structure they build on (Tasks 2 and 3); and for which any complexity-theoretic barriers to further progress can be clearly articulated (subtask 1.2).

1.1 Task 1: Optimality program in algorithmic meta-theorems

The current state-of-the-art in algorithmic meta-theorems lacks in two principal aspects: the understanding of the different logical and structural conditions that induce meta-algorithmic settings as well as tight algorithmic complexities. These two **fine-grained aspects** are deeply interconnected and shape the main theme of this task. We start with some basic background on logics and meta-theorems.

The logics FO and MSO. Our focus is on logics on graphs that are undirected, without multiple edges or loops. Arguably the main logic (on graphs) is *first-order logic* (FO). Formulas of first-order logic have (vertex) variables corresponding to vertices of the graph and are built by closing atomic formulas of the form $x = y$ and $E(x, y)$ (where the symmetric and irreflexive relation $E(\cdot, \cdot)$ encodes the edge set) under Boolean connectives ($\vee, \wedge, \neg$) and existential and universal quantification $\exists x$ and $\forall x$. In FO, one can express graph properties like the existence of a fixed-size subgraph, independent set, dominating set, etc; e.g., the existence of a dominating set of size at most k can be expressed by the FO formula $\exists x_1 \dots \exists x_k \forall y \bigvee_{i=1}^{k} \big((x_i = y) \vee E(x_i, y)\big)$. The FO logic can only express *local properties*; e.g., the existence of a path (of unbounded length) joining two fixed vertices is not definable in FO.

A logic with significantly larger expressive power than FO is *monadic second-order logic* (MSO). This logic extends FO by allowing the existential and universal quantification over *set variables* corresponding to *subsets of vertices* and the extra atomic formula $x \in X$ for a vertex variable x and set variable X. The ability to quantify over sets (of vertices) allows MSO to go well beyond expressiveness of FO and encode many global graph properties like connectivity or planarity.

The CMSO_2 logic extends MSO in two directions: one can quantify over edge subsets, and there is a vertex-edge incidence predicate (this is the 2), and counting predicates are added, allowing constraints of the form $|X| \equiv i \pmod p$ for fixed integers $p \geqslant 2$ and $0 \leqslant i < p$ (this is the C).

Algorithmic meta-theorems for FO and MSO. In the field of parameterized complexity, a prototypical *algorithmic meta-theorem* asserts that for a logic $\mathcal{L}$ and a graph class $\mathcal{C}$, the *model-checking problem*, that is, deciding whether a given graph $G \in \mathcal{C}$ satisfies a given formula $\varphi \in \mathcal{L}$, denoted by $G \models \varphi$, can be solved in time $\mathcal{O}_{|\varphi|,\mathcal{C}}(1) \cdot |V(G)|^{\mathcal{O}(1)}$; such an algorithm is called an *efficient model checking*, and we also say that $\mathcal{C}$ is *tractable* (for $\mathcal{L}$). (We use the notation $\mathcal{O}_y(g(x))$, where y can be a tuple of variables, for at most $f(y)g(x)$ for some function f.) More generally, it is any algorithm that works for every problem definable in some logic, rather than only for some specific problem.

The most classic algorithmic meta-theorem is the celebrated *Courcelle's theorem* [24], which states that bounded-treewidth classes are tractable for MSO (even for CMSO_2); this has been extended to bounded cliquewidth for CMSO [25]. As for FO, the quest for classes beyond bounded treewidth that admit efficient model checking has been an active and fruitful research direction which has recently enjoyed major successes [39, 20, 19, 28, 30]; see Task 3.

1.1.1 Algorithmic meta-theorems for FO extensions

Recently there has been a shift in studying extensions of FO on graphs, aiming to express problems that are not FO-definable but can still give algorithmic meta-theorems beyond classes of bounded treewidth or cliquewidth. An illustrative example is FO+conn (or *separator logic*) introduced in [7, 53], which is an extension of FO allowing the use of the *connectivity predicate* $conn_k(x, y, z_1, \ldots, z_k)$ which evaluates to true if there is a path in the graph from x to y that avoids $z_1, \ldots, z_k$. This logic was shown to admit efficient model checking on topological-minor-free classes [49]; these classes essentially forbid the existence of a fixed number of vertices that are pairwise connected via internally vertex-disjoint paths (generalizing the notion of minors). Subsequent works focused on getting further extensions of FO and FO+conn and showing efficient model checking for the majority of them [52, 32, 51, 8].

Our principal goal is to *systematically explore* the tractability frontier of the model checking problem for different logics on restricted graph classes. The general driving question is the following:

Meta-question 1. *For every tractable class $\mathcal{C}$, what predicates can enrich FO while preserving efficient model checking on $\mathcal{C}$?*

We plan to study this question through two different ways of extending FO: by adding predicates, and by allowing quantification over restricted sets. We will also pursue the complementary perspective: for selected extensions of FO, we will ask which graph classes form the most general tractable family. We believe that both viewpoints can yield fruitful insights into the connections between graph classes and logics.

Logics for sparse classes. Minor-free classes are arguably among the most studied classes in algorithmic and structural graph theory. Recent works have shed light on *logically* understanding several structural and algorithmic aspects of Graph Minors [38, 32, 51]. However, most of the obtained meta-theorems have been extended to topological-minor-free classes [52, 50]. Therefore, it is currently unclear whether there is a logic whose tractability is delimited by minor-freeness.

To answer this question, one may consider tweaking FO+conn. For example, *color-connectivity logic* is a variant of FO+conn on colored graphs, where the connectivity predicate now requires the path between x and y to be monochromatic.

Question 1. *Does color-connectivity logic admit efficient model checking on minor-free classes?*

Recent work on minor-free classes [22] has strong potential to lead to a positive answer. However, the logic is expressive enough that such an algorithm beyond minor-free classes would be surprising.

Another approach, which considers logics with restricted set quantifications, was introduced in [51]. There, set quantification is allowed on sets that forbid certain routing behaviour through them. *Low flow MSO* is the MSO fragment where quantification is allowed only on sets X for which the maximum number of pairwise-disjoint paths in G with both endpoints in X is at most some fixed constant.

Question 2. *For which tractable classes for FO can we extend tractability to low flow MSO?*

This was already done for minor-free classes [51] and extended to topological-minor-free classes [50]. The principle of restricted set quantification suggested in [51] yields a template for various extensions of FO (or rather, MSO fragments), via different parameters used on vertex subsets. We believe that this template could provide *the* logic for (some of) the graph classes arising in sparsity theory [46].

