Counting Small Induced Subgraphs Satisfying Monotone Properties

Marc Roth (ID)
Merton College, University of Oxford, United Kingdom
marc.roth@merton.ox.ac.uk

Johannes Schmitt (ID)
Mathematical Institute, University of Bonn, Germany
schmitt@math.uni-bonn.de

Philip Wellnitz (ID)
Max Planck Institute for Informatics,
Saarland Informatics Campus (SIC), Saarbrücken, Germany
wellnitz@mpi-inf.mpg.de

──── **Abstract** ────

Given a graph property Φ, the problem #INDSUB(Φ) asks, on input a graph G and a positive integer k, to compute the number #IndSub($\Phi, k \to G$) of induced subgraphs of size k in G that satisfy Φ. The search for *explicit* criteria on Φ ensuring that #INDSUB(Φ) is hard was initiated by Jerrum and Meeks [J. Comput. Syst. Sci. 15] and is part of the major line of research on counting small patterns in graphs. However, apart from an implicit result due to Curticapean, Dell and Marx [STOC 17] proving that a full classification into "easy" and "hard" properties is possible and some partial results on edge-monotone properties due to Meeks [Discret. Appl. Math. 16] and Dörfler et al. [MFCS 19], not much is known.

In this work, we fully answer and explicitly classify the case of monotone, that is subgraph-closed, properties: We show that for any non-trivial monotone property Φ, the problem #INDSUB(Φ) cannot be solved in time $f(k) \cdot |V(G)|^{o(k/\log^{1/2}(k))}$ for any function f, unless the Exponential Time Hypothesis fails. By this, we establish that any significant improvement over the brute-force approach is unlikely; in the language of parameterized complexity, we also obtain a #W[1]-completeness result.

To prove our result, we use that for fixed Φ and k, we can express the function $G \mapsto$ #IndSub($\Phi, k \to G$) as a finite linear-combination of homomorphism counts from graphs H_i to G. The coefficient vectors of these homomorphism counts in the linear combination are called the *homomorphism vectors* associated to Φ; by the Complexity Monotonicity framework of Curticapean, Dell and Marx [STOC 17], the positions of non-zero entries of these vectors are known to determine the complexity of #INDSUB(Φ). Our main technical result lifts the notion of f-polynomials from simplicial complexes to graph properties and relates the derivatives of the f-polynomial of Φ to its homomorphism vector. We then apply results from the theory of Hermite-Birkhoff interpolation to the f-polynomial to establish sufficient conditions on Φ which ensure that certain entries in the homomorphism vector do not vanish—which in turn implies hardness. For monotone graph properties, non-triviality then turns out to be a sufficient condition. Using the same method, we also prove a conjecture by Jerrum and Meeks [TOCT 15, Combinatorica 19]: #INDSUB(Φ) is #W[1]-complete if Φ is a non-trivial graph property only depending on the number of edges of the graph.

Keywords and phrases Counting complexity, fine-grained complexity, graph homomorphisms, induced subgraphs, parameterized complexity

1   Introduction

Detection, enumeration and counting of patterns in graphs are among the most well-studied computational problems in theoretical computer science with a plethora of applications in diverse disciplines, including biology [59, 33], statistical physics [62, 41, 42], neural and social networks [50] and database theory [34], to name but a few. At the same time, those problems subsume in their unrestricted forms some of the most infamous NP-hard problems such as Hamiltonicity, the clique problem, or, more generally, the subgraph isomorphism problem [17, 65]. In the modern-day era of "big data", where even quadratic-time algorithms may count as inefficient, it is hence crucial to find relaxations of hard computational problems that allow for tractable instances.

A very successful approach for a more fine-grained understanding of hard computational problems is a multivariate analysis of the complexity of the problem: Instead of establishing upper and (conditional) lower bounds only depending on the input size, we aim to find additional parameters that, in the best case, are small in real-world instances and allow for efficient algorithms if assumed to be bounded. In case of detection and counting of patterns in graphs, it turns out that the size of the pattern is often significantly smaller than the size of the graph: Consider as an example the evaluation of database queries. While a classical analysis of this problem requires considering instances where the size of the query is as large as the database, a multivariate analysis allows us to impose the restriction of the query being much smaller than the database, which is the case for real-world instances. More concretely, suppose we are given a query φ of size k and a database B of size n, and we wish to evaluate the query φ on B. Assume further, that we are given two algorithms for the problem: One has a running time of $O(n^k)$, and the other one has a running time of $O(2^k \cdot n)$. While, classically, both algorithms are inefficient in the sense that their running times are not bounded by a polynomial in the input size $n + k$, the second algorithm is significantly better than the first one for real-world instances and can even be considered efficient.

In this work, we focus on *counting* of small patterns in large graphs. The field of counting complexity was founded by Valiant's seminal result on the complexity of computing the permanent [66, 67], where it was shown that computing the number of perfect matchings in a graph is #P-complete, and thus harder than every problem in the polynomial-time hierarchy PH [64]. This is in sharp contrast to the fact that *finding* a perfect matching in a graph can be done in polynomial-time [28]. Hence, a perfect matching is a pattern that allows for efficient detection but is unlikely to admit efficient counting. Initiated by Valiant, computational counting evolved into a well-studied subfield of theoretical computer science. In particular, it turns out that counting problems are closely related to computing partition functions in statistical physics [62, 41, 42, 32, 15, 3]. Indeed, one of the first algorithmic result in the field of computational counting is the famous FKT-Algorithm by the statistical physicists Fisher, Kasteleyn and Temperley [62, 41, 42] that computes the partition function of the so-called dimer model on planar structures, which is essentially equivalent to computing the number of perfect matchings in a planar graph. The FKT-Algorithm is the foundation of the framework of holographic algorithms, which, among others, have been used to identify the tractable cases of a variety of complexity classifications for counting constraint satisfaction problems [68, 7, 5, 6, 35, 8, 9, 2]. Unfortunately, the intractable cases of those classifications indicate that, except for rare examples, counting is incredibly hard (from a complexity theory point of view). In particular, many efficiently solvable combinatorial decision problems turn out to be intractable in their counting versions, such as counting of satisfying assignments of monotone 2-CNFs [67], counting of independent sets in bipartite graphs [52] or counting of s-t-paths [67], to name but a few. For this reason, we follow the multivariate approach as outlined previously and restrict ourselves in this work on counting of *small* patterns in large graphs. Among others, problems of this kind find applications in neural and social networks [50], computational biology [1, 57], and database theory [27, 11, 12, 24].

Formally, we follow the approach of Jerrum and Meeks [37] and study the family of problems #INDSUB(Φ): Given a graph property Φ, the problem #INDSUB(Φ) asks, on input a graph G and a positive integer k, to compute the number of induced subgraphs of size k in G that satisfy Φ.[1] As observed by Jerrum and Meeks, the generality of the definition allows to express counting of almost arbitrary patterns of size k in a graph, subsuming counting of k-cliques and k-independent sets as very special cases.

Assuming that Φ is computable, we note that the problem #INDSUB(Φ) can be solved by brute-force in time $O(f(k) \cdot |V(G)|^k)$ for some function f only depending on Φ. The corresponding algorithm enumerates all subsets of k vertices of G and counts how many of those subsets satisfy Φ. As we consider k to be significantly smaller than $|V(G)|$, we are interested in the dependence of the exponent on k. More precisely, the goal is to find the best possible $g(k)$ such that #INDSUB(Φ) can be solved in time

$$O(f(k) \cdot |V(G)|^{g(k)}) \tag{1}$$

for some function f such that f and g only depend on Φ. Readers familiar with parameterized complexity theory will identify the case of $g(k) \in O(1)$ as fixed-parameter tractability (FPT) results. We first provide some background and elaborate on the existing results on #INDSUB(Φ) before we present the contributions of this paper.

1.1 Prior Work

So far, the problem #INDSUB(Φ) has been investigated using primarily the framework of parameterized complexity theory. As indicated before, #INDSUB(Φ) is in FPT if the function g in Equation (1) is bounded by a constant (independent of k), and the problem is #W[1]-complete, if it is at least as hard as the parameterized clique problem; here #W[1] should be considered a parameterized counting equivalent of NP and we provide the formal details in Section 2. In particular, the so-called Exponential Time Hypothesis (ETH) implies that #W[1]-complete problems are not in FPT; again, this is made formal in Section 2.

The problem #INDSUB(Φ) was first studied by Jerrum and Meeks [37]. They introduced the problem and proved that #INDSUB(Φ) is #W[1]-complete if Φ is the property of being connected. Implicitly, their proof also rules out the function g of Equation (1) being in $o(k)$, unless ETH fails, which establishes a tight conditional lower bound. In a subsequent line of research [38, 48, 39], Jerrum and Meeks proved #INDSUB(Φ) to be #W[1]-complete if at least one of the following is true:

(1) Φ has *low edge-densities*; this is true for instance for all sparse properties such as planarity and made formal in Section 4.1.

(2) Φ holds for a graph H if and only if the number of edges of H is even/odd.

(3) Φ is closed under the addition of edges, and the minimal elements have large treewidth.

Unfortunately, none of the previous results establishes a conditional lower bound that comes close to the upper bound given by the brute-force algorithm. This is particularly true due to the application of Ramsey's Theorem in the proofs of many of the prior results: Ramsey's Theorem states that there is a function $R(k) = 2^{\Theta(k)}$ such that every graph with at least $R(k)$ vertices contains either a k-independent set or a k-clique [53, 61, 29]. Relying on this result for a reduction from finding or counting k-independent sets or k-cliques, the best implicit conditional lower bounds achieved only rule out an algorithm running in time $f(k) \cdot |V(G)|^{o(\log k)}$ for any function f.

Moreover, the previous results only apply to a very specific set of properties. In particular, Jerrum and Meeks posed the following open problem concerning a generalization of the second result (2); we say that Φ is k-trivial, if it is either true or false for all graphs with k vertices.

---

[1] Note that #INDSUB(Φ) is identical to p-#UNLABELLEDINDUCEDSUBGRAPHWITHPROPERTY(Φ) as defined in [37].

Conjecture 1.1 ([38, 39]). *Let Φ be a graph property that only depends on the number of edges of a graph. If for infinitely many k the property Φ is not k-trivial, then $\#\textsc{IndSub}(\Phi)$ is $\#\mathsf{W}[1]$-complete.*

Note that the condition of Φ not being k-trivial for infinitely many k is necessary for hardness, as otherwise, the problem becomes trivial if k exceeds a constant depending only on Φ.

The first major breakthrough towards a complete understanding of the complexity of $\#\textsc{IndSub}(\Phi)$ is the following implicit classification due to Curticapean, Dell and Marx [20]:

Theorem 1.2 ([20]). *Let Φ denote a graph property. Then the problem $\#\textsc{IndSub}(\Phi)$ is either* FPT *or* $\#\mathsf{W}[1]$*-complete.*

While the previous classification provides a very strong result for the structural complexity of $\#\textsc{IndSub}(\Phi)$, it leaves open the question of the precise bound on the function g. Furthermore, it is implicit in the sense that it does not reveal the complexity of $\#\textsc{IndSub}(\Phi)$ if a concrete property Φ is given. Nevertheless, the technique introduced by Curticapean, Dell and Marx, which is now called *Complexity Monotonicity*, turned out to be the right approach for the treatment of $\#\textsc{IndSub}(\Phi)$. In particular, the subsequent results on $\#\textsc{IndSub}(\Phi)$, including the classification in this work, have been obtained by strong refinements of *Complexity Monotonicity*; we provide a brief introduction when we discuss the techniques used in this paper. More concretely, a superset of the authors established the following classifications for edge-monotone properties in recent years [56, 26]: The problem $\#\textsc{IndSub}(\Phi)$ is $\#\mathsf{W}[1]$-complete and, assuming ETH, cannot be solved in time $f(k) \cdot |V(G)|^{o(k)}$ for any function f, if at least one of the following is true:[2]

- Φ is non-trivial, closed under the removal of edges and false on odd cycles.
- Φ is non-trivial on bipartite graphs and closed under the removal of edges.

While the second result completely answers the case of edge-monotone properties on bipartite graphs, a general classification of edge-monotone properties is still unknown.

1.2 Our Results

We begin with monotone properties, that is, properties that are closed under taking subgraphs. We classify those properties completely and explicitly; the following theorem establishes hardness and an almost tight conditional lower bound.

Main Theorem 1. *Let Φ denote a monotone graph property. Suppose that for infinitely many k the property Φ is not k-trivial. Then $\#\textsc{IndSub}(\Phi)$ is $\#\mathsf{W}[1]$-complete and cannot be solved in time $f(k) \cdot |V(G)|^{o(k/\sqrt{\log k})}$ for any function f, unless ETH fails.*

In fact, we obtain a tight bound, that is, we can drop the factor of $1/\sqrt{\log k}$ in the exponent, assuming the conjecture that "you cannot beat treewidth" [47]. The latter is an important open problem in parameterized and fine-grained complexity theory asking for a tight conditional lower bound for the problem of *finding* a homomorphism from a small graph H to a large graph G: The best known algorithm for that problem runs in time $\mathsf{poly}(|V(H)|) \cdot |V(G)|^{O(\mathsf{tw}(H))}$, where $\mathsf{tw}(H)$ is the treewidth[3] of H (see for instance [25, 47, 20]), and the question is whether this running time is essentially optimal; we discuss the details later in the paper.

---

[2] We provide simplified statements; the formal and more general results can be found in [56, 26].

[3] We will only rely on treewidth in a black-box manner in this paper and thus refer the reader for instance to [22, Chapter 7] for a detailed treatment.