Dense analogues of (topological-)minor-free classes. As explained in Task 3, the study of the tractability limits of FO gave exciting new developments in the *dense* counterpart of sparsity theory [46]. This line of research led to introducing *low rank MSO* [8], a fragment of MSO in which set quantification is restricted to vertex sets of bounded cutrank (i.e., with a bounded number of adjacency profiles). This logic was shown to be equivalent to FO+conn on *weakly sparse classes* [8] (i.e. forbidding certain complete bipartite graphs as subgraphs), therefore inheriting efficient model checking in the sparse setting. The emerging challenge is to study tractability of this logic on dense classes.

Question 3. *In which dense classes does low rank MSO admit efficient model checking?*

Since this logic is a fragment of MSO and due to the results in [25], the interesting cases are *hereditary* (i.e., closed under vertex deletion) classes of unbounded cliquewidth. Motivated by the connection between FO+conn and low rank MSO, we conjecture that a tight answer to the above question is given by the dense version of topological-minor-freeness. Decompositions based on bounded rank cuts and notions related to *rankwidth* [47] are expected to be of key importance in the study of the above question, with their own structural and algorithmic interest.

Logics for induced minors. Another regime that we aim to work on is that of *induced minors*, the subject of Task 2, which are graphs obtained after a series of vertex deletions and edge contractions (but unlike minors, edge deletions are disallowed). While the questions related to their structural and algorithmic theory will be discussed later, here we would like to mention one main difference: while detecting a minor can be done in almost-linear FPT time [43], there are fixed graphs (even trees [42]) that are NP-hard to find as induced minors in general graphs. Motivated by the challenging problem of identifying which classes allow a fast detection of induced minors, we plan to study logics expressing induced minor containment and algorithmic meta-theorems thereof (also see subtask 2.2).

From a logical viewpoint, induced minor containment is very similar to minor containment: it is not expressible in bare FO. In the case of minors (or rather topological minors), this shortcoming led to *disjoint-paths logic* (FO+DP), introduced in [53]. This logic extends FO by allowing the use of the disjoint-paths predicate $\mathrm{dp}_k(u_1, v_1, \ldots, u_k, v_k)$ which holds if there are k vertex-disjoint paths $P_1, \ldots, P_k$ such that P_i is a u_i–v_i path for every i. In order to get a finer control over the distances between the paths, FO+DP was further extended [38] to allow fixing a minimum distance between the paths, giving the *scattered disjoint-paths logic* (FO+SDP). Combining the predicates of FO+DP and FO+SDP to a single more general predicate, one can express the induced minor containment.

When it comes to tractability, for FO+DP we have a full characterization in subgraph-closed classes, given by topological-minor-freeness [52]. For FO+SDP, tractability is established only up to bounded-genus graphs [38]. Still, tractability of FO+SDP and its proposed extension can, a priori, go beyond classes of bounded genus, leaving room for classes where induced-minor containment is tractable.

Question 4. *Does (the extended) FO+SDP admit efficient model checking on minor-free classes?*

The difficulty in settling this question comes from lifting the rerouting arguments in [38] to work on almost-embeddable graphs. In the same spirit, one may consider the scattered variant of FO+conn by modifying the connectivity predicate to ask for a path P connecting x and y with no edge between P and the vertices $z_1, \dots, z_k$.

Question 5. *Which classes are tractable for the scattered variant of FO+conn?*

We are interested in this question for several reasons. First, as for their non-scattered variants, we expect these two logics to enjoy similar meta-theorems. Understanding the tameness (in terms of graph classes) of the scattered FO+conn seems to be the major step toward grasping the corresponding tameness of FO+SDP. Also, studying the scattered notion of connectivity defined by this logic is of strong independent interest (see e.g. the emerging *coarse graph theory* [37]). This logic also seems to be connected with *vertex minors*, another "minor-like" notion of containment that is well-suited for dense graphs [21]. Identifying the extension of FO that allows expressing vertex-minor containment and studying meta-theorems about this logic is another ambitious goal that we would like to carry out.

Parametric dependence of meta-theorems. Apart from the logical and combinatorial dimension of meta-theorems, there is a third dimension: the one of efficiency. While most meta-theorems come with an immense dependence on the size of the formula in their running times, we would like to further understand whether and when it is possible to get improved bounds.

A first level of distinction is between *elementary* and *non-elementary* dependences, where the former translates to *fixed* (independent of φ) towers of exponentiation. In subgraph-closed classes, elementary FO model checking is fairly understood [33], while for hereditary classes this is still open. We believe that there is more at stake than answering this challenging question. In particular, we would like to track elementary running times in different meta-algorithmic regimes.

Question 6. *Is there a variant of twin-width that characterizes elementary FO model checking on hereditary classes of ordered graphs?*

Adopting a more fine-grained approach, we are also interested in going further than the distinction between elementary and non-elementary running times, by focusing on specific parametric dependences, say single or double exponential.

Meta-question 2. *Which logic admits single-exponential FPT model checking on a given class $\mathcal{C}$?*

The ultimate goal here is to find for some fixed class $\mathcal{C}$, a logic that *characterizes* model checking on $\mathcal{C}$ within certain running times, e.g. single-exponential FPT. A meta-theorem of this kind is the one of Pilipczuk [48], which shows that for bounded treewidth graphs the famous Cut&Count technique is captured by a variant of modal logic called *existential counting modal logic* (ECML).

1.1.2 Fine-grained complexity of meta-theorems

One of the main ingredients of this project is the use of **fine-grained** computational complexity theory. Whereas traditional complexity theory uses standard conjectures (e.g. $\mathrm{P} \neq \mathrm{NP}$) to prove qualitative lower bounds for various problems (e.g. that Independent Set cannot be solved in polynomial time), fine-grained complexity relies on more precise hypotheses to obtain **quantitative** lower bounds (e.g. that Independent Set cannot be solved in time $2^{o(n)}$ under the ETH or in time $1.99^{\mathrm{tw}} n^{\mathcal{O}(1)}$, where tw is the graph's treewidth, under the SETH). The first application of this framework in our context will be in proving precise lower bounds on the complexity of both specific problems and model-checking (that is, meta-theorems). In other words, we intend to tackle questions of the following form:

Meta-question 3. *Can we prove the optimality of our meta-theorems by showing tight lower bounds under fine-grained complexity hypotheses such as (S)ETH?*

This question is of clear interest to our project: the development of algorithmic meta-theorems is our main objective and we want to know whether our algorithms are optimal. In particular, the investigation into cases allowing an elementary dependence on the parameters, such as in Question 6, can already be seen as an embryonic type of "fine-grained" question, as we are aiming for a more quantitative bound than fixed-parameter tractability. We want to push such questions further and try to distinguish not just between elementary and non-elementary, or double-exponential and single-exponential dependencies, but also try to determine specific constants in cases where single-exponential algorithms are possible. In particular, the meta-theorem of Pilipczuk about ECML mentioned previously unifies numerous algorithmic results for treewidth that lead to single-exponential FPT algorithms and for many such problems **tight and precise** bounds are known under the SETH. Can we extend these problem-specific bounds to general statements about ECML-expressible problems?