As a concrete example of a property that is classified by Main Theorem 1, but that was not classified before, consider the (monotone) property of being 3-colorable: Recall that a 3-coloring of a graph is a function mapping each vertex to one of three colors such that no two adjacent vertices are mapped to the same color. Clearly any subgraph of a graph G admits a 3-coloring if G does.

Note that in Main Theorem 1, the assumption of Φ not being k-trivial for infinitely many k is necessary, as otherwise the problem $\#\textsc{IndSub}(\Phi)$ becomes trivial for all k that are greater than a constant only depending on Φ.

Note further, that $\#\mathsf{W}[1]$-completeness in Main Theorem 1 is not surprising, as the decision version of $\#\textsc{IndSub}(\Phi)$ was implicitly shown to be $\mathsf{W}[1]$-complete by Khot and Raman [43]; $\mathsf{W}[1]$ is the decision version of $\#\mathsf{W}[1]$ and should be considered a parameterized equivalent of NP. However, their reduction is not parsimonious. Also, their proof uses Ramsey's Theorem and thus only yields an implicit conditional lower bound of $f(k) \cdot |V(G)|^{o(\log k)}$, whereas our lower bound is almost tight.

Our second result establishes an almost tight lower bound for sparse properties, that is, properties Φ that admit a constant s such that ever graph H for which Φ holds has at most $s \cdot |V(H)|$ many edges. Furthermore, the bound can be made tight if the set $\mathcal{K}(\Phi)$ of positive integers k for which Φ is not k-trivial is additionally *dense*. By this we mean that there is a constant ℓ such that for every positive integer n, there exists $n \leq k \leq \ell n$ such that Φ is not k-trivial. Note that density rules out artificial properties such as $\Phi(H) = 1$ if and only if H is an independent set and has precisely $2 \uparrow n$ vertices for some positive integer n, with $2 \uparrow n$ the n-fold exponential tower with base 2. In particular, for every property Φ that is k-trivial only for finitely many k, the set $\mathcal{K}(\Phi)$ is dense.

Main Theorem 2. *Let Φ denote a sparse graph property such that Φ is not k-trivial for infinitely many k. Then, $\#\textsc{IndSub}(\Phi)$ is $\#\mathsf{W}[1]$-complete and cannot be solved in time $f(k) \cdot |V(G)|^{o(k/\log k)}$ for any function f, unless ETH fails.*

If $\mathcal{K}(\Phi)$ is additionally dense, then $\#\textsc{IndSub}(\Phi)$ cannot be solved in time $f(k) \cdot |V(G)|^{o(k)}$ for any function f, unless ETH fails. ⌟

Our third result solves the open problem posed by Jerrum and Meeks by proving that (a strengthened version of) Conjecture 1.1 is true.

Main Theorem 3. *Let Φ denote a computable graph property that only depends on the number of edges of a graph. If Φ is not k-trivial for infinitely many k, then $\#\textsc{IndSub}(\Phi)$ is $\#\mathsf{W}[1]$-complete and cannot be solved in time $f(k) \cdot |V(G)|^{o(k/\log k)}$ for any function f, unless ETH fails.*

If $\mathcal{K}(\Phi)$ is additionally dense, then $\#\textsc{IndSub}(\Phi)$ cannot be solved in time $f(k) \cdot |V(G)|^{o(k/\sqrt{\log k})}$ for any function f, unless ETH fails. ⌟

Note that, similar to Main Theorem 1, the conditional lower bounds in the previous two theorems become tight, if "you cannot beat treewidth" [47]; in particular, the condition of being dense can be removed in that case.

Finally, we consider properties that are *hereditary*, that is, closed under taking *induced* subgraphs. We obtain a criterion on such graph properties that, if satisfied, yields a tight conditional lower bound for the complexity of $\#\textsc{IndSub}(\Phi)$. While the statement of the criterion is deferred to the technical discussion, we can see that every hereditary property that is defined by a single forbidden induced subgraph satisfies the criterion.

Main Theorem 4. *Let H be a graph with at least 2 vertices and let Φ denote the property of being H-free, that is, a graph satisfies Φ if and only if it does not contain H as an induced subgraph. Then, $\#\textsc{IndSub}(\Phi)$ is $\#\mathsf{W}[1]$-complete and cannot be solved in time $f(k) \cdot |V(G)|^{o(k)}$ for any function f, unless ETH fails.* ⌟

Note that the case of H being the graph with one vertex, which is excluded above, yields the property Φ which is false on all graphs G with at least one vertex, for which $\#\textsc{IndSub}(\Phi)$ is the constant zero-function and thus trivially solvable. Hence, Main Theorem 4 establishes indeed a complete classification for all properties $\Phi=$"H-free".

1.3   Technical Overview

We rely on the framework of Complexity Monotonicity of computing linear combinations of homomorphism counts [20]. More precisely, it is known that for every computable graph property Φ and positive integer k, there exists a unique computable function a from graphs to rational numbers such that for all graphs G

$$\#\mathsf{IndSub}(\Phi, k \to G) = \sum_H a(H) \cdot \#\mathsf{Hom}(H \to G), \tag{2}$$

where $\#\mathsf{IndSub}(\Phi, k \to G)$ denotes the number of induced subgraphs of size k in G that satisfy Φ, and $\#\mathsf{Hom}(H \to G)$ denotes the number of graph homomorphisms from H to G. It is known that the function a has finite support, that is, there is only a finite number of graphs H for which $a(H) \neq 0$.

Intuitively, Complexity Monotonicity states that computing a linear combination of homomorphism counts is *precisely* as hard as computing its hardest term [20]. Furthermore, the complexity of computing the number of homomorphisms from a small graph H to a large graph G is (almost) precisely understood by the dichotomy result of Dalmau and Jonsson [23] and the conditional lower bound under ETH due to Marx [47]: Roughly speaking, it is possible to compute $\#\mathsf{Hom}(H \to G)$ efficiently if and only if H has small treewidth. As a consequence, the complexity of computing $\#\mathsf{IndSub}(\Phi, k \to G)$ is precisely determined by the support of the function a. Unfortunately, determining the latter turned out to be an incredibly hard task: It was shown in [56] and [26] that the function a subsumes a variety of algebraic and even topological invariants. As a concrete example, a subset of the authors showed that for edge-monotone properties Φ, the coefficient $a(K_k)$ of the complete graph in Equation (2) is, up to a factor of $k!$, equal to the reduced Euler characteristic of what is called the simplicial graph complex of Φ and k [56]. By this, a connection to Karp's Evasiveness Conjecture[4] was established. In particular, it is known that Φ is evasive on k-vertex graphs if the reduced Euler characteristic is non-zero [40]. As a consequence, the coefficient $a(K_k)$ can reveal a property to be evasive on k-vertex graphs if shown to be non-zero.

The previous example illustrated that identifying the support of the function a in Equation (2) is a hard task, but using the framework of Complexity Monotonicity requires us to solve this task. In this work, we present a solution for properties whose f-vectors (see below) have low Hamming weight: Given a property Φ and a positive integer k, we define a $\binom{k}{2} + 1$ dimensional vector $f^{\Phi,k}$ by setting $f_i^{\Phi,k}$ to be the number of edge-subsets of size i of the complete graph with k vertices such that the induced graph satisfies Φ, that is,

$$f_i^{\Phi,k} := \#\{A \subseteq E(K_k) \mid \#A = i \wedge \Phi(K_k[A]) = 1\}$$

for all $i = 0, \dots, \binom{k}{2}$. By this, we lift the notion of f-vectors from abstract simplicial complexes to graph properties; readers familiar with the latter will observe that the f-vector of an edge-monotone property Φ equals the f-vector of its associated graph complex (see for instance [4]). Similarly, we introduce the notions of h-vectors $h^{\Phi,k}$ and f-polynomials $\mathtt{f}_{\Phi,k}$ of graph properties, defined as follows; we set $d = \binom{k}{2}$.

$$h_\ell^{\Phi,k} := \sum_{i=0}^{\ell} (-1)^{\ell-i} \cdot \binom{d-i}{\ell-i} \cdot f_i^{\Phi,k}, \text{ where } \ell \in \{0, \dots, d\}; \quad \text{and} \quad \mathtt{f}_{\Phi,k}(x) := \sum_{i=0}^{d} f_i^{\Phi,k} \cdot x^{d-i}.$$

---

[4] Intuitively, the Evasiveness Conjecture states that every decision tree algorithm verifying a non-trivial edge-monotone graph property has to query every edge of the input graph in the worst case [54, 49].

Our main combinatorial insight relates the function a in Equation (2) to the h-vector of Φ. For the formal statement, we let $\mathcal{H}(\Phi, k, i)$ denote the set of all graphs H with k vertices and i edges that satisfy Φ. We then show that for all $i = 0, \ldots, d$, we have

$$k! \sum_{H \in \mathcal{H}(\Phi, k, i)} a(H) = h_i^{\Phi, k}.$$

In particular, the previous equation shows that there is a graph H with i edges that survives with a non-zero coefficient $a(H)$ in Equation (2) whenever the i-th entry of the h-vector $h^{\Phi, k}$ is non-zero. As graphs with many edges have high treewidth, we can thus establish hardness of computing $\#\mathsf{IndSub}(\Phi, k \to G)$ by proving that there is a non-zero entry with a high index in $h^{\Phi, k}$. To this end, we relate $h^{\Phi, k}$ and $f^{\Phi, k}$ by observing that their entries are evaluations of the derivatives of the f-polynomial $\mathtt{f}_{\Phi, k}(x)$. More concretely, our goal is to show that a large amount of high-indexed zero entries of $h^{\Phi, k}$ yields that the only polynomial of degree at most d that satisfies the constrains given by the evaluations of the derivatives is the zero polynomial. However, the latter can only be true if Φ is trivially false on k-vertex graphs. Using Hermite-Birkhoff interpolation and Pólya's Theorem we are able to achieve this goal whenever the Hamming weight of $f^{\Phi, k}$ is small. Our meta-theorem thus classifies the complexity of $\#\mathrm{IndSub}(\Phi)$ in terms of the Hamming weight of the f-vectors of Φ.

Main Theorem 5. *Let Φ denote a computable graph property and suppose that Φ is not k-trivial for infinitely many k. Let $\beta : \mathcal{K}(\Phi) \to \mathbb{Z}_{\geq 0}$ denote the function that maps k to $\binom{k}{2} - \mathsf{hw}(f^{\Phi, k})$. If $\beta(k) \in \omega(k)$ then the problem $\#\mathrm{IndSub}(\Phi)$ is $\#\mathsf{W}[1]$-complete and cannot be solved in time*

$$g(k) \cdot n^{o((\beta(k)/k)/(\log(\beta(k)/k)))}$$

for any function g, unless ETH fails.

For the refined conditional lower bounds in case of monotone properties and properties for which the set $\mathcal{K}(\Phi)$ is dense (see Main Theorems 1 to 3), we furthermore rely on a consequence of the Kostochka-Thomason-Theorem [44, 63] that establishes a lower bound on the size of the smallest clique-minors of graphs with many edges.

In contrast to the previous families of properties, we do not rely on the general meta-theorem (Main Theorem 5) for our treatment of hereditary properties. Instead, we carefully construct a reduction from counting k-independent sets in bipartite graphs. Given a hereditary graph property Φ defined by the (possibly infinite) set $\Gamma(\Phi)$ of forbidden induced subgraphs, we say that Φ is *critical* if there is a graph $H \in \Gamma(\Phi)$ and an edge e of H such that the graph obtained from H by deleting e and then cloning the former endpoints of e satisfies Φ; the formal definition is provided in Section 6. The reduction from counting k-independent sets in bipartite graphs then yields the following result:

Main Theorem 6. *Let Φ denote a computable and critical hereditary graph property. Then $\#\mathrm{IndSub}(\Phi)$ is $\#\mathsf{W}[1]$-complete and cannot be solved in time $g(k) \cdot n^{o(k)}$ for any function g, unless ETH fails.*

We then establish that every hereditary property with precisely one non-trivial forbidden subgraph H is critical, which yields Main Theorem 4.

1.4 Organization of the Paper

We begin with providing all necessary technical background in Section 2. In particular, we introduce the most important notions in parameterized and fine-grained complexity theory, as well as the principle of Hermite-Birkhoff interpolation.

Section 3 presents and proves our main combinatorial result which relates the f-vectors and h-vectors of a computable graph property Φ on k-vertex graphs to the coefficients in the associated linear combination of homomorphisms as given by Equation (2).

We derive the meta-theorem for the complexity classification of #INDSUB(Φ) afterwards in Section 4 and illustrate its applicability by establishing new conditional lower bounds for properties that are monotone, that have low edge-densities, and that depend only on the number of edges of a graph. Those bounds are refined to match the statements of Main Theorems 1 and 2 in Section 5 by combining our main combinatorial result with results from extremal graph theory that relate the number of edges of a graph to the size of its largest clique-minor.

Finally, we present our treatment of hereditary properties in Section 6.

Acknowledgements

2   Preliminaries

Given a finite set S, we write $\#S$ and $|S|$ for the cardinality of S. Further, given a non-negative integer r, we set $[r] := \{1, \ldots, r\}$; in particular, we have $[0] = \emptyset$. The *hamming weight* of a vector $f \in \mathbb{Q}^n$, denoted by $\mathsf{hw}(f)$, is defined to be the number of non-zero entries of f.