Question 7. *Can we formulate an improved version of Pilipczuk's meta-theorem and prove its optimality under the SETH?*

Pilipczuk's meta-theorem gives algorithms of the form $c^{\mathrm{tw}} n^{\mathcal{O}(1)}$, where tw is the treewidth and c depends on the formula describing the problem. The goal would be to obtain **syntactic** conditions on the formula allowing us to determine the constant c as tightly as possible, that is, obtain a theorem which based on analyzing the formula (quantifiers used, depth, width) gives a lower bound on the complexity of the corresponding problem. Of course, this type of question will be asked not only for this particular algorithmic meta-theorem, but for most of the meta-theorems of this project.

In addition to the direction discussed above, we strongly believe that this type of investigation into the interplay between fine-grained hypotheses and logic will also be fruitful in the converse direction, in the same way that descriptive complexity theory has been a key tool in the development of traditional complexity theory. In this sense, we intend to use fine-grained complexity theory in a more ambitious way in the investigation of algorithmic meta-theorems, as explained below.

In order to better understand our goal, it is instructive to think about both the current state of the art in fine-grained complexity and the historical interplay between logic and complexity theory. On the one hand, fine-grained complexity theory is by now a rich field which, however, suffers from a proliferation of different complexity hypotheses of varying plausibilities. Besides the obvious drawback of this situation (we are not sure which hypotheses are more credible), this renders the field much more chaotic than traditional complexity theory, because rather than having problems neatly organized into **equivalence classes**, we have a **web of reductions** which sometimes reproves the same bounds under different assumptions. It would clearly be of benefit to better organize this web and understand how fine-grained complexity bounds can also be organized into equivalence classes.

Our approach is based on the belief that logic is an effective remedy for this conundrum. In the same way that the classes of traditional complexity theory can be rediscovered through the lens of logic, as exemplified by the field of **descriptive complexity**, we believe that logical formalisms will allow us to discover some basic classes of fine-grained complexity. By studying the fine-grained complexity of model-checking, we will obtain a two-fold payoff: (*i*) we will achieve lower bounds assuring us that our meta-theorems are optimal and (*ii*) we will be able to pinpoint the class of fine-grained questions which are **equivalent** to each hypothesis by giving a characterization through logical expressibility.

The approach we propose above has not yet been much explored, but some early results exist which indicate that it has strong potential. In particular, in [34] it is shown that the k-OV hypothesis (stating that k-Orthogonal Vectors cannot be solved in time $n^{k-\varepsilon}$) is **equivalent** to the absence of an algorithm that model checks FO formulas with $k+1$ quantifiers in time $m^{k-\varepsilon}$, where m is the size of the input. This type of result is **foundational**: not only does it justify that the (easy) $\mathcal{O}(m^k)$-time FO model-checking algorithm is optimal, but it also shows that the k-OV hypothesis captures the complexity of **all** problems expressible in FO logic with $k+1$ quantifiers, thus building a complexity-equivalence class.

A recent development which renders this direction particularly promising is the emergence of fine-grained hypotheses tailored to parameterized complexity, such as the pw-SETH and tw-SETH [44], which state that SAT cannot be solved in time $(2-\varepsilon)^{\mathrm{w}} n^{\mathcal{O}(1)}$, where w is respectively the pathwidth or treewidth of the primal graph of the formula. These hypotheses succeed in building an equivalence class of fine-grained lower bounds for fundamental graph problems (Independent Set, Dominating Set, Coloring), when parameterized by pathwidth or treewidth, as well as (linear) clique-width. A question we would like to tackle is whether these hypotheses have a logical characterization, and we believe that the correct answer is a fragment of FO logic where we bound the number of variables (rather than the number of quantifiers). We will therefore attempt to establish the following conjecture:

Conjecture 1. *The tw-SETH is false if and only if there exists $\varepsilon > 0$ and an algorithm which takes as input a structure of size m and an existential FO sentence φ with $k+1$ variables and decides if $G \models \varphi$ in time $m^{k-\varepsilon}|\varphi|^{\mathcal{O}(1)}$. Furthermore, there exists an algorithm that solves this problem in time $m^k|\varphi|^{\mathcal{O}(1)}$.*

If true, the conjecture above can be seen as a generalization of the result of [34] because the tw-SETH is a consequence of the k-OV hypothesis and FO sentences with k quantifiers can only have at most k variables. To be clear, the same variables are allowed to be quantified multiple times, and the size of φ may depend on m; in particular, connectivity can be expressed with three variables (by a long sentence). This direction is then related to the variants of FO studied in subtask 1.1, such as FO+DP. We note here that if an equivalence such as the one above can be established, this would imply that for a specific FO model-checking problem we also obtain fine-grained hardness under the k-SUM hypothesis, following recent results [45].

More broadly, we plan to investigate numerous questions about the complexity of FO model checking in this spirit. The link with fine-grained hypotheses will allow us to both obtain tight lower bounds for our model-checking algorithms and new characterizations of equivalence classes of problems for each hypothesis, making the first steps towards a new **fine-grained descriptive complexity theory**.

1.2 Task 2: Optimality program for induced-minor-free classes

The theory of classes excluding a fixed minor is one of the central achievements of structural graph theory; some of it is mentioned in subtask 1.1. Over the past three decades, approximation and subexponential algorithms have been developed for NP-hard problems in these classes, and, more generally, algorithms with better performance guarantees than on general graphs. In parallel, our structural understanding has become more refined. This has given rise to a beautiful theory with concrete applications. Its scope, however, remains limited.

Sparsity theory is a prominent offshoot of minor theory, featuring successive generalizations through polynomial expansion, bounded expansion, nowhere denseness, and monadic stability. This line of work has been particularly active over the last fifteen years. Yet even the most general family of monadically stable classes does not cover very simple graphs known as half-graphs. Task 3 addresses the dense generalization of this program: monadically dependent classes.

We focus here on an extension of minor theory that is incomparable with the sparsity program: **understanding classes excluding a fixed *induced* minor**. Unlike the classes of the sparsity program, induced-minor-free classes encompass intersection graph classes (which are of particular interest to many members of the consortium), such as interval graphs, disk graphs, segment graphs, or string graphs.

The study of induced-minor-free graphs has attracted growing attention in recent years. A wealth of conjectures and partial results points to the existence of a deep underlying theory that remains to be uncovered. The aim of this task is to lay its foundations and resolve its central conjectures.

1.2.1 Planar induced minors

There is something special in excluding a *planar* graph as an induced minor. Classes excluding a planar graph as a *minor* have bounded treewidth, so on them, every CMSO_2-definable problem can be solved in polynomial time. This cannot be expected from classes excluding a planar induced minor; for instance, Dominating Set is NP-hard on graphs excluding the 5-vertex path as an induced minor. However, the question is wide open in the case of Maximum Independent Set (MIS). This is a central conjecture that has given rise to dozens of partial results.