Graphs in this work are simple and do not contain self-loops. Given a graph G, we write $V(G)$ for the vertices and $E(G)$ for the edges of G. Furthermore, we define $\mathcal{G}$ to be the set of all (isomorphism classes of) graphs. The *complement* $\overline{G}$ of a graph G has vertices $V(G)$ and edges $\overline{E(G)} \setminus \{\{v, v\} \mid v \in V(G)\}$.

Given a subset $\hat{E}$ of edges of a graph G, we write $G[\hat{E}]$ for the graph with vertices $V(G)$ and edges $\hat{E}$. Given a subset $\hat{V}$ of vertices of a graph G, we write $G[\hat{V}]$ for the graph with vertices $\hat{V}$ and edges $E(G) \cap \hat{V}^2$. In particular, we say that $G[\hat{V}]$ is an *induced subgraph* of G. Given graphs H and G, we define $\mathsf{IndSub}(H \to G)$ to be the set of all induced subgraphs of G that are isomorphic to H.

Given graphs H and G, a *homomorphism* from H to G is a function $\varphi : V(H) \to V(G)$ such that $\{\varphi(u), \varphi(v)\} \in E(G)$ whenever $\{u, v\} \in E(H)$. We write $\mathsf{Hom}(H \to G)$ for the set of all homomorphisms from H to G. In particular, we write $\#\mathsf{Hom}(H \to \star)$ for the function that maps a graph G to $\#\mathsf{Hom}(H \to G)$. A bijective homomorphism from a graph H to itself is an *automorphism* and we write $\mathsf{Aut}(H)$ to denote the set of all automorphisms of H.

For a graph H, we define the *average degree* of H as

$$d(H) := \frac{1}{|V(H)|} \cdot \sum_{v \in V(H)} \mathsf{deg}(v).$$

Further, we rely on the *treewidth* of a graph, which is a graph parameter $\mathsf{tw} : \mathcal{G} \to \mathbb{N}$. However, we only work with the treewidth in a black-box manner, and thus we omit the definition and refer the interested reader to the literature, (see for instance [22, Chapter 7]). In particular, we use the following well-known result from extremal graph theory, which relates the treewidth of a graph H to its average degree.

Lemma 2.1 (Folklore, see for instance [10, Corollary 1]). *For any graph H with average degree at least d, we have $\mathsf{tw}(H) \geq \frac{d}{2}$.*

Finally, we also rely on the following celebrated result from extremal graph theory:

Theorem 2.2 (Turán's Theorem, see for instance [46, Section 2.1]). *A graph H with more than $\left(1 - \frac{1}{r}\right) \cdot \frac{1}{2}|V(H)|^2$ edges contains the clique K_{r+1} as a subgraph.*

Graph Properties

A *graph property* Φ is a function from graphs to $\{0,1\}$ such that $\Phi(H) = \Phi(G)$ whenever H and G are isomorphic. We say that a graph H *satisfies* Φ if $\Phi(H) = 1$. Given a positive integer k and a graph property Φ, we write Φ_k for the set of all (isomorphism classes of) graphs with k vertices that satisfy Φ. Furthermore, given a graph G, a positive integer k, and a graph property Φ, we write $\mathsf{IndSub}(\Phi, k \to G)$ for the set of all induced subgraphs with k vertices of G that satisfy Φ. In particular, we write $\#\mathsf{IndSub}(\Phi, k \to \star)$ for the function that maps a graph G to $\#\mathsf{IndSub}(\Phi, k \to G)$.

Given a graph property Φ, we define $\neg\Phi(H) = 1 :\Leftrightarrow \Phi(H) = 0$ as the *negation* of Φ. Furthermore, we define $\overline{\Phi}(H) = 1 :\Leftrightarrow \Phi(\overline{H}) = 1$ as the *inverse* of Φ.[5] We observe the following identities:

Fact 2.3. *For every graph property Φ, graph G and positive integer k, we have*

$$\#\mathsf{IndSub}(\neg\Phi, k \to G) = \binom{|V(G)|}{k} - \#\mathsf{IndSub}(\Phi, k \to G), \ \ and$$

$$\#\mathsf{IndSub}(\overline{\Phi}, k \to G) = \#\mathsf{IndSub}(\Phi, k \to \overline{G}).$$

Proof. The first identity is immediate. For the second one, we observe that

$$\#\mathsf{IndSub}(\overline{\Phi}, k \to G) = \sum_{H \in \overline{\Phi}_k} \#\mathsf{IndSub}(H \to G) = \sum_{\overline{H} \in \Phi_k} \#\mathsf{IndSub}(H \to G)$$

$$= \sum_{\overline{H} \in \Phi_k} \#\mathsf{IndSub}(\overline{H} \to \overline{G}) = \#\mathsf{IndSub}(\Phi, k \to \overline{G}),$$

where we use the equality $\#\mathsf{IndSub}(H \to G) = \#\mathsf{IndSub}(\overline{H} \to \overline{G})$ from [46, Section 5.2.3]. ∎

Fine-Grained and Parameterized Complexity Theory

Given a computable graph property Φ, the problem $\#\textsc{IndSub}(\Phi)$ asks, given a graph G with n vertices and a positive integer k, to compute $\#\mathsf{IndSub}(\Phi, k \to G)$, that is, the number of induced subgraphs of size k in G that satisfy Φ. Note that the problem can be solved by brute-force in time $f(k) \cdot O(n^k)$ by iterating over all subsets of k vertices in G and testing which of the subsets induce a graph that satisfies Φ; the latter part takes time $f(k)$ for some f depending on Φ.

As elaborated in the introduction, our goal is to understand the complexity of $\#\textsc{IndSub}(\Phi)$ for instances with small k and large n. More precisely, we wish to identify the best possible exponent of n in the running time. To this end, we rely on the frameworks of fine-grained and parameterized complexity theory. Regarding the former, we prove conditional lower bounds based on the *Exponential Time Hypothesis* due to Impagliazzo and Paturi [36]:

Conjecture 2.4 (Exponential Time Hypothesis (ETH)). *The problem* 3-SAT *cannot be solved in time* $\exp(o(m))$, *where m is the number of clauses of the input formula.* ∎

Assuming ETH, we are able to prove that the exponent (k) of the brute-force algorithm for $\#\textsc{IndSub}(\Phi)$ cannot be improved significantly for non-trivial monotone properties by establishing that no algorithm with a running time of $f(k) \cdot |V(G)|^{o(k/\sqrt{\log k})}$ for any function f exists.

---

[5] We omit using the word "complement" for graph properties to avoid confusion on whether we mean $\neg\Phi$ or $\overline{\Phi}$.

In the language of parameterized complexity theory, our reductions also yield #W[1]-completeness results, where #W[1] should be considered the parameterized counting equivalent of NP; we provide a rough introduction in what follows and refer the interested reader to references like [22] and [30] for a detailed treatment.

A *parameterized counting problem* is a pair of a function $P : \Sigma^* \to \mathbb{N}$ and a computable parameterization $\kappa : \Sigma^* \to \mathbb{N}$. Examples include the problems #VERTEXCOVER and #CLIQUE which ask, given a graph G and a positive integer k, to compute the number $P(G, k)$ of vertex covers or cliques, respectively, of size k. Both problems are parameterized by the solution size, that is $\kappa(G, k) := k$. Similarly, the problem #INDSUB(Φ) can be viewed as a parameterized counting problem when parameterized by $\kappa(G, k) := k$; we implicitly assume this parameterization of #INDSUB(Φ) in the remainder of this paper.

A parameterized counting problem is called *fixed-parameter tractable* (FPT) if there is a computable function f such that the problem can be solved in time $f(\kappa(x)) \cdot |x|^{O(1)}$, where $|x|$ is the input size.

Given two parameterized counting problems (P, κ) and $(\hat{P}, \hat{\kappa})$, a *parameterized Turing-reduction* from (P, κ) to $(\hat{P}, \hat{\kappa})$ is an algorithm $\mathbb{A}$ that is given oracle access to $\hat{P}$ and, on input x, computes $P(x)$ in time $f(\kappa(x)) \cdot |x|^{O(1)}$ for some computable function f; furthermore, the parameter $\kappa(y)$ of every oracle query posed by $\mathbb{A}$ must be bounded by $g(\kappa(x))$ for some computable function g.

While #VERTEXCOVER is known to be fixed-parameter tractable [30], #CLIQUE is not fixed-parameter tractable, unless ETH fails [13, 14]. Moreover, #CLIQUE is the canonical complete problem for the parameterized complexity class #W[1], see [30]; in particular, we use the following definition of #W[1]-completeness in this work.

▸ **Definition 2.5.** *A parameterized counting problem is* #W[1]-complete *if it is interreducible with* #CLIQUE *with respect to parameterized Turing-reductions.*   ◂

Note that that the absence of an FPT algorithm for #CLIQUE under ETH and the definition of parameterized Turing-reductions yield that #W[1]-complete problems are not fixed-parameter tractable, unless ETH fails, legitimizing the notion of #W[1]-completeness as evidence for (fixed-parameter) intractability. Jerrum and Meeks [37] have shown that #INDSUB(Φ) reduces to #CLIQUE for every computable property Φ with respect to parameterized Turing-reductions. Thus we will only treat the "hardness part" of the #W[1]-completeness results in this paper.

The fine-grained and parameterized complexity of the homomorphism counting problem are the foundation of the lower bounds established in this work: Given a class of graphs $\mathcal{H}$, the problem #HOM($\mathcal{H}$) asks, on input a graph $H \in \mathcal{H}$ and an arbitrary graph G, to compute #Hom($H \to G$). The parameter is given by $|V(H)|$. The following classification shows that, roughly speaking, the complexity of #HOM($\mathcal{H}$) is determined by the treewidth of the graphs in $\mathcal{H}$.

▸ **Theorem 2.6** ([23, 47]). *Let $\mathcal{H}$ denote a recursively enumerable class of graphs. If the treewidth of $\mathcal{H}$ is bounded by a constant, then* #HOM($\mathcal{H}$) *is solvable in polynomial time. Otherwise, the problem is* #W[1]-*complete and cannot be solved in time*

$$f(|V(H)|) \cdot |V(G)|^{o\left(\frac{\operatorname{tw}(H)}{\log \operatorname{tw}(H)}\right)}$$

for any function f, unless ETH fails.

Note that the classification of #HOM($\mathcal{H}$) into polynomial-time and #W[1]-complete cases is explicitly stated and proved in the work of Dalmau and Jonson [23]. However, the conditional lower bound follows only implicitly by a result of Marx [47]. We provide a proof for completeness.

Proof. As discussed, we focus on the lower bound, which follows implicitly from a result of Marx [47] on the complexity of *Partitioned Subgraph Isomorphism*: The problem PARTITIONEDSUB($\mathcal{H}$) asks, given a graph $H \in \mathcal{H}$, an arbitrary graph G and a (not necessarily proper) vertex coloring $c : V(G) \to V(H)$, to *decide* whether there is an injective homomorphism φ from H to G such that $c(\varphi(v)) = v$ for each vertex v of H.

The result of Marx [47, Corollary 6.2] states that for every $\mathcal{H}$ of unbounded treewidth, the problem PARTITIONEDSUB($\mathcal{H}$) cannot be solved in time

$$f(|V(H)|) \cdot |V(G)|^{o(\mathsf{tw}(H)/\log \mathsf{tw}(H))} \tag{3}$$

for any function f, unless ETH fails. Now suppose we are given a graph $H \in \mathcal{H}$, an arbitrary graph G and a coloring $c : V(G) \to V(H)$. We wish to decide whether there is an injective homomorphism φ from H to G such that $c(\varphi(v)) = v$ holds for each vertex v of H. Note first, that we can drop the requirement of φ being injective, as every homomorphism that preserves the coloring is injective. Note further, that without loss of generality, we can assume that c is a homomorphism from G to H: Every edge $\{u, v\}$ of G such that $\{c(u), c(v)\} \notin E(H)$ is irrelevant for finding a homomorphism φ from H to G that preserves the coloring c. Hence, we can delete all of those edges from G. Thus, the problem PARTITIONEDSUB($\mathcal{H}$) is equivalent to the problem CP-HOM($\mathcal{H}$) which asks, given a graph $H \in \mathcal{H}$, an arbitrary graph G, and a homomorphism $c \in \mathsf{Hom}(G \to H)$, to decide whether there is a $\varphi \in \mathsf{Hom}(H \to G)$ such that $c(\varphi(v)) = v$ for each $v \in V(H)$. Finally, it is known that (the counting version of) CP-HOM($\mathcal{H}$) tightly reduces to #HOM($\mathcal{H}$) via the principle of inclusion and exclusion [55, Lemma 2.52] or polynomial interpolation [24, Section 3.2]. Thus the conditional lower bound in Equation (3) holds for #HOM($\mathcal{H}$) as well. ◾

The question whether the lower bound from Theorem 2.6 can be strengthened to $f(|V(H)|) \cdot |V(G)|^{o(\mathsf{tw}(H))}$ is known as "Can you beat treewidth?" and constitutes a major open problem in parameterized complexity theory and an obstruction for tight conditional lower bounds on the complexity of a variety of (parameterized) problems, see for instance [45, 18, 21, 20].

As described in the introduction, the complexity of computing a finite linear combination of homomorphism counts is precisely determined by the complexity of computing the non-vanishing terms. The formal statement is provided subsequently.

Theorem 2.7 (Complexity Monotonicity [12, 20])**.** *Let* $a : \mathcal{G} \to \mathbb{Q}$ *denote a function of finite support and let F denote a graph such that $a(F) \neq 0$. There are a computable function g and a deterministic algorithm $\mathbb{A}$ with oracle access to the function*

$$G \mapsto \sum_{H \in \mathcal{G}} a(H) \cdot \#\mathsf{Hom}(H \to G),$$

and which, given a graph G with n vertices, computes $\#\mathsf{Hom}(F \to G)$ in time $g(a) \cdot n^c$, where c is a constant independent of a. Furthermore, each queried graph has at most $g(a) \cdot n$ vertices. ◾

As observed by Curticapean, Dell and Marx [20], counting induced subgraphs of size k that satisfy Φ is equivalent to computing a finite linear combination of homomorphism counts. Thus, the previous results yield an *implicit* dichotomy for #INDSUB(Φ).

Theorem 2.8 ([20])**.** *Let Φ denote a computable graph property and let k denote a positive integer. There is a unique and computable function $a : \mathcal{G} \to \mathbb{Q}$ of finite support such that*

$$\#\mathsf{IndSub}(\Phi, k \to \star) = \sum_{H \in \mathcal{G}} a(H) \cdot \#\mathsf{Hom}(H \to \star).$$

Furthermore, the problem #INDSUB(Φ) is either fixed-parameter tractable or #W[1]-complete. ◾

Note that the result on $\#\textsc{IndSub}(\Phi)$ in the previous theorem does not concern the fine-grained complexity of the problem. To reveal the latter, it is necessary to understand the support of the function a; we tackle this task in detail in Section 3.

f-Vectors and h-Vectors

It was observed in [56] that there is a close connection between the structure of the simplicial graph complex of edge-monotone properties Φ and the complexity of $\#\textsc{IndSub}(\Phi)$. In this work, we generalize two important topological invariants of simplicial complexes to arbitrary graph properties: The f-vector and the h-vector.

Definition 2.9. *Let Φ denote a graph property, let k denote a positive integer and set $d = \binom{k}{2}$. The f-vector $f^{\Phi,k} = (f_i^{\Phi,k})_{i=0}^d$ of Φ and k is defined by*

$$f_i^{\Phi,k} := \#\{A \subseteq E(K_k) \mid \#A = i \wedge \Phi(K_k[A]) = 1\}, \quad \text{where } i \in \{0,\dots,d\},$$

that is, $f_i^{\Phi,k}$ is the number of edge-subsets of size i of K_k such that the induced graph satisfies Φ.
The h-vector $h^{\Phi,k} = (h_\ell^{\Phi,k})_{\ell=0}^d$ is defined by

$$h_\ell^{\Phi,k} := \sum_{i=0}^{\ell} (-1)^{\ell-i} \cdot \binom{d-i}{\ell-i} \cdot f_i^{\Phi,k}, \quad \text{where } \ell \in \{0,\dots,d\}.$$

As mentioned before, note that those notions of f and h-vectors correspond to the eponymous notions for simplicial (graph) complexes.[6] We omit the definition of the latter as we are only concerned with the generalized notions and refer the interested reader e.g. to [4].

It turns out that the *non-vanishing* of suitable entries $h_\ell^{\Phi,k}$ of the h-vector implies hardness for $\#\textsc{IndSub}(\Phi)$. The result in [56] can be considered as a very restricted special case as it shows that the non-vanishing of the reduced Euler characteristic of the complex (which is equal to the entry $h_d^{\Phi,k}$) implies hardness. On the other hand, for many graph properties it is easy to deduce information about the f-vector (for instance that $f_\ell^{\Phi,k} = 0$ for sufficiently large ℓ with respect to k). We observe that the f and h-vectors of a graph property are related by the so-called the f-polynomial which is again a generalization of the epynomous notion for simplicial complexes:

Definition 2.10. *Let Φ denote a graph property, let k denote a positive integer and set $d = \binom{k}{2}$. The f-polynomial of Φ and k is a univariate polynomial of degree at most d defined as follows:*

$$\mathtt{f}_{\Phi,k}(x) := \sum_{i=0}^{d} f_i^{\Phi,k} \cdot x^{d-i}.$$

As we see in the proof of Lemma 3.5, the entries of the f and h-vectors are given up to combinatorial factors by derivatives of the f-polynomial at 0 and -1. Intuitively, we apply Hermite-Birkhoff interpolation on $\mathtt{f}_{\Phi,k}$ and its derivatives to prove that specific entries of $h^{\Phi,k}$ cannot vanish in case a sufficient number of entries of $f^{\Phi,k}$ do, unless Φ is trivially false on k-vertex graphs.

---

[6] In some parts of the literature, the f-vector comes with an index shift of -1 due to the topological interpretation of simplicial complexes.

Hermite-Birkhoff Interpolation and Pólya's Theorem

While a univariate polynomial of degree d is uniquely determined by $d+1$ evaluations in pairwise different points, the problem of *Hermite-Birkhoff interpolation* asks under which conditions we can uniquely recover the polynomial if we instead impose conditions on the derivatives of the polynomial at m distinct points. Following the notation of Schoenberg [58], the problem is formally expressed as follows. Given a matrix $E = (\varepsilon_{ij}) \in \{0,1\}^{m \times d+1}$ where $i \in \{1,\ldots,m\}$ and $j \in \{0,\ldots d\}$, as well as reals $x_1 < \cdots < x_m$, the goal is to find a polynomial $\mathtt{f}$ of degree at most d such that for all i and j with $\varepsilon_{ij} = 1$ we have

$$\mathtt{f}^{(j)}(x_i) = 0$$

Here, $\mathtt{f}^{(j)}$ denotes the j-th derivative of $\mathtt{f}$. In particular, we are interested under which conditions on the matrix E, the zero polynomial is the *unique* solution. In this case, E is called *poised*. It turns out that the case $m = 2$ is sufficient for our purposes; fortunately, this case was fully solved by Pólya:

Theorem 2.11 (Pólya's Theorem [51, 58]). *Let E be defined as above with $m = 2$. Suppose that $\sum_{i,j} \varepsilon_{ij} = d+1$ and for every $j \in \{0,\ldots,d\}$ set*

$$M_j := \sum_{i=0}^{j} \varepsilon_{1,i} + \varepsilon_{2,i}.$$

Then, E is poised if and only if $M_j \geq j+1$ holds true for all $j \in \{0,\ldots,d-1\}$.

3   Homomorphism Vectors of Graph Properties

In this section we discuss and prove our main technical result:

Theorem 3.1. *Let Φ denote a computable graph property, let k denote a positive integer, and let w denote the Hamming weight of the f-vector $f^{\Phi,k}$. Suppose that Φ is not trivially false on k-vertex graphs. Then there is a unique and computable function $a : \mathcal{G} \to \mathbb{Q}$ of finite support such that*

$$\#\mathsf{IndSub}(\Phi, k \to \star) = \sum_{H \in \mathcal{G}} a(H) \cdot \#\mathsf{Hom}(H \to \star),$$

satisfying that there is a graph K on k vertices and at least $\binom{k}{2} - w + 1$ edges such that $a(K) \neq 0$.

First, recall from Theorem 2.8 that for any computable graph property Φ and positive integer k, there is a unique computable function $a : \mathcal{G} \to \mathbb{Q}$ (with finite support) satisfying

$$\#\mathsf{IndSub}(\Phi, k \to \star) = \sum_{H \in \mathcal{G}} a(H) \cdot \#\mathsf{Hom}(H \to \star). \tag{4}$$

Now, for the remainder of the section, fix a (computable) graph property Φ and a positive integer k (and thus the function a). This allows us to simplify the notation for the f and h-vectors, as well as for the f-polynomial: We write $f := f^{\Phi,k}$, $h := h^{\Phi,k}$, and $\mathtt{f} := \mathtt{f}_{\Phi,k}$. Furthermore, we set $d := \binom{k}{2}$ and we write $\mathcal{H}_i$ for the set of all graphs on k vertices and with i edges.

Next, we define the vector $\tilde{h}_i$ as

$$\tilde{h}_i := \sum_{K \in \mathcal{H}_i} a(K), \text{ where } i \in \{0,\ldots,d\},$$

that is, the i-th entry of $\tilde{h}$ is the sum of the coefficients of graphs with k vertices and i edges in Equation (4). Now we establish the aforementioned connection between the coefficients of Equation (4) and the h-vector of the property Φ.

Lemma 3.2. *We have $k! \cdot \tilde{h} = h$.*

Note that as a consequence, the h-vector of a simplicial graph complex is determined by the coefficients of its associated linear combination of homomorphisms.

Proof. Given two graphs H and H' on k vertices each, we write $\#\{H' \supseteq H\}$ for the number of possibilities of adding edges to H such that (a graph isomorphic to) H' is obtained. We start with the following claim which was implicitly shown in [56]; we include a proof for completeness.

Claim 3.3. Let K denote a graph with k vertices and define a as in Equation (4). We have

$$a(K) = \sum_{H \in \Phi_k} \#\mathsf{Aut}(H)^{-1} \cdot (-1)^{\#E(K) - \#E(H)} \cdot \#\{K \supseteq H\}.$$

Proof. Fix a graph K with k vertices. Using the standard transformations from strong embeddings to embeddings and from embeddings to homomorphisms (see for instance Lovász [46]), we obtain the following:[7]

$$\begin{aligned}
&\#\mathsf{IndSub}(\Phi, k \to \star) \\
&\quad = \sum_{H \in \Phi_k} \#\mathsf{Aut}(H)^{-1} \sum_{H' \in \mathcal{G}} (-1)^{\#E(H') - \#E(H)} \cdot \#\{H' \supseteq H\} \sum_{\rho \geq \emptyset} \mu(\emptyset, \rho) \cdot \#\mathsf{Hom}(H'/\rho \to \star),
\end{aligned}$$

where μ denotes the Möbius function and the rightmost sum ranges over the partition lattice of the set of vertices of H'. Furthermore, H'/ρ is the quotient graph obtained by identifying vertices of H' along the partition ρ. In particular, $H'/\emptyset = H'$. We omit the details, which can be found in [46, 56], as we only need that H'/ρ has strictly less than k vertices for all $\rho > \emptyset$ and that $\mu(\emptyset, \emptyset) = 1$. This allows us to rewrite the previous equation as follows:

$$\begin{aligned}
&\#\mathsf{IndSub}(\Phi, k \to \star) \\
&\quad = \sum_{H \in \Phi_k} \#\mathsf{Aut}(H)^{-1} \sum_{H' \in \mathcal{G}} (-1)^{\#E(H') - \#E(H)} \#\{H' \supseteq H\} \cdot \#\mathsf{Hom}(H' \to \star) + R(\Phi_k),
\end{aligned}$$

where the remainder $R(\Phi_k)$ does not depend on any numbers $\#\mathsf{Hom}(F \to \star)$ for graphs F with k vertices. In particular, reordering and grouping the coefficients of $\#\mathsf{Hom}(K \to \star)$ yields the claim.   ⌟

Next, we investigate the term $\sum_{K \in \mathcal{H}_\ell} \#\{K \supseteq H\}$.

Claim 3.4. Let $\ell \in \{0, \dots, d\}$ denote an integer and let H denote a graph with k vertices and at most ℓ edges. Then, we have

$$\sum_{K \in \mathcal{H}_\ell} \#\{K \supseteq H\} = \binom{d - \#E(H)}{\ell - \#E(H)}.$$

Proof. Any extension from the graph H to a graph with ℓ edges has to add $\ell - \#E(H)$ edges to H; there are exactly $d - \#E(H)$ possible choices for these $\ell - \#E(H)$ edges. Hence the claim follows from basic combinatorics.   ⌟

---

[7] This step is done explicitly in [56].

Now, fix an $\ell \in \{0, \ldots, d\}$; we proceed to show that $k! \cdot \tilde{h}_\ell = h_\ell$, which proves the lemma. To that end, from the definition of $\tilde{h}$, we obtain

$$
\begin{aligned}
k! \cdot \tilde{h}_\ell &= k! \cdot \sum_{K \in \mathcal{H}_\ell} a(K) \\
&= k! \cdot \sum_{K \in \mathcal{H}_\ell} \sum_{H \in \Phi_k} \#\mathsf{Aut}(H)^{-1} \cdot (-1)^{\ell - \#E(H)} \cdot \#\{K \supseteq H\} \\
&= \sum_{H \in \Phi_k} k! \cdot \#\mathsf{Aut}(H)^{-1} \cdot (-1)^{\ell - \#E(H)} \sum_{K \in \mathcal{H}_\ell} \#\{K \supseteq H\},
\end{aligned}
$$

where the second equality holds due to Claim 3.3. Now observe that $\#\{K \supseteq H\} = 0$ if H has more edges than K. Thus we see that

$$
\begin{aligned}
k! \cdot \tilde{h}_\ell &= \sum_{\substack{H \in \Phi_k \\ \#E(H) \leq \ell}} k! \cdot \#\mathsf{Aut}(H)^{-1} \cdot (-1)^{\ell - \#E(H)} \sum_{K \in \mathcal{H}_\ell} \#\{K \supseteq H\} \\
&= \sum_{\substack{H \in \Phi_k \\ \#E(H) \leq \ell}} k! \cdot \#\mathsf{Aut}(H)^{-1} \cdot (-1)^{\ell - \#E(H)} \cdot \binom{d - \#E(H)}{\ell - \#E(H)},
\end{aligned}
$$

where the last equality holds due to Claim 3.4. Next we use the fact that $k!$ is the order of the symmetric group Sym_k: For any graph H in the above sum, choose a set A of edges of the labeled complete graph K_k on k vertices such that the corresponding subgraph $K_k[A]$ is isomorphic to H. The group Sym_k acts on the vertices and thus on the edges of K_k. By the definition of a graph automorphism, the stabilizer of the set A has exactly $\#\mathsf{Aut}(H)$ elements.