Conjecture 2 (**Refuted**, Dallard–Milanič–Štorgel (DMS) [26]). *For every planar graph H,* Maximum Independent Set *can be solved in polynomial time on graphs excluding H as an induced minor.*

Resolved negatively: Maximum Independent Set remains NP-hard in graphs excluding the 5×5 grid as an induced minor [12]. The conjecture remains open if further the graphs are weakly sparse.

(In what follows, K_t denotes the t-vertex clique, $K_{s,t}$ the biclique with s vertices against t vertices, P_t the t-vertex path, C_t the t-vertex cycle, W_t the wheel obtained from C_t by adding a universal vertex.) Conjecture 2 was shown when H is K_5 minus an edge, $K_{2,t}$, W_4 [26], P_6 [40], any windmill graph (a matching plus a universal vertex) [15]. Our intuition is that Conjecture 2 fails for some planar graph H. There are, however, weakenings of DMS: namely, q-DMS and qptas-DMS, asserting the same with quasipolynomial running time and a quasipolynomial-time approximation scheme (QPTAS), respectively.

We believe that qptas-DMS should hold, whereas we are less certain about q-DMS. Nevertheless, q-DMS has many more resolved cases than DMS. For instance, DMS is open when $H = P_7$, whereas q-DMS is known when H is any path or any cycle [36], a disjoint union of triangles [9] and even $tC_3 \uplus C_4$ [15]. Very recently, qptas-DMS was further shown when H is any graph of maximum degree 2 (i.e., disjoint union of paths and cycles) [13]. If indeed q-DMS holds but DMS does not, one will not be able to show it via an NP-hardness proof for a certain graph H (unless NP is in QP). We need to rely on a fine-grained complexity-theoretic assumption, most likely the ETH.

The conjecture qptas-DMS is known to be implied by the following structural statement.

Conjecture 3 (Gartland–Lokshtanov [35]). *For every planar graph H, there is a constant $k := k(H)$ such that every n-vertex H-induced-minor-free graph G admits a set $X \subseteq V(G)$ of size at most k such that $N[X]$ is a* balanced separator *of G.*

Conjecture 3 is known to hold when H is K_5 minus an edge, $K_{2,t}$, W_4 [26], and has recently been shown for tC_3 [3] and W_t [23]. Improved algorithms for MIS might extend to any problem where one seeks to maximize a subset inducing a graph of low treewidth satisfying a prescribed CMSO_2 sentence. Fix a positive integer t, a CMSO_2 sentence φ, and define:

($\mathsf{tw} \leqslant t, \varphi$)-Weighted Maximum Induced Subgraph (($\mathsf{tw} \leqslant t, \varphi$)-WMIS)
Input: A vertex-weighted graph (G, w).
Goal: Find a set $S \subseteq V(G)$, such that
- $G[S] \models \varphi$,
- $\mathsf{tw}(G[S]) \leqslant t$, and
- S is of maximum weight subject to these conditions,

or conclude that no such set S exists.

Special cases of the ($\mathsf{tw} \leqslant t, \varphi$)-WMIS problem include MIS, Feedback Vertex Set, Even Cycle Transversal, Weighted Linear Forest Vertex Deletion. For instance, the latter problem, which asks for a deletion set of minimum weight leaving a disjoint union of paths in the graph, is ($\mathsf{tw} \leqslant 1, \varphi_{\neg\text{deg}3}$)-WMIS with

$$\varphi_{\neg\text{deg}3} = \forall x \neg \left(\exists y_1 \exists y_2 \exists y_3\ y_1 \neq y_2 \wedge y_1 \neq y_3 \wedge y_2 \neq y_3 \wedge E(x, y_1) \wedge E(x, y_2) \wedge E(x, y_3)\right).$$

It was further conjectured that DMS more generally holds for ($\mathsf{tw} \leqslant t, \varphi$)-WMIS.

Conjecture 4 (**Refuted**, Gartland–Lokshtanov [35])**.** *For every planar graph H, positive integer t, and CMSO_2 sentence φ, ($\mathsf{tw} \leqslant t, \varphi$)-*Weighted Maximum Induced Subgraph *can be solved in polynomial time on H-induced-minor-free graphs.*

Resolved negatively, as a consequence of [12].

Again, we only expect the weakenings of Conjecture 4 to hold. Building on the frameworks of subtasks 1.1 and 1.2, we want to determine which other logics could yield efficient algorithms in classes excluding a planar induced minor, and to match our algorithms with tight conditional lower bounds.

1.2.2 General induced minors

For qptas-DMS, we conjecture that the condition that the excluded graph H is planar can be lifted.

Conjecture 5. *For every graph H,* Maximum Independent Set *admits a quasipolynomial-time approximation scheme in H-induced-minor-free graphs.*

Establishing Conjecture 5 in full generality would be a major breakthrough. Indeed, on string graphs, a QPTAS for MIS is only known when geometric representations of the string graphs are given, and this was itself a celebrated result in computational geometry [2]. We suggest starting with special cases: for instance, non-represented string graphs. It can be noted that Conjecture 5 could even hold with a polynomial-time approximation scheme (PTAS). Again, a second step is to extend Conjecture 5 to ($\mathsf{tw} \leqslant t, \varphi$)-WMIS and other logics.

An intriguing conjecture relates induced minors and twin-width (central in Task 3), and could reveal an unexpected connection between seemingly orthogonal notions, with potentially important consequences for some of the questions of Tasks 2 and 3.

Conjecture 6. *For every graph H and every integer t, the graphs excluding H as an induced minor and $K_{t,t}$ as a subgraph have twin-width $\mathcal{O}_{H,t}(1)$.*

Graphs satisfying the precondition of Conjecture 6 have *polynomial expansion*, i.e., their n-vertex (induced) subgraphs have balanced separators of size $\mathcal{O}(n^{1-\varepsilon})$ for some constant $\varepsilon > 0$. It is another (a priori harder) open question whether every class with polynomial expansion has bounded twin-width.

Finally, we are interested in detecting induced minors. While there is now an almost-linear algorithm to find a fixed *minor* [43], some fixed graphs are NP-hard to find as induced minors (see the recent works [42, 1]). A full classification of which induced minors can be found in polynomial time is currently beyond reach. However, within bounded-degree graphs, we conjecture that this characterizes string graphs.

Conjecture 7. *For every graph H, deciding whether a graph from a bounded-degree class admits H as an induced minor is polynomial-time solvable if and only if H is a string graph.*

Our approach for the algorithmic side of Conjecture 7 relies on a conjecture on unavoidable induced minors of graphs with bounded maximum degree and unbounded clique minors (ongoing work), and efficient model checking of an extension of $\mathsf{FO{+}SDP}$ in K_t-minor-free graphs (see subtask 1.1). For the complexity side, we plan on harnessing the reductions in [1].