Now observe that the orbit of A under Sym_k is the collection of all sets A' such that $K_k[A'] \cong H$. Therefore, by the Orbit Stabilizer Theorem, we have

$$
k! \cdot \#\mathsf{Aut}(H)^{-1} = \#\{A' \subseteq E(K_k) \mid K_k[A'] \cong H\}.
$$

Hence we can conclude that

$$
\begin{aligned}
k! \cdot \tilde{h}_\ell &= \sum_{\substack{H \in \Phi_k \\ \#E(H) \leq \ell}} \#\{A \subseteq E(K_k) \mid K_k[A] \cong H\} \cdot (-1)^{\ell - \#E(H)} \cdot \binom{d - \#E(H)}{\ell - \#E(H)} \\
&= \sum_{i=0}^{\ell} \sum_{\substack{H \in \Phi_k \\ \#E(H) = i}} \#\{A \subseteq E(K_k) \mid K_k[A] \cong H\} \cdot (-1)^{\ell - i} \cdot \binom{d - i}{\ell - i} \\
&= \sum_{i=0}^{\ell} \#\{A \subseteq E(K_k) \mid \#A = i \wedge \Phi(K_k[A]) = 1\} \cdot (-1)^{\ell - i} \cdot \binom{d - i}{\ell - i} \\
&= \sum_{i=0}^{\ell} f_i \cdot (-1)^{\ell - i} \cdot \binom{d - i}{\ell - i} = h_\ell,
\end{aligned}
$$

completing the proof.   ◾

In the next step, we use Pólya's Theorem to prove that the Hamming weight of the f-vector determines an index β of the h-vector such that at least one entry of h with index at least β is non-zero. By Lemma 3.2 the same then follows for $\tilde{h}$.

Lemma 3.5. *Let w denote the Hamming weight of f and set $\beta = d - w$. If Φ is not trivially false on k-vertex graphs then at least one of the values $h_d, \ldots, h_{\beta+1}$ is non-zero.*

Proof. Recall the definition of the f-polynomial $\mathbf{f}(x) = \sum_{i=0}^{d} f_i \cdot x^{d-i}$ and observe that

$$\mathbf{f}^{(j)}(x) = \sum_{i=0}^{d-j} f_i \cdot (d-i)_j \cdot x^{d-j-i}.$$

By $j_j = j!$, we immediately obtain $\mathbf{f}^{(j)}(0) = f_{d-j} \cdot j!$. Therefore, by assumption, we have $\mathbf{f}^{(j)}(0) = 0$ for $\beta + 1$ many indices j.

Furthermore, we see that

$$\mathbf{f}^{(j)}(-1) = \sum_{i=0}^{d-j} f_i \cdot (d-i)_j \cdot (-1)^{d-j-i} \qquad\qquad = j! \cdot \sum_{i=0}^{d-j} f_i \cdot \binom{d-i}{j} \cdot (-1)^{d-j-i}$$

$$= j! \cdot \sum_{i=0}^{d-j} f_i \cdot \binom{d-i}{(d-j)-i} \cdot (-1)^{d-j-i} \quad = j! \cdot h_{d-j}.$$

Now assume for the sake of contradiction that each of the values $h_d, \ldots, h_{\beta+1}$ is zero. Consequently, $\mathbf{f}^{(j)}(-1) = 0$ for $j = 0, \ldots, w - 1$. Interpreting those evaluations of the derivatives of the f-polynomial as an instance of Hermite-Birkhoff interpolation, the corresponding matrix E looks as follows:[8]

$$
\begin{array}{cccccccc}
0 & 1 & 2 & \ldots & w-1 & w & \ldots & d
\end{array}
$$

$$
\begin{pmatrix}
1 & 1 & 1 & \ldots & 1 & 0 & \ldots & 0 \\
\varepsilon_{20} & \varepsilon_{21} & \varepsilon_{22} & \ldots & \varepsilon_{2(w-1)} & \varepsilon_{2w} & \ldots & \varepsilon_{2d}
\end{pmatrix}
$$

In particular, at least $\beta + 1 = d + 1 - w$ of the values ε_{2j} are 1; As $\beta + 1$ and w sum up to $d + 1$, we can easily verify that the conditions of Pólya's Theorem (Theorem 2.11) are satisfied: Let us modify E by arbitrarily choosing *precisely* $\beta + 1$ of the $\varepsilon_{2,j}$ that are 1 and set the others to 0, and call the resulting matrix $\hat{E}$. We then have both $M_j \geq j + 1$ (for all $j \in \{0, \ldots, d-1\}$) and the first and second row of $\hat{E}$ sum up to *precisely* $d + 1$. Hence the matrix $\hat{E}$ is poised, that is, the only polynomial of degree at most d that satisfies the corresponding instance of Hermite-Birkhoff interpolation is the zero polynomial. As we obtained $\hat{E}$ from E just by ignoring some vanishing conditions, the same conclusion is true for E and thus $\mathbf{f} = 0$ is the unique solution. This, however, contradicts the fact that the property Φ is not trivially false on k-vertex graphs, completing the proof. ◢

Combining Lemmas 3.2 and 3.5 yields our main technical result, which we restate here for convenience.

Theorem 3.1. *Let Φ denote a computable graph property, let k denote a positive integer, and let w denote the Hamming weight of the f-vector $f^{\Phi,k}$. Suppose that Φ is not trivially false on k-vertex graphs. Then there is a unique and computable function $a : \mathcal{G} \to \mathbb{Q}$ of finite support such that*

$$\#\mathsf{IndSub}(\Phi, k \to \star) = \sum_{H \in \mathcal{G}} a(H) \cdot \#\mathsf{Hom}(H \to \star),$$

satisfying that there is a graph K on k vertices and at least $\binom{k}{2} - w + 1$ edges such that $a(K) \neq 0$.

---

[8] Recall that an entry 1 in the matrix E represents an evaluation $\mathbf{f}^{(j)}(-1) = 0$ in the first row and an evaluation $\mathbf{f}^{(j)}(0) = 0$ in the second row.

Proof. Set $d = \binom{k}{2}$ and $\beta = d - w$. By Equation (4) the function a exists and is computable and has a finite support. Now, Lemma 3.5 implies that at least one of the values $h_d^{\Phi,k}, \ldots, h_{\beta+1}^{\Phi,k}$ is non-zero and thus, by Lemma 3.2, at least one of the values $\tilde{h}_d, \ldots, \tilde{h}_{\beta+1}$ is non-zero as well. Next, observe that $\tilde{h}_i = \sum_{K \in \mathcal{H}_i} a(K)$ for all $i \in \{0, \ldots, d\}$, where $\mathcal{H}_i$ is the set of all graphs on k vertices and i edges. In particular, $\tilde{h}_i \neq 0$ implies that $a(K) \neq 0$ for at least one $K \in \mathcal{H}_i$, yielding the claim. ⌟

4 A Classification of #IndSub(Φ) by the Hamming Weight of the f-Vectors

In this section, we derive a general hardness result for #INDSUB(Φ) based on the Hamming weight of the f-vector. In a sense, we "black-box" Theorem 3.1; using the resulting classification, we establish first hardness results and almost tight conditional lower bounds for a variety of families of graph properties.

However, note that taking a closer look at the number of edges of the graphs with non-vanishing coefficients (as provided by Theorem 3.1) often yields improved, sometimes even matching conditional lower bounds; we defer the treatment of the refined analysis to Section 5.

In what follows, we write $\mathcal{K}(\Phi)$ for the set of all k such that Φ_k is non-empty.

Theorem 4.1. *Let Φ denote a computable graph property and suppose that the set $\mathcal{K}(\Phi)$ is infinite. Let $\beta : \mathcal{K}(\Phi) \to \mathbb{Z}_{\geq 0}$ denote the function that maps k to $\binom{k}{2} - \mathsf{hw}(f^{\Phi,k})$. If $\beta(k) \in \omega(k)$ then the problem #INDSUB(Φ) is #W[1]-complete and cannot be solved in time*

$$g(k) \cdot |V(G)|^{o((\beta(k)/k)/\log(\beta(k)/k))}$$

for any function g, unless ETH fails. The same statement holds for #INDSUB($\overline{\Phi}$) and #INDSUB($\neg\Phi$).

Note that the condition of $\mathcal{K}(\Phi)$ being infinite is necessary for hardness: Otherwise there is a constant c such that we can output 0 whenever $k \geq c$ and solve the problem by brute-force if $k < c$, yielding an algorithm with a polynomial running time. Note further that the $(\log(\beta(k)/k))^{-1}$-factor in the exponent is related to the question of whether it is possible to "beat treewidth" [47].[9] In particular, if the factor of $(\log \mathsf{tw}(H))^{-1}$ in Theorem 2.6 can be dropped, then all further results in this section can be strengthened to yield tight conditional lower bounds under ETH.

Proof. By Theorem 3.1, for each $k \in \mathcal{K}(\Phi)$ we obtain a graph H_k with k vertices and at least $\beta(k)$ edges such that $a(H_k) \neq 0$, where a is the function in Equation (4). The average degree of H_k satisfies

$$d(H_k) = \frac{1}{k} \cdot \sum_{v \in V(H_k)} \deg(v) = \frac{2|E(H_k)|}{k} \geq \frac{2\beta(k)}{k},$$

where the second equality is due to the Handshaking Lemma. By Lemma 2.1, we thus obtain that $\mathsf{tw}(H_k) \geq \frac{\beta(k)}{k}$, which is unbounded as $\beta(k) \in \omega(k)$ by assumption.

Now let $\mathcal{H}$ denote the set of all graphs H_k for $k \in \mathcal{K}(\Phi)$. By Theorem 2.6, we obtain that #HOM($\mathcal{H}$) is #W[1]-complete and cannot be solved in time

$$g(k) \cdot |V(G)|^{o((\beta(k)/k)/\log(\beta(k)/k))}$$

for any function g, unless ETH fails. Further, by Complexity Monotonicity (Theorem 2.7), the same is true for #INDSUB(Φ) as well. Finally, we use Fact 2.3 to obtain the same result for #INDSUB($\overline{\Phi}$) and #INDSUB($\neg\Phi$); completing the proof. ⌟

---

[9] See Theorem 2.6 and its discussion.

4.1   Low Edge-Densities and Sparse Graph Properties

As a first application of Theorem 4.1, we consider properties Φ that satisfy

$$\mathsf{hw}(f^{\Phi,k}) \in o(k^2).$$

We say that such a property Φ has a *low edge-densities*. Properties with low edge-density subsume, for example, exclusion of a set of fixed minors such as planarity. They have been studied by Jerrum and Meeks [38], where they show that $\#\mathrm{INDSUB}(\Phi)$ is $\#\mathsf{W}[1]$-complete for these properties. However, their proof uses Ramsey's Theorem and thus only establishes an implicit conditional lower bound of $g(k) \cdot |V(G)|^{o(\log k)}$. In contrast, we achieve the following, almost tight lower bound:

Theorem 4.2. *Let Φ denote a computable graph property with low edge-densities. Suppose that the set $\mathcal{K}(\Phi)$ is infinite. Then $\#\mathrm{INDSUB}(\Phi)$ is $\#\mathsf{W}[1]$-complete and cannot be solved in time*

$$g(k) \cdot |V(G)|^{o(k/\log k)}$$

for any function g, unless ETH fails. The same is true for $\#\mathrm{INDSUB}(\overline{\Phi})$ and $\#\mathrm{INDSUB}(\neg\Phi)$.

Proof. If Φ has low edge-densities, then we have $\beta(k) = \binom{k}{2} - \mathsf{hw}(f^{\Phi,k}) \in \Theta(k^2)$. Thus

$$o\left(\frac{\beta(k)/k}{\log(\beta(k)/k)}\right) = o\left(k/\log k\right).$$

The claim hence follows by Theorem 4.1.   ◾

The previous result applies, in particular, to sparse properties. In Section 5 we show, that a refined analysis based on Turán's Theorem as well as Theorem 3.1 establishes a tight conditional lower bound for sparse properties Φ that additionally satisfy a density condition on $\mathcal{K}(\Phi)$; the combination of those two results then implies Main Theorem 2.

4.2   Graph Properties Depending Only on the Number of Edges

Jerrum and Meeks [38, 39] asked whether $\#\mathrm{INDSUB}(\Phi)$ is $\#\mathsf{W}[1]$-complete whenever Φ is non-trivial infinitely often and only depends on the number of edges of a graph, that is,

$$\forall H_1, H_2 : |E(H_1)| = |E(H_2)| \Rightarrow \Phi(H_1) = \Phi(H_2).$$

We answer this question affirmatively, even for properties that can depend both on the number of edges and vertices of the graph, and additionally provide an almost tight conditional lower bound:

Theorem 4.3. *Let Φ denote a computable graph property that only depends on the number of edges and the number of vertices of a graph. If Φ_k is non-trivial only for finitely many k then $\#\mathrm{INDSUB}(\Phi)$ is fixed-parameter tractable. Otherwise, $\#\mathrm{INDSUB}(\Phi)$ is $\#\mathsf{W}[1]$-complete and cannot be solved in time*

$$g(k) \cdot |V(G)|^{o(k/\log k)}$$

for any function g, unless ETH fails.

Note that Theorem 4.3 is also true for $\#\mathrm{INDSUB}(\overline{\Phi})$ and $\#\mathrm{INDSUB}(\neg\Phi)$, as $\neg\Phi$ and $\overline{\Phi}$ depend only on the number of edges and vertices of a graph if and only if Φ does.

Proof. First, assume that Φ_k is non-trivial only for finitely many k. Then, there is a constant c such that for every $k > c$, the property Φ_k is either trivially true or trivially false. Hence, given as input a graph G and an integer k, we check whether $k \leq c$. If this is the case, we solve the problem by brute-force. Otherwise, we check whether Φ_k is trivially false or trivially true.[10] If Φ_k is false, we output 0; otherwise we output $\binom{n}{k}$. It is immediate that this algorithm yields fixed-parameter tractability.

Now assume that Φ_k is non-trivial for infinitely many k. Since for Φ_k we fix the number of vertices to be k, by assumption Φ_k only depends on the number of edges of a graph. Thus, we have

$$\mathsf{hw}(f^{\neg\Phi,k}) = \binom{k}{2} - \mathsf{hw}(f^{\Phi,k}). \tag{5}$$

Hence, set

$$\hat{\Phi}_k := \begin{cases} \Phi_k & \text{if } \mathsf{hw}(f^{\Phi,k}) \leq \frac{1}{2}\binom{k}{2} \\ \neg\Phi_k & \text{if } \mathsf{hw}(f^{\Phi,k}) > \frac{1}{2}\binom{k}{2}. \end{cases}$$

We observe that, by assumption, $\mathcal{K}(\hat{\Phi})$ is infinite, and by Fact 2.3 the problems #INDSUB(Φ) and #INDSUB$(\hat{\Phi})$ are equivalent. By definition and by Equation (5), we see that $\mathsf{hw}(f^{\hat{\Phi},k}) \leq \frac{1}{2}\binom{k}{2}$ and therefore $\beta(k) = \binom{k}{2} - \mathsf{hw}(f^{\hat{\Phi},k}) \in \Theta(k^2)$. Thus, we have

$$o\left(\frac{\beta(k)/k}{\log(\beta(k)/k)}\right) = o\left(k/\log k\right).$$

The claim now follows by Theorem 4.1. ◾

4.3 Monotone Graph Properties

Recall that a property Φ is called monotone if it is closed under taking subgraphs. The decision version of #INDSUB(Φ), that is, deciding whether there is an induced subgraph of size k that satisfies Φ is known to be W[1]-complete if Φ is monotone. Further, Φ is non-trivial and $\mathcal{K}(\Phi)$ is infinite (this follows implicitly by a result of Khot and Raman [43]). However, as the reduction of Khot and Raman is not parsimonious, the reduction does not yield #W[1]-completeness of the counting version. More importantly, the proof of Khot and Raman uses Ramsey's Theorem and thus only implies a conditional lower bound of $g(k) \cdot |V(G)|^{o(\log k)}$. Using our main result, we achieve a much stronger and almost tight lower bound under ETH.

Theorem 4.4. *Let Φ denote a computable graph property that is monotone and non-trivial. Suppose that $\mathcal{K}(\Phi)$ is infinite. Then #INDSUB(Φ) is #W[1]-complete and cannot be solved in time*

$$g(k) \cdot |V(G)|^{o(k/\log k)}$$

for any function g, unless ETH fails. The same is true for #INDSUB$(\overline{\Phi})$ and #INDSUB$(\neg\Phi)$.

Proof. As Φ is non-trivial, there is a graph F such that $\Phi(H)$ is false for every H that contains F as a (not necessarily induced) subgraph. Set $r = |V(F)|$ and fix $k \in \mathcal{K}(\Phi)$. By Turáns Theorem (Theorem 2.2) we have that every graph H on k vertices with more than $\left(1 - \frac{1}{r}\right) \cdot \frac{k^2}{2}$ edges contains the clique K_{r+1} and thus F as a subgraph. Consequently, Φ is false on every graph with k vertices and more than $\left(1 - \frac{1}{r}\right) \cdot \frac{k^2}{2}$ edges. Therefore, we have

$$\beta(k) = \binom{k}{2} - \mathsf{hw}(f^{\Phi,k}) \geq \binom{k}{2} - \left(1 - \frac{1}{r}\right) \cdot \frac{k^2}{2} = \frac{k^2}{2r} - \frac{k}{2} \in \Omega(k^2).$$

---

[10] This step is the reason why we only get fixed-parameter tractability and not necessarily polynomial-time tractability.

Thus $\beta(k) \in \Theta(k^2)$ and we conclude that

$$o\left(\frac{\beta(k)/k}{\log(\beta(k)/k)}\right) = o\left(k/\log k\right).$$

The claim hence follows by Theorem 4.1. ◢

5   Refined Lower Bounds and Clique-Minors

Recall that the lower bounds of the previous section become tight if it is impossible to "beat treewidth", that is, if the $(\log k)^{-1}$ factor in the exponent of Theorem 2.6 can be dropped. In this section, we show that the lower bounds of the previous section can also be refined—and in case of sparse properties even be made tight—without the latter assumption. This requires relying on two results from extremal graph theory on forbidden cliques and clique-minors. The first one is Turán's Theorem, which we have seen already. The second one is a consequence of the Kostochka-Thomason-Theorem:

Theorem 5.1 ([44, 63])**.** *There is a constant $c > 0$ such that every graph H with an average degree of at least $ct\sqrt{\log t}$ contains the clique K_t as a minor.*

Note that Theorem 5.1 is often stated in terms of the number of edges of a graph, instead of its average degree. However, due to the Handshaking-Lemma, both statements are equivalent.

Roughly speaking, we combine the Kostochka-Thomason-Theorem with Theorem 3.1 to find graphs with large clique-minors in the linear combination of homomorphisms associated with a graph property Φ as given by Equation (4). This then allows us to derive hardness by a reduction from the problem of *finding* cliques, instead of relying on Theorem 2.6.

In what follows, we say that a subset $\mathcal{K}$ of the natural numbers is *dense* if there is a constant $\ell \geq 1$ such that for all $n \in \mathbb{N}_{>0}$ there is a $k \in \mathcal{K}$ that satisfies $n \leq k \leq \ell n$. Now recall that $\mathcal{K}(\Phi)$ is the set of all k such that Φ_k is not empty. The lower bounds for #INDSUB(Φ) in the current section require the set $\mathcal{K}(\Phi)$ to be dense. Roughly speaking, this is to exclude artificial properties (such as for instance $\Phi(H) = 1$ if and only if H is an independent set and $|V(H)| = 2 \uparrow m$ for some $m \in \mathbb{N}$, where $2 \uparrow m$ is the m-fold exponential tower of 2). While the latter property satisfies the conditions of Theorem 4.1, we cannot construct a *tight* reduction to #INDSUB(Φ) as the only non-trivial oracle queries satisfy $k = 2 \uparrow m$ for some $m \in \mathbb{N}$. Fortunately, monotone properties exclude such artificial properties:

Lemma 5.2. *Let Φ denote a non-trivial monotone graph property such that $\mathcal{K}(\Phi)$ is infinite. Then $\mathcal{K}(\Phi)$ is the set of all positive integers and thus dense.*

Proof. Fix a $n \in \mathbb{N}_{>0}$. As $\mathcal{K}(\Phi)$ is infinite, there is a $k \geq n$ in $\mathcal{K}(\Phi)$. Thus there is a graph $H \in \Phi_k$. Now delete $k - n$ arbitrary vertices of H and call the resulting graph H'. As Φ is monotone, we have that $H' \in \Phi_n$ and hence $n \in \mathcal{K}(\Phi)$ as well. ◢

The following technical lemma is the basis for the lower bounds in this section; recall that the problem #HOM($\mathcal{H}$) asks, given a graph $H \in \mathcal{H}$ and an arbitrary graph G, to compute the number of homomorphisms from H to G.

Lemma 5.3. *Let $r \geq 1$ denote a constant and let $\mathcal{H}$ denote a decidable class of graphs. Further, let $\mathcal{K}(\mathcal{H})$ denote the set of all positive integers k such that there is a graph $H \in \mathcal{H}$ with k vertices and at least $k^2/2r - k/2$ edges. If $\mathcal{K}(\mathcal{H})$ is dense, then #HOM($\mathcal{H}$) cannot be solved in time*

$$f(|V(H)|) \cdot |V(G)|^{o(|V(H)|/\sqrt{\log |V(H)|})}$$

for any function f, unless ETH fails.

Proof. We construct a tight reduction from the problem CLIQUE, which asks, given a graph G and a parameter $\hat{k} \in \mathbb{N}_{>0}$, to *decide* whether there is a clique of size $\hat{k}$ in G. It is known that CLIQUE cannot be solved in time $\hat{f}(\hat{k}) \cdot |V(G)|^{o(\hat{k})}$ for any function $\hat{f}$, unless ETH fails [13, 14].

Now assume there is an algorithm $\mathbb{A}$ that solves #INDSUB(Φ) in time

$$f(|V(H)|) \cdot |V(G)|^{o(|V(H)|/\sqrt{\log |V(H)|})}$$

for some function f. We use $\mathbb{A}$ to solve CLIQUE in time $\hat{f}(\hat{k}) \cdot |V(G)|^{o(\hat{k})}$ for some function $\hat{f}$.

As $\mathcal{K}(\mathcal{H})$ is dense, there is a constant $\ell \geq 1$ such that for all $\hat{k} \in \mathbb{N}_{>0}$ there is a $k \in \mathcal{K}(\mathcal{H})$ such that $\hat{k} \leq k \leq \ell\hat{k}$. Given a graph G with n vertices and $\hat{k} \in \mathbb{N}_{>0}$, we construct the graph G' as follows: We first search for a $k \in \mathcal{K}(\mathcal{H})$ that satisfies $\hat{k} \leq k \leq \ell\hat{k}$. Note that finding k is computable as $\mathcal{K}(\mathcal{H})$ is decidable. Then, we add $k - \hat{k}$ "fresh" vertices to G and add edges between all pairs of new vertices and between all pairs of a new vertex and an old vertex.

Next, let c denote the constant from Theorem 5.1 and set

$$h(k) := cr' \cdot \sqrt{\log(k/cr')} \, ,$$

where $r' := \max\{1/c, r + 1\}$. Now, we construct a graph $\hat{G}$ from G' as follows: The vertices of $\hat{G}$ are the $h(k)$-cliques of G',[11] and two vertices C_1 and C_2 of $\hat{G}$ are made adjacent if all edges between vertices in C_1 and vertices in C_2 are present in G'. Note that $\hat{G}$ has $O(n^{h(k)})$ vertices and can be constructed in time $g(\hat{k}) \cdot O(n^{h(k)})$ for some computable function g.

⌐ **Claim 5.4.**   The graph G contains a clique of size $\hat{k}$ if and only if $\hat{G}$ contains a clique of size $k/h(k)$.

Proof. Let C denote a $\hat{k}$-clique in G. Then we obtain a k-clique in G' by adding the fresh $k - \hat{k}$ vertices to C. Next, partition C in blocks of size $h(k)$. Each block will be a vertex of $\hat{G}$ and all corresponding vertices are pairwise adjacent by the construction of $\hat{G}$. As there are $k/h(k)$ many blocks, we found the desired clique in $\hat{G}$.

For the other direction, let $\hat{C}$ denote a $k/h(k)$-clique in $\hat{G}$. By the definition of $\hat{G}$, each vertex of $\hat{C}$ corresponds to a clique of size $h(k)$ in G'. Furthermore, two different of those cliques cannot share a common vertex as the corresponding vertices in $\hat{G}$ are adjacent—recall that we do not allow self-loops. Consequently, the union of the $k/h(k)$ many cliques constitutes a clique of size k in G'. Finally, at most $k - \hat{k}$ of the vertices of this clique can be fresh vertices, and thus G contains a clique of size (at least) $\hat{k}$.

⌟

Next, we search for a graph $H \in \mathcal{H}$ with k vertices and at least $k^2/2r - k/2$ edges. By assumption, this can be done in time $g'(k)$ for some computable function g as $\mathcal{H}$ is decidable. Now we see that

$$d(H) = \frac{1}{k} \cdot \sum_{v \in V(H)} \deg(v) = \frac{2|E(H)|}{k} \geq \frac{k}{r} - 1 \, .$$

---

[11] Formally, we have to round $h(k)$ and later $k/h(k)$. For the sake of readability, we assume that all logs and fractions yield integers, but we point out that this might require to find a $\lceil k/h(k) \rceil$-clique at the end of the proof, while the oracle can only determine the existence of a $\lfloor k/h(k) \rfloor$-clique. However, using the latter, we can easily decide whether there is a $\lceil k/h(k) \rceil$-clique by checking for each vertex whether its neighbourhood contains a $\lfloor k/h(k) \rfloor$-clique.