1.3 Task 3: Optimality program in monadically dependent classes

Since Courcelle's theorems, the frontier of tractability for MSO model checking has been well understood: it essentially coincides with *bounded clique-width* for adjacency relations (MSO_1) [25], and *bounded treewidth* for incidence relations (MSO_2) [24]. The first-order counterpart of these theorems is a central open question, and has attracted substantial attention over the last decade. A class is *monadically dependent* if it cannot be transformed into the class of all graphs by a first-order transformation, called an FO transduction (see for instance [11]). This notion encompasses, for example, classes of bounded twin-width and nowhere dense classes. It is conjectured that every monadically dependent class admits efficient FO model checking, while the converse is known for hereditary classes [29].

Conjecture 8. *Every monadically dependent class is tractable for FO.*

If true, monadic dependence would exactly characterize FO tractability. An important step toward Conjecture 8 came in 2020 with the introduction of twin-width [20]. Classes of bounded twin-width are monadically dependent. They include various families, such as classes of bounded clique-width, minor-free classes, proper permutation classes, but do not include all bounded-degree classes. The first algorithmic application was progress toward Conjecture 8.

Theorem 1 ([20])**.** *There is an algorithm that decides $G \models \varphi$ in time $\mathcal{O}_{d,k}(n)$ for every n-vertex graph G given together with a d-sequence, and every first-order sentence φ of quantifier rank k.*

In 2025, the family of merge-width parameters was introduced [30]. Classes of bounded merge-width unify bounded twin-width and bounded-degree graphs, and in fact cover classes of bounded expansion. A cubic-time FO model checking algorithm was obtained for classes of effectively bounded merge-width, and was recently improved to quadratic time.

Theorem 2 ([30, 17])**.** *Fix $d, k \in \mathbb{N}$. For some $r = \mathcal{O}_k(1)$, there is an algorithm that decides $G \models \varphi$ in time $\mathcal{O}_{d,k}(n^2)$ for every n-vertex graph G given together with a construction sequence witnessing $\mathsf{mw}_r(G) \leqslant d$, and every first-order sentence φ of quantifier rank k.*

(In Theorem 2, $\mathsf{mw}_r(G)$ denotes the radius-r merge-width of G.) The community's strong focus on Conjecture 8 has overshadowed other important questions in algorithms for structured graph classes. We propose several such questions in fine-grained complexity, approximation algorithms, and dynamic algorithms. These questions fit naturally within the broader optimality program. They may also shed a different light on monadically dependent classes than the one provided by the pursuit of Conjecture 8 (see for instance [29]).

1.3.1 Fine-grained complexity and approximability

In both Theorems 1 and 2, the dependence on k is a tower of exponentials of height $\Theta(k)$. In classes of bounded twin-width, single-exponential parameterized algorithms have been obtained for several problems including k-Independent Set and k-Dominating Set [18]. Algorithms with similar running times are not known in classes of effectively bounded merge-width.

Question 8. *For some $r = \mathcal{O}_k(1)$, is there an algorithm that solves k-*Independent Set *in time $2^{\mathcal{O}_d(k)}\, n^{\mathcal{O}(1)}$ on n-vertex graphs G given with a construction sequence witnessing that $\mathsf{mw}_r(G) \leqslant d$?*

We are especially interested in a parameterized algorithm with linear dependence on n, since Theorem 2 is only quadratic in n. A promising direction is to combine the dynamic programming approach of [18] with the freezing technique for merge-width sequences developed in [10]. We ask Question 8 because k-Independent Set is arguably the simplest NP-hard FO-definable problem. A positive answer would likely lead to fast parameterized algorithms for many other problems, as happened in the case of twin-width.

On graphs of effectively bounded twin-width, these parameterized algorithms do not perfectly match the current ETH lower bounds. Relying on the fact that $\Theta(\log n)$-subdivisions of n-vertex graphs have bounded twin-width, it is known that several problems, including k-Independent Set and k-Dominating Set, cannot be solved in time $2^{o(k/\log k)} n^{\mathcal{O}(1)}$, unless the ETH fails. Moreover, the corresponding unparameterized problems (MIS, Dominating Set, etc.) admit a $2^{\Omega(n/\log n)}$ ETH lower bound in classes of bounded twin-width. For no problem has this gap been closed against the trivial $2^{\mathcal{O}(n)}$ upper bound.

Question 9. *For every problem Π with a $2^{\Theta(n)}$-time algorithm on general graphs, optimal under the ETH, is there a $2^{\mathcal{O}(n/\log n)}$-time algorithm that solves Π on n-vertex graphs of bounded twin-width? Or is there such a problem with a $2^{\Omega(n)}$ ETH lower bound on bounded twin-width graphs?*

If we find a problem positively answering the first question of Question 9, we will naturally try and extend it to all MSO-definable problems. We suspect that the main obstacle behind this question lies in the growth rate of bounded twin-width classes: they have, up to isomorphism, only $2^{\mathcal{O}(n)}$ many n-vertex graphs. This may point to a deeper connection between computation and enumerative combinatorics, like a quantified version of Mahaney's theorem.

Maximum Independent Set is notoriously hard to approximate: For every $\varepsilon > 0$, it is NP-hard to approximate MIS on n-vertex graphs within an approximation factor of $n^{1-\varepsilon}$ [41, 54]. The best known polynomial-time algorithms achieve approximation ratio $n/\text{polylog}\ n$. In *subgraph-closed* monadically dependent classes, which coincide with nowhere dense classes, an $n^{o(1)}$-approximation algorithm follows readily from their $n^{o(1)}$-degeneracy. The same result was established for classes of effectively bounded twin-width [6]. We conjecture that this holds for every monadically dependent class.

Conjecture 9. *For every monadically dependent class $\mathcal{C}$,* Maximum Independent Set *admits, for every $\varepsilon > 0$, a polynomial n^ε-approximation algorithm on n-vertex graphs of $\mathcal{C}$.*

A natural first step toward Conjecture 9 is to treat classes of effectively bounded merge-width. Currently, no approximation algorithm with a better approximation factor than on general graphs is known.

Question 10. *Does* Maximum Independent Set *admit an n^c-approximation algorithm on classes of effectively bounded merge-width with $c < 1$? Does it admit an $n^{o(1)}$-approximation* *algorithm?*

Can an algorithm for Question 10 then be lifted to classes of subpolynomial merge-width (which are conjectured to coincide with monadically dependent classes)? For classes of bounded twin-width, even a PTAS is not currently ruled out.