Set $t = \frac{k}{h(k)}$. For k large enough,[12] we have

$$ct\sqrt{\log t} = c \cdot \frac{k}{cr' \cdot \sqrt{\log(k/cr')}} \cdot \sqrt{\log(k/cr') - \log\sqrt{\log(k/cr')}}$$

$$\leq c \cdot \frac{k}{cr' \cdot \sqrt{\log(k/cr')}} \cdot \sqrt{\log(k/cr')} = \frac{k}{r'} \leq \frac{k}{r} - 1 \leq d(H)$$

Theorem 5.1 thus implies that K_t is a minor of H. Furthermore, it is known that, whenever a graph F is a minor of a graph H, there is a tight reduction from counting homomorphisms from F to counting homomorphisms from H — see for instance [55, Chapter 2.5] and Section 3 in the full version[13] of [24]. Consequently, we can use the algorithm $\mathbb{A}$ to compute the number of homomorphisms from K_t to $\hat{G}$. By assumption on $\mathbb{A}$, this takes time at most

$$f(|V(H)|) \cdot |V(\hat{G})|^{o(|V(H)|/\sqrt{\log|V(H)|})} = f(k) \cdot (n^{h(k)})^{o(k/\sqrt{\log k})} = f(k) \cdot n^{o(k)} \,,$$

where the latter holds as $h(k) \in \Theta(\sqrt{\log k})$—recall that r' and c are constants. However, it is easy to see that $\hat{G}$ contains a t-clique if and only if the number of homomorphisms from K_t to $\hat{G}$ is at least 1. By Claim 5.4, this is equivalent to G having a clique of size $\hat{k}$. As $k \in O(\hat{k})$ (recall that ℓ is a constant), the overall running time is bounded by

$$g(\hat{k}) \cdot O(n^{h(k)}) + g'(k) + f(k) \cdot n^{o(k)} \leq \hat{f}(\hat{k}) \cdot n^{o(\hat{k})}$$

for $\hat{f}(\hat{k}) := g(\hat{k}) + g'(\ell\hat{k}) + f(\ell\hat{k})$. This yields the desired contradiction and concludes the proof. ◻

We are now able to establish the refined lower bounds for $\#\text{INDSUB}(\Phi)$.

Theorem 5.5. *Let Φ denote a computable graph property that is monotone and non-trivial. Suppose that $\mathcal{K}(\Phi)$ is infinite. Then $\#\text{INDSUB}(\Phi)$ cannot be solved in time*

$$g(k) \cdot |V(G)|^{o(k/\sqrt{\log k})}$$

for any function g, unless ETH fails. The same is true for $\#\text{INDSUB}(\overline{\Phi})$ and $\#\text{INDSUB}(\neg\Phi)$.

Proof. We begin similarly as in the proof of Theorem 4.4: As Φ is non-trivial, there is a graph F such that $\Phi(H)$ is false for every H that contains F as a (not necessarily induced) subgraph. Set $r = |V(F)|$ and fix $k \in \mathcal{K}(\Phi)$. By Turáns Theorem (Theorem 2.2) we have that every graph H on k vertices with more than $\left(1 - \frac{1}{r}\right) \cdot \frac{k^2}{2}$ edges contains the clique K_{r+1} and thus F as a subgraph. Consequently, Φ is false on every graph with k vertices and more than $\left(1 - \frac{1}{r}\right) \cdot \frac{k^2}{2}$ edges. We now use Theorem 3.1 and obtain a computable and unique function a of finite support such that

$$\#\mathsf{IndSub}(\Phi, k \to *) = \sum_{H \in \mathcal{G}} a(H) \cdot \#\mathsf{Hom}(H \to *),$$

satisfying that there is a graph H_k on k vertices and at least

$$\binom{k}{2} - \mathsf{hw}(f^{\Phi,k}) + 1 \geq \binom{k}{2} - \left(1 - \frac{1}{r}\right) \cdot \frac{k^2}{2} + 1 \geq \frac{k^2}{2r} - \frac{k}{2}$$

---

[12] If k is not large enough for the inequalities to hold, then k and thus $\hat{k}$ are bounded by a constant, and we can compute the number of $\hat{k}$-cliques in G by brute-force.

[13] Full version available at `https://arxiv.org/abs/1902.04960`.

edges such that $a(H_k) \neq 0$. Consequently, Complexity Monotonicity (Theorem 2.7) yields a tight reduction from the problem #$\mathrm{HOM}(\mathcal{H})$ where $\mathcal{H} := \{H_k \mid k \in \mathcal{K}(\Phi)\}$. By the previous observation, the graph H_k has k vertices and at least $k^2/2r - k/2$ many edges, and by Lemma 5.2 the set of k such that $H_k \in \mathcal{H}$ is dense. Thus we can use Lemma 5.3, which concludes the proof—note that the results for #$\mathrm{INDSUB}(\overline{\Phi})$ and #$\mathrm{INDSUB}(\neg\Phi)$ follow by Fact 2.3. ⌟

We continue with the refined lower bound for properties that only depend on the number of edges of a graph; in this case, we have to assume density.

Theorem 5.6. *Let Φ denote a computable graph property that only depends on the number of edges of a graph. If the set of k for which Φ_k is non-trivial is dense, then #$\mathrm{INDSUB}(\Phi)$ cannot be solved in time*

$$g(k) \cdot |V(G)|^{o(k/\sqrt{\log k})}$$

for any function g, unless ETH fails.

Proof. We use the same set-up as in the proof of Theorem 4.3. In particular, we obtain $\hat{\Phi}$ such that #$\mathrm{INDSUB}(\Phi)$ and #$\mathrm{INDSUB}(\hat{\Phi})$ are equivalent, $\mathcal{K}(\hat{\Phi})$ is dense, and $\mathsf{hw}(f^{\hat{\Phi},k}) \leq \frac{1}{2}\binom{k}{2}$ for all $k \in \mathcal{H}(\hat{\Phi})$. We use Theorem 3.1 and obtain a computable and unique function a of finite support such that

$$\#\mathsf{IndSub}(\Phi, k \to *) = \sum_{H \in \mathcal{G}} a(H) \cdot \#\mathsf{Hom}(H \to *),$$

satisfying that there is a graph H_k on k vertices and at least

$$\binom{k}{2} - \mathsf{hw}(f^{\Phi,k}) + 1 \geq \frac{k^2}{4} - \frac{k}{2}$$

edges such that $a(H_k) \neq 0$. The application of Complexity Monotonicity (2.7) and Lemma 5.3 is now similar to the previous proof; the only difference is, that we can choose $r = 2$. ⌟

As a final result in this section, we establish a tight conditional lower bound for sparse properties; recall that a property Φ is called *sparse* if there is a constant s only depending on Φ such that Φ is false on every graph with k vertices and more than sk edges. Instead of relying in the Kostochka-Thomason-Theorem, it suffices to use Turán's Theorem, however.

Theorem 5.7. *Let Φ denote a computable sparse graph property such that $\mathcal{K}(\Phi)$ is dense. Then #$\mathrm{INDSUB}(\Phi)$ cannot be solved in time*

$$g(k) \cdot |V(G)|^{o(k)}$$

for any function g, unless ETH fails. The same is true for #$\mathrm{INDSUB}(\overline{\Phi})$ and #$\mathrm{INDSUB}(\neg\Phi)$.

Proof. Let s denote the constant given by the definition of sparsity, and fix $k \in \mathcal{K}(\Phi)$. We use Theorem 3.1 and obtain a computable and unique function a of finite support such that

$$\#\mathsf{IndSub}(\Phi, k \to *) = \sum_{H \in \mathcal{G}} a(H) \cdot \#\mathsf{Hom}(H \to *),$$

satisfying that there is a graph H_k on k vertices and at least $\binom{k}{2} - \mathsf{hw}(f^{\Phi,k}) + 1$ edges such that $a(H_k) \neq 0$. Now choose $r := \lceil \frac{k}{2(s+1)+1} \rceil$, and observe that

$$\binom{k}{2} - \mathsf{hw}(f^{\Phi,k}) + 1 > \binom{k}{2} - sk > \binom{k}{2} - (s+1)k \geq \left(1 - \frac{1}{r}\right) \cdot \frac{1}{2}|V(H_k)|^2.$$

By Turáns Theorem (Theorem 2.2), H_k hence contains K_{r+1} as a subgraph and, in particular, K_r as a minor. Furthermore, Complexity Monotonicity (Theorem 2.7) shows that we can, given a graph G compute $\#\mathsf{Hom}(H_k \to G)$ in linear time if we are given oracle access to $\#\mathsf{IndSub}(\Phi, k \to \star)$. As we have seen in the proof of Lemma 5.3, it is known that, whenever a graph F is a minor of a graph H, there is a tight reduction from counting homomorphisms from F to counting homomorphisms from H [24, 55]. In particular, this implies that we can compute $\#\mathsf{Hom}(K_r \to G)$ in linear time if we are given oracle access to $\#\mathsf{IndSub}(\Phi, k \to \star)$. Note further that $\#\mathsf{Hom}(K_r \to G)$ is at least 1 if and only if G contains a clique of size r.

We continue similarly as in the proof of Lemma 5.3: Assume that there is an algorithm $\mathbb{A}$ that solves $\#\mathrm{INDSUB}(\Phi)$ in time $g(k) \cdot |V(G)|^{o(k)}$ for some function g. We show that $\mathbb{A}$ can be used to solve the problem of *finding* a clique of size $\hat{k}$ in a graph G in time $\hat{f}(\hat{k}) \cdot |V(G)|^{o(\hat{k})}$ for some function $\hat{f}$. Given $\hat{k}$ and G, search for the smallest $k \in \mathcal{K}(\Phi)$ such that

$$\hat{k} \leq \lceil \frac{k}{2(s+1)+1} \rceil \,,$$

and note that finding such a k is computable in time only depending on $\hat{k}$ as Φ is computable. Note further that $k \in O(\hat{k})$ as s is a constant and $\mathcal{K}(\Phi)$ is dense.

We construct the graph G' from G by adding $\lceil \frac{k}{2(s+1)+1} \rceil - \hat{k}$ "fresh" vertices and adding edges between every pair of new vertices and every pair containing one new and one old vertex. It is easy to see that G has a clique of size $\hat{k}$ if and only if G' has a clique of size $\lceil \frac{k}{2(s+1)+1} \rceil$. By the analysis and assumptions above, we can decide whether the latter is true by using $\mathbb{A}$ in time $g(k) \cdot |V(G')|^{o(k)}$. As $|V(G')| \in O(|V(G)|)$ and $k \in O(\hat{k})$, the overall time to decide whether G has a clique of size $\hat{k}$ is hence bounded by $\hat{f}(\hat{k}) \cdot |V(G)|^{o(\hat{k})}$ for some function $\hat{f}$, which is impossible, unless ETH fails [13, 14]. The results for $\#\mathrm{INDSUB}(\overline{\Phi})$ and $\#\mathrm{INDSUB}(\neg\Phi)$ follow by Fact 2.3. ◾

6   Hereditary Graph Properties

While we established hardness for a variety of graph properties with f-vectors of small hamming weight in the previous sections, we observe that our meta-theorem (Theorem 4.1) does not apply for properties Φ for which $f^{\Phi,k}$, $f^{\neg\Phi,k}$ and $f^{\overline{\Phi},k}$ have large hamming weight. A well-studied class of properties containing examples of such Φ is the family of hereditary graph properties: In contrast to monotone properties, which are closed under taking subgraphs, a property Φ is called *hereditary* if it is closed under taking *induced* subgraphs. It is a well-known fact that every hereditary property Φ is characterized by a (possibly infinite) set $\Gamma(\Phi)$ of forbidden induced subgraphs, that is

$$\Phi(G) = 1 \Leftrightarrow \forall H \in \Gamma(\Phi) : \#\mathsf{IndSub}(H \to G) = 0 \,.$$

Given any graph H, the property Φ of being *H-free*, that is, not containing H as an induced subgraph, is hereditary with $\Gamma(\Phi) = \{H\}$.

In this section, we settle the hardness-question for many hereditary graph properties as well. Note that Φ is hereditary if and only if its inverse $\overline{\Phi}$ is. In particular, $H \in \Gamma(\Phi) \Leftrightarrow \overline{H} \in \Gamma(\overline{\Phi})$.

Theorem 6.1. *Let H denote graph that is not the trivial graph with a single vertex and let Φ denote the property $\Phi(G) = 1 :\Leftrightarrow G$ is H-free. Then $\#\mathrm{INDSUB}(\Phi)$ is $\#\mathsf{W}[1]$-complete and cannot be solved in time*

$$g(k) \cdot |V(G)|^{o(k)}$$

for any function g, unless ETH fails. The same is true for the problem $\#\mathrm{INDSUB}(\neg\Phi)$. ◾
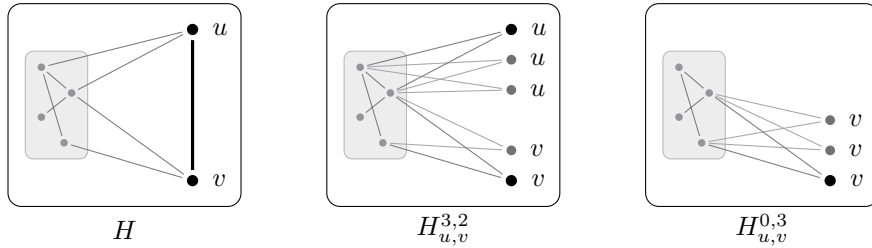
Figure 1 Different explosions of the edge $\{u, v\}$ of a graph H.

We start with some terminology used in this section. Given a graph H, a pair $(u, v) \in V(H)^2$, and two non-negative integers x and y, we construct the *exploded* graph $H(u, v, x, y)$ by adding $x - 1$ clones of u and $y - 1$ clones of v, including all incident edges; if x or y are zero, then we delete u or v, respectively. Given an edge $e = \{u, v\}$ of H and two non-negative integers x and y, we define the *e-exploded* graph as $H_{u,v}^{x,y} := (V(H), E(H) \setminus \{u, v\})(u, v, x, y)$. Consult Figure 1 for a visualization. An edge $e = \{u, v\}$ of a graph H is called *critical*, if $\#\mathsf{IndSub}(H \to H_{u,v}^{x,y}) = 0$ for every pair $x, y \in \mathbb{N}_{\geq 0}$.