Question 11. *[solved] Is there a polynomial $r(n)$-approximation algorithm for* Maximum Independent Set *in classes of effectively bounded twin-width, with $r(n)$ at most polylogarithmic?*

Solved negatively: there is a constant c such that an $n^{c/(\log \log n)^2}$-approximation algorithm for this problem would refute the ETH [14]. This is close to the upper bound of $n^{\mathcal{O}(1/\log \log n)}$ on the best approximation factor [6].

Failing a PTAS, we would seek a matching conditional lower bound for Question 11.

The MaxFO model-checking problem asks, given a graph G and an FO sentence φ on the language $\{E, X\}$ where E is interpreted as the edge set, and X as a vertex subset, to find a maximum-size set $S \subseteq V(G)$ such that $G, S \models \varphi$. Let Monotone MaxFO denote the fragment of MaxFO such that for every $S' \subseteq S$, if $G, S \models \varphi$ holds then so does $G, S' \models \varphi$. Note that [27] introduces a syntactically more restrictive notion of Monotone MaxFO, in which X may appear only negatively. (Indeed, these notions are not equivalent, since Lyndon's theorem fails over finite structures.) One can also add vertex weights to G, and seek a set S of maximum total weight.

Conjecture 10. *For every monadically dependent class $\mathcal{C}$, Monotone MaxFO model checking admits, for every $\varepsilon > 0$, a polynomial n^ε-approximation algorithm on n-vertex graphs of $\mathcal{C}$.*

The monotonicity assumption is needed because there is a polynomial reduction from Planar 3-Coloring to MaxFO model checking on planar graphs (it is claimed in [27, Example 11], and not difficult to show). A related question appears in [31], where Conjecture 10 is formulated for nowhere dense classes and a version of Monotone MaxFO where several unary relations are allowed (instead of a single one). The recent characterization of monadic dependence by flip-separability [11] is a promising starting point, but proving Conjecture 10 will require substantial new work. Its special case, Conjecture 9, is wide open, even on classes of effectively bounded merge-width.

1.3.2 Faster polynomial algorithms in monadically dependent classes and beyond

Very recently, sparse signed tree models were used to obtain polynomial algorithms on classes of bounded twin-width or merge-width that are much faster than in general graphs [17]. Importantly, these algorithms do *not* require witness sequences for low twin-width or merge-width.

Two pairs of nodes u_1v_1, u_2v_2 in a tree *cross* if one of u_1, v_1 is a strict ancestor of one of u_2, v_2, and vice versa. A *signed tree model* is a triple $\mathcal{T} = (T, A, B)$ consisting of a full binary tree T, and two disjoint sets of transversal pairs A and B of T such that no two pairs in $A \cup B$ cross. The transversal pairs of A and B are called *negative edges* and *positive edges*, respectively. We say that a transversal pair $u'v'$ *covers* a pair uv (not necessarily in $A \cup B$) if u' is an ancestor of u, v' is an ancestor of v, and there is no transversal pair $u''v''$ lower than $u'v'$ with the same property. The graph $G_\mathcal{T}$ represented by $\mathcal{T}$ has as vertex set the leaves of T, and an edge uv whenever uv is covered by a positive edge in $\mathcal{T}$; see Fig. 1. A signed tree model is *sparse* (resp. *almost sparse*) if the total number of transversal pairs is at most $\mathcal{O}(n)$ (resp. $n^{1+o(1)}$), where n is the number of leaves.

Sparse signed tree models are related to two parameters: symmetric difference and sd-degeneracy. The *symmetric difference* of two vertices u, v in a graph G is $\mathrm{sd}_G(u, v) := |(N_G(u) \setminus \{v\}) \triangle (N_G(v) \setminus \{u\})|$. The *symmetric difference* $\mathrm{sd}(G)$ of a graph G is the least d such that for every non-singleton induced subgraph H of G, there are $u \neq v \in V(H)$ with $\mathrm{sd}_H(u, v) \leqslant d$. An *sd-degeneracy sequence* of G of width d is a list of pairs of vertices of G: $(u_1, v_1), \ldots, (u_{n-1}, v_{n-1})$ such that $u_i \neq v_i$ for every

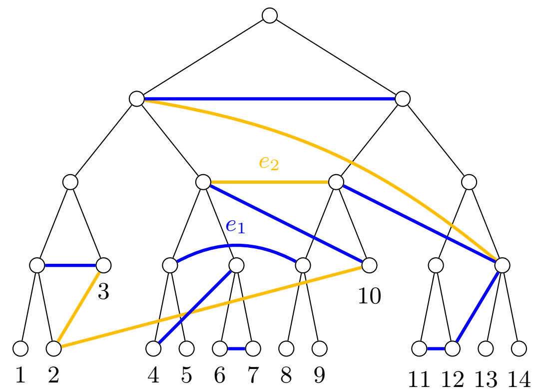

Figure 1: A signed tree model with A in amber and B in blue. Vertex 4 is adjacent to 8 because 4,8 is covered by the positive edge $e_1 \in B$, while 6 is non-adjacent to 8 because 6,8 is covered by $e_2 \in A$.

$i \in [n-1]$, $V(G) = \{u_1, u_2, \ldots, u_{n-2}, u_{n-1}, v_{n-1}\}$, there is no $i < j$ such that $u_i \in \{u_j, v_j\}$, and $\max_{i \in [n-1]} \mathrm{sd}_{G - \{u_1, u_2, \ldots, u_{i-1}\}}(u_i, v_i) \leqslant d$. The *sd-degeneracy* of G is the least d such that it admits an sd-degeneracy sequence of width d.

One can efficiently find sd-degeneracy sequences of low width in graphs of bounded twin-width.

Theorem 3. *There is an algorithm that, given an n-vertex m-edge graph G of twin-width at most d, outputs an sd-degeneracy sequence of width $\mathcal{O}_d(\log n)$, in time $\mathcal{O}_d((m+n)\log n)$ with high probability.*

In turn, sd-degeneracy sequences can be quickly converted into signed tree models.

Lemma 1 ([17])**.** *There is an $\mathcal{O}(dn + |E(G)|)$-time algorithm that, given an sd-degeneracy sequence of G of width d, outputs a signed tree model of G with $\mathcal{O}(dn)$ transversal pairs.*

Now operating on the signed tree models, one can design fast algorithms on graphs of bounded twin-width, *without* requiring that a contraction sequence is provided.

Theorem 4 ([17])**.** All-Pairs Shortest Path *can be solved in $\mathcal{O}(n^2 \log^2 n)$ time in n-vertex graphs of bounded twin-width.*

Solving All-Pairs Shortest Path answers most distance-based problems such as Diameter, Radius, Eccentricity, Wiener Index, etc.