Now let Φ denote a hereditary graph property and let $\Gamma(\Phi)$ denote the associated set of forbidden induced subgraphs. We say that Φ has a *critical edge* if there is a graph $H \in \Gamma(\Phi)$ and an edge $\{u, v\} \in E(H)$ such that for all positive integers x and y, the graph $H_{u,v}^{x,y}$ satisfies Φ, that is, for every $\hat{H} \in \Gamma(\Phi)$, we have

$$\#\mathsf{IndSub}(\hat{H} \to H_{u,v}^{x,y}) = 0.$$

Finally, we say that a hereditary property Φ is *critical* if either Φ or its inverse $\overline{\Phi}$ has a critical edge. We will see later in this section that every critical property Φ will induce hardness of $\#\textsc{IndSub}(\Phi)$.

First, we establish that for every graph H with at least two vertices, the property "H-free" is critical. To this end, we rely on neighbour-sharing vertices: Given two vertices u and v of a graph H, we say that u and v are *false twins* if they have the same set of adjacent vertices. Note that, in particular, false twins cannot be adjacent as we consider graphs without self-loops. Furthermore, for a graph H, we define the partition $P(H)$ by adding two vertices to the same block if and only if they are false twins. Finally, we define the graph $H\!\downarrow$ by identifying all vertices in H with the block of $P(H)$ they belong to, that is, the vertices of $H\!\downarrow$ are the blocks of $P(H)$ and two blocks B and B' are adjacent if there are vertices $v \in B$ and $v' \in B'$ such that $\{v, v'\} \in E(H)$.

▸ **Lemma 6.2.** *Let H denote a graph with at least 2 vertices and let Φ denote a hereditary graph property such that $\Gamma(\Phi) = \{H\}$. Then Φ is critical.*

Proof. We show that either Φ or $\overline{\Phi}$ has a critical edge. As $\Gamma(\Phi) = \{H\}$, we need to prove that at least one of H and $\overline{H}$ has a critical edge. Let us start with the following claim.

▷ Claim 6.3. Let F denote a graph with an edge $\{u, v\}$ such that $\{u\}$ and $\{v\}$ are singleton sets in the partition $P(F)$. Then $\{u, v\}$ is a critical edge of F.

Proof. Suppose there are integers $x, y \in \mathbb{N}_{\geq 0}$ such that there is an induced subgraph F' of $F_{u,v}^{x,y}$ that is isomorphic to F. Then there is a natural bijection between the blocks of F and the blocks of $F_{u,v}^{x,y}$ sending the class of each vertex not equal to u, v to themselves, sending the class $\{u\}$ to the class of its x clones and similar for v. But note that the graph $F_{u,v}^{x,y}\!\downarrow$ has one fewer edge than $F\!\downarrow$ (since the previous edge $\{u, v\}$ was removed). However, the induced subgraph F' of $F_{u,v}^{x,y}$ satisfies that $F'\!\downarrow$ is a subgraph of $F_{u,v}^{x,y}\!\downarrow$ and thus also has at least one fewer edges than $F\!\downarrow$, a contradiction to F' being isomorphic to F. ◁

Using the previous claim, it suffices to show that there are two vertices u and v such that $\{u, v\}$ is an edge and $\{u\}$ and $\{v\}$ are singletons in either one of H and $\overline{H}$.

We first show that for every vertex $z \in V(H) = V(\overline{H})$, the set $\{z\}$ is either a singleton in $P(H)$ or in $P(\overline{H})$. To this end, assume that z has a false twin z' in H and a false twin z'' in $\overline{H}$. Consequently, $\{z, z'\} \notin E(H)$ and $\{z, z''\} \notin E(\overline{H})$, and thus $\{z, z'\} \in E(\overline{H})$ and $\{z, z''\} \in E(H)$. Now, as z and z' are false twins in H and $\{z, z''\} \in E(H)$, we see that $\{z', z''\} \in E(H)$. However, as z and z'' are false twins in $\overline{H}$ and $\{z, z'\} \in E(\overline{H})$, we see that $\{z', z''\} \in E(\overline{H})$ as well, which leads to the desired contradiction.

Now assume without loss of generality that H has at least one edge; otherwise we consider $\overline{H}$. If there are false twins u and v in H, then, by the previous argument, $\{u\}$ and $\{v\}$ are singletons in $P(\overline{H})$ and u and v are adjacent in $\overline{H}$. By Claim 6.3, we obtain a critical edge of $\overline{H}$. If there are no false twins in H, we can choose an arbitrary edge of H which is then again critical by Claim 6.3. This concludes the proof.[14] ⌙

We have shown that every hereditary property defined by a single (non-trivial) forbidden induced subgraph is critical. Let us now provide some examples of critical hereditary properties that are defined by multiple forbidden induced subgraphs:

1. $\Phi(H) = 1 :\Leftrightarrow H$ is perfect. A graph H is *perfect* if for every induced subgraph of H, the size of the largest clique equals the chromatic number. By the Strong Perfect Graph Theorem [16], we have that $\Gamma(\Phi)$ is the set of all odd cycles of length at least 5 and their complements. Now observe that every edge of the cycle of length 5 is critical for Φ as the exploded graph is bipartite and thus perfect.
2. $\Phi(H) = 1 :\Leftrightarrow H$ is chordal. A graph is *chordal* if it does not contain an induced cycle of length 4 or more. Consequently, we can choose an arbitrary edge of the cycle of length 4 as a critical edge for Φ, as the resulting exploded graphs do not contain any cycle.
3. $\Phi(H) = 1 :\Leftrightarrow H$ is a split graph. A *split graph* is a graph whose vertices can be partitioned into a clique and an independent set. It is known [31] that $\Gamma(\Phi)$ contains the cycles of length 4 and 5, and the complement of the cycle of length 4. The latter is the graph containing two disjoint edges, and it is easy to see that any of those two edges is critical for Φ.

We now establish hardness of $\#\textsc{IndSub}(\Phi)$ for critical hereditary properties. We start with the following lemma, which constructs a (tight) parameterized Turing-reduction from counting cliques of size k in bipartite graphs.

▢ **Lemma 6.4.** *Let Φ denote a computable and critical hereditary graph property. There is an algorithm $\mathbb{A}$ with oracle access to $\#\textsc{IndSub}(\Phi)$ that expects as input a bipartite graph G and a positive integer k, and computes the number of independent sets of size k in G in time $O(|G|)$. Furthermore, the number of calls to the oracle is bounded by $O(1)$ and every queried pair $(\hat{G}, \hat{k})$ satisfies $|V(\hat{G})| \in O(|V(G)|)$ and $\hat{k} \in O(k)$.*

Proof. Assume without loss of generality that Φ has a critical edge; otherwise we use Fact 2.3 and proceed with $\overline{\Phi}$. Hence choose $H \in \Gamma(\Phi)$ and $e = \{u, v\} \in E(H)$ such that $\#\mathsf{IndSub}(\hat{H} \to H_{u,v}^{x,y}) = 0$ for every $\hat{H} \in \Gamma(\Phi)$ and for all non-negative integers x, y.

Now let $G = (U \dot\cup V, E)$ and k denote the given input. If U or V are empty, then we can trivially compute the number of independent sets of size k. Hence, we have $U = \{u_1, \ldots, u_{n_1}\}$ and $V = \{v_1, \ldots, v_{n_2}\}$ for some integers $n_1, n_2 > 0$. Now, we proceed as follows: In the first step, we construct the graph $H_{u,v}^{n_1,n_2}$. In the next step, we identify u and its $n_1 - 1$ clones with the vertices of U and v and its $n_2 - 1$ clones with

---

[14] For your amusement: if you color all vertices red that are singletons in H and color all vertices blue that are singletons only in $\overline{H}$, then the fact that the Ramsey number $R(2)$ is 3 shows that for H having at least 3 vertices, we find a critical edge in H or $\overline{H}$.
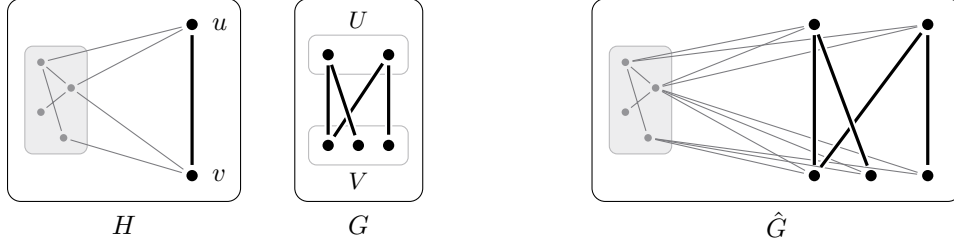
Figure 2 The construction of $\hat{G}$ from Lemma 6.4.

the vertices of V. Finally, we add the edges E of G. We call the resulting graph $\hat{G}$ and we observe that $\hat{G}$ can clearly be constructed in time $O(|G|)$; note that $|H|$ is a constant as Φ is fixed. Consult Figure 2 for a visualization of the construction.

Note that the construction induces a partition of the vertices of $\hat{G}$ into three sets:

$$V(\hat{G}) = R \,\dot{\cup}\, U \,\dot{\cup}\, V,$$

where $R = V(H) \setminus \{u, v\}$; we set $r := |R|$. Now define

$$\mathsf{IndSub}(\Phi, k + r \to \hat{G})[R] := \{F \in \mathsf{IndSub}(\Phi, k + r \to \hat{G}) \mid R \subseteq V(F)\},$$

that is, $\mathsf{IndSub}(\Phi, k + r \to \hat{G})[R]$ is the set of all induced subgraphs F of size $k + r$ in $\hat{G}$ that satisfy Φ and that contain all vertices in R. Next, we show that the cardinality of $\mathsf{IndSub}(\Phi, k + r \to \hat{G})[R]$ reveals the number of independent sets of size k in G.

☞ **Claim 6.5.** Let IS_k denote the set of independent sets of size k in G. We have

$$\#\mathsf{IndSub}(\Phi, k + r \to \hat{G})[R] = \#\mathsf{IS}_k.$$

Proof. Let b denote the function that maps a graph $F \in \mathsf{IndSub}(\Phi, k + r \to \hat{G})[R]$ to a k-vertex subset of G given by

$$b(F) := V(F) \cap (U \,\dot{\cup}\, V).$$

We show that $\mathsf{im}(b) = \mathsf{IS}_k$.

"$\subseteq$": Fix a graph $F \in \mathsf{IndSub}(\Phi, k + r \to \hat{G})[R]$ and write $b(F) = U' \,\dot{\cup}\, V'$ where $U' \subseteq U$ and $V' \subseteq V$. As $|V(F)| = k + r$, as well as $V(F) = R \,\dot{\cup}\, U' \,\dot{\cup}\, V'$, and $|R| = r$, we see that $|b(F)| = k$. Now assume that $b(F)$ is not an independent set, that is, there is an edge $(u, v) \in U' \times V'$ in F. Observing that the induced subgraph $F[R \cup \{u, v\}]$ of F is isomorphic to H, and that H is a forbidden induced subgraph of the property Φ, yields the desired contradiction.

"$\supseteq$": Let $U' \,\dot{\cup}\, V'$ denote an independent set of size k of G with $|U'| = k_1$ and $|V'| = k_2$; note that k_1 or k_2 might be zero. Let F denote the induced subgraph of $\hat{G}$ with vertices $U' \,\dot{\cup}\, V' \,\dot{\cup}\, R$. Then F has $k + r$ vertices and is isomorphic to $H_{u,v}^{k_1,k_2}$. Suppose F does not satisfy Φ. Then F has an induced subgraph isomorphic to a graph in $\Gamma(\Phi)$. However, this is impossible as $\mathsf{IndSub}(\hat{H} \to H_{u,v}^{x,y}) = \emptyset$ for all $\hat{H} \in \Gamma(\Phi)$ and non-negative integers x, y.

This shows that b is a surjective function from $\mathsf{IndSub}(\Phi, k + r \to \hat{G})[R]$ to IS_k. Furthermore, injectivity is immediate by the definition of b as $V(F) \setminus (U \,\dot{\cup}\, V) = R$ for every $F \in \mathsf{IndSub}(\Phi, k + r \to \hat{G})[R]$, which proves the claim. ⌟

It hence remains to show how our algorithm $\mathbb{A}$ can compute the cardinality of $\mathsf{IndSub}(\Phi, k + r \to \hat{G})[R]$.

Claim 6.6. Write $R = \{z_1, \ldots, z_r\}$. We see that

$$\#\mathsf{IndSub}(\Phi, k + r \to \hat{G})[R] = \sum_{J \subseteq [r]} (-1)^{|J|} \cdot \#\mathsf{IndSub}(\Phi, k + r \to \hat{G} \setminus J),$$

where $\hat{G} \setminus J$ is the graph obtained from $\hat{G}$ by deleting all vertices z_i with $i \in J$.

Proof. Using the principle of inclusion and exclusion, we obtain that

$$\begin{aligned}
\#&\mathsf{IndSub}(\Phi, k + r \to \hat{G})[R] \\
&= \#\mathsf{IndSub}(\Phi, k + r \to \hat{G}) - \#\{F \in \mathsf{IndSub}(\Phi, k + r \to \hat{G}) \mid \exists i \in [r] : z_i \notin V(F)\} \\
&= \#\mathsf{IndSub}(\Phi, k + r \to \hat{G}) - \left| \bigcup_{i=1}^{r} \{F \in \mathsf{IndSub}(\Phi, k + r \to \hat{G}) \mid z_i \notin V(F)\} \right| \\
&= \#\mathsf{IndSub}(\Phi, k + r \to \hat{G}) - \sum_{\emptyset \neq J \subseteq [r]} (-1)^{|J|+1} \left| \bigcap