Theorem 5 ([17])**.** *Given any adjacency matrix M of an n-vertex graph of bounded twin-width and any $n \times n$ matrix N over some additive group, the product MN can be computed in time $\mathcal{O}(n^2 \log n)$.*

Theorem 4 is in fact a consequence of:

Theorem 6. *There is an $\mathcal{O}(p \log n)$-time algorithm that, given a signed tree model with p transversal pairs of an n-vertex graph G and $v \in V(G)$, outputs a shortest-path tree of G rooted at v.*

This raises several questions.

Question 12. ***[solved]*** *Can* All-Pairs Shortest Path *be solved in $\mathcal{O}(n^2)$ time in n-vertex graphs of bounded twin-width? What about bounded merge-width?*

Solved positively: more generally for classes of linear neighborhood complexity [16].

Such an algorithm is currently known only when an $\mathcal{O}(1)$-sequence is provided [4]. Interestingly, a positive answer to Question 12 is matched by the following SETH lower bound even for Diameter on $\mathcal{O}(n)$-edge graphs: for every $\varepsilon > 0$, there is no $\mathcal{O}(n^{2-\varepsilon})$-time algorithm. In particular, an $\tilde{\mathcal{O}}(m)$-time algorithm for Diameter in m-edge graphs of bounded twin-width is ruled out.

Question 13. *[solved] Given any adjacency matrix M of an n-vertex graph of bounded twin-width (resp. of bounded merge-width) and any $n \times n$ matrix N over some additive group, can the product MN be computed in time $\mathcal{O}(n^2)$?*

Solved positively: more generally for classes of linear neighborhood complexity [16].

A positive answer in particular gives a time-optimal algorithm for Triangle Detection in dense graphs of bounded twin-width (resp. merge-width). Indeed, an algorithm for Question 13 can be used to compute the square or any constant power of a graph. Unlike for Diameter, no analogous conditional lower bound rules out faster algorithms for Triangle Detection/Counting in structured sparse graphs.

Question 14. *[solved] Can* Triangle Detection *(or* Triangle Counting*) be solved in time $\tilde{\mathcal{O}}(m)$ in m-edge graphs of bounded twin-width? What about bounded merge-width?*

Solved positively: more generally for classes of linear neighborhood complexity, and more generally for detecting a clique of size 3, 4, or 5 [16]. Actually the triangle-detection algorithm takes $\mathcal{O}(m)$ without log factors.

In general graphs, the current best algorithm runs in time $\tilde{\mathcal{O}}(m^{2\omega/(\omega+1)}) = \mathcal{O}(m^{1.407})$, and the best combinatorial algorithm runs in time $\mathcal{O}(m^{3/2})$.

Every monadically dependent class has almost sparse signed tree models. On the other hand, some monadically *independent* graph classes still admit sparse signed tree models. This is the case, for instance, for classes of bounded symmetric difference and for interval graphs. Questions 12 to 14 can be asked, with some multiplicative polylogarithmic factors, on every such class. It is thus relevant to try and characterize classes with (almost) sparse signed tree models, or at least to classify the established graph classes.

Question 15. *Which classes have (almost) sparse signed tree models?*

Remark: within monotone classes, having sparse signed tree models is equivalent to having bounded degeneracy.

By a counting argument, a class with sparse signed tree models must be *factorial*, i.e., its number of n-vertex graphs must be $2^{\mathcal{O}(n \log n)}$. It is challenging, however, to find a factorial class without sparse signed tree models.

Question 16. *[solved] Is there a hereditary factorial class without sparse signed tree models?*

An earlier paper [5] showed that yes for sparse tree models, but the same holds for signed tree models due to the anti-edge removal of [17].

We believe that permutation graphs or 2-track interval graphs should answer Question 16.

Then, the question is which of these classes admit an efficient (say, almost-linear-time) algorithm to compute the signed tree models. Recall that, by Lemma 1, it suffices to compute an sd-degeneracy sequence of bounded width.

Question 17. *[partially solved] For which classes $\mathcal{C}$ is there an algorithm that, given a graph $G \in \mathcal{C}$ of size m, computes an (almost) sparse signed tree model of G in time $\tilde{\mathcal{O}}(m)$?*

There are randomized $\tilde{\mathcal{O}}(m)$-time and deterministic $\mathcal{O}(n^2)$-time algorithms for classes of linear neighborhood complexity [16].

Signed tree models can be used to greatly improve, for structured graph classes, the dynamic guarantees achievable for general graphs. We view the following result as a proof of concept that could be strengthened and extended to other classes and problems.

Theorem 7 ([17]). *Let $\mathcal{C}$ be a class of bounded twin-width. There is a fully dynamic randomized algorithm for* Single-Source Shortest Paths *on graphs guaranteed to stay in $\mathcal{C}$, with $\mathcal{O}(n)$ amortized update time, and $\mathcal{O}(n \log n)$ worst-case query time.*

Knowing in advance that the number of edits is $\mathcal{O}(n \log n)$, the update time becomes $\mathcal{O}(1)$ worst-case.

References

[1] P. Aboulker, É. Bonnet, T. Picavet, et al. Induced Disjoint Paths Without an Induced Minor. ICALP, 2025.

[2] A. Adamaszek, S. Har-Peled, and A. Wiese. Approximation Schemes for Independent Set and Sparse Subsets of Polygons. J. ACM, 2019.

[3] J. Ahn, J. P. Gollin, T. Huynh, et al. A coarse Erdős-Pósa theorem. SODA, 2025.

[4] M. Bannach, F. A. Marwitz, and T. Tantau. Faster Graph Algorithms Through DAG Compression. STACS, 2024.

[5] M. Bannach, F. A. Marwitz, and T. Tantau. Faster Graph Algorithms Through DAG Compression. 41st International Symposium on Theoretical Aspects of Computer Science, STACS 2024, Clermont-Ferrand, France, March 12-14, 2024, 2024.

[6] P. Bergé, É. Bonnet, H. Déprés, et al. Approximating Highly Inapproximable Problems on Graphs of Bounded Twin-Width. STACS, 2023.

[7] M. Bojańczyk. Separator logic and star-free expressions for graphs. arXiv, 2021.

[8] M. Bojańczyk, M. Pilipczuk, W. Przybyszewski, et al. Low rank MSO. arXiv, 2025.

[9] M. Bonamy, É. Bonnet, H. Déprés, et al. Sparse graphs with bounded induced cycle packing number have logarithmic treewidth. J. Comb. Theory B, 2024.

[10] M. Bonamy and C. Geniet. χ-Boundedness and Neighbourhood Complexity of Bounded Merge-Width Graphs. arXiv, 2025.

[11] É. Bonnet, S. Braunfeld, I. Eleftheriadis, et al. Separability Properties of Monadically Dependent Graph Classes. ICALP, 2025.

[12] É. Bonnet and Y. Chang. Max Independent Set Remains NP-hard when Excluding a Planar Induced Minor. 2026.

[13] É. Bonnet, J. Czyżewska, T. Masařík, et al. QPTAS for MWIS and finding large sparse induced subgraphs in graphs with few independent long holes. arXiv, 2026.

[14] É. Bonnet, M. Dumas, and J. Duron. Independent Set Hardness in Graphs of Bounded Twin-Width and Low-Radius Merge-Width. 2026.

[15] É. Bonnet, J. Duron, C. Geniet, et al. Maximum Independent Set when Excluding an Induced Minor: $K_1 + tK_2$ and $tC_3 \uplus C_4$. Algorithmica, 2026.

[16] É. Bonnet, J. Duron, M. Pilipczuk, et al. Time-Optimal APSP and Matrix Multiplication in Classes of Linear Neighborhood Complexity. 2026.

[17] É. Bonnet, C. Geniet, E. J. Kim, et al. Fast Shortest Path in Graphs With Sparse Signed Tree Models and Applications. arXiv, 2026.

[18] É. Bonnet, C. Geniet, E. J. Kim, et al. Twin-width III: Max Independent Set, Min Dominating Set, and Coloring. ICALP, 2021.

[19] É. Bonnet, U. Giocanti, P. O. de Mendez, et al. Twin-Width IV: Ordered Graphs and Matrices. J. ACM, 2024.

[20] É. Bonnet, E. J. Kim, S. Thomassé, et al. Twin-width I: Tractable FO Model Checking. J. ACM, 2022.

[21] H. Buffière, E. J. Kim, and P. Ossona de Mendez. Shallow vertex minors, stability, and dependence. Innovations in Graph Theory, 2024.

[22] H. Buffière, Y. Lin, J. Nešetřil, et al. Characterizations of monadically dependent tree-ordered weakly sparse structures. arXiv, 2026.

[23] M. Chudnovsky, J. P. Gollin, M. Krnc, et al. Dominated balanced separators in wheel-induced-minor-free graphs. arXiv, 2025.

[24] B. Courcelle. The Monadic Second-Order Logic of Graphs. I. Recognizable Sets of Finite Graphs. Information and Computation, 1990.

[25] B. Courcelle, J. A. Makowsky, and U. Rotics. Linear Time Solvable Optimization Problems on Graphs of Bounded Clique-Width. Theory Comput. Syst., 2000.

[26] C. Dallard, M. Milanič, and K. Štorgel. Treewidth versus clique number. III. Tree-independence number of graphs with a forbidden structure. J. Comb. Theory B, 2024.

[27] A. Dawar, M. Grohe, S. Kreutzer, et al. Approximation schemes for first-order definable optimisation problems. LICS, 2006.

[28] J. Dreier, I. Eleftheriadis, N. Mählmann, et al. First-Order Model Checking on Monadically Stable Graph Classes. FOCS, 2024.

[29] J. Dreier, N. Mählmann, and S. Toruńczyk. Flip-Breakability: A Combinatorial Dichotomy for Monadically Dependent Graph Classes. STOC, 2024.

[30] J. Dreier and S. Toruńczyk. Merge-Width and First-Order Model Checking. STOC, 2025.

[31] Z. Dvorák. Approximation Metatheorems for Classes with Bounded Expansion. SWAT, 2022.

[32] F. V. Fomin, P. A. Golovach, I. Sau, et al. Compound Logics for Modification Problems. ACM Trans. Comput. Logic, 2024.

[33] J. Gajarský, M. Pilipczuk, M. Sokolowski, et al. Elementary first-order model checking for sparse graphs. LICS, 2024.

[34] J. Gao, R. Impagliazzo, A. Kolokolova, et al. Completeness for First-order Properties on Sparse Structures with Algorithmic Applications. ACM Trans. Algorithms, 2019.

[35] P. Gartland. Quasi-Polynomial Time Techniques for Independent Set and Beyond in Hereditary Graph Classes. University of California, Santa Barbara, 2023.

[36] P. Gartland, D. Lokshtanov, M. Pilipczuk, et al. Finding large induced sparse subgraphs in $C_{>t}$-free graphs in quasipolynomial time. STOC, 2021.

[37] A. Georgakopoulos and P. Papasoglu. Graph Minors and Metric Spaces. Combinatorica, 2025.

[38] P. A. Golovach, G. Stamoulis, and D. M. Thilikos. Model-Checking for First-Order Logic with Disjoint Paths Predicates in Proper Minor-Closed Graph Classes. SODA, 2023.

[39] M. Grohe, S. Kreutzer, and S. Siebertz. Deciding First-Order Properties of Nowhere Dense Graphs. J. ACM, 2017.

[40] A. Grzesik, T. Klimosová, M. Pilipczuk, et al. Polynomial-Time Algorithm for Maximum Weight Independent Set on P_6-Free Graphs. ACM Trans. Algorithms, 2022.

[41] J. Håstad. Clique is Hard to Approximate Within $n^{1-\varepsilon}$. FOCS, 1996.

[42] T. Korhonen and D. Lokshtanov. Induced-Minor-Free Graphs: Separator Theorem, Subexponential Algorithms, and Improved Hardness of Recognition. SODA, 2024.

[43] T. Korhonen, M. Pilipczuk, and G. Stamoulis. Minor Containment and Disjoint Paths in Almost-Linear Time. FOCS, 2024.

[44] M. Lampis. The Primal Pathwidth SETH. SODA, 2025.

[45] M. Lampis. k-SUM Hardness Implies Treewidth-SETH. SODA, 2026.

[46] J. Nešetřil and P. O. de Mendez. Sparsity - Graphs, Structures, and Algorithms. 2012.

[47] S.-I. Oum. Rank-width: Algorithmic and structural results. Discrete Applied Mathematics, 2017.

[48] M. Pilipczuk. Problems Parameterized by Treewidth Tractable in Single Exponential Time: A Logical Approach. MFCS, 2011.

[49] M. Pilipczuk, N. Schirrmacher, S. Siebertz, et al. Algorithms and Data Structures for First-Order Logic with Connectivity Under Vertex Failures. ICALP, 2022.

[50] I. Sau, N. Schirrmacher, S. Siebertz, et al. Model Checking for Low Monodimensionality Fragments of CMSO on Topological-minor-free Graph Classes. Manuscript, 2026.

[51] I. Sau, G. Stamoulis, and D. M. Thilikos. Parameterizing the quantification of CMSO: model checking on minor-closed graph classes. SODA, 2025.

[52] N. Schirrmacher, S. Siebertz, G. Stamoulis, et al. Model Checking Disjoint-Paths Logic on Topological-Minor-Free Graph Classes. LICS, 2024.

[53] N. Schirrmacher, S. Siebertz, and A. Vigny. First-Order Logic with Connectivity Operators. ACM Trans. Comput. Logic, 2023.

[54] D. Zuckerman. Linear Degree Extractors and the Inapproximability of Max Clique and Chromatic Number. Theory Comput., 2007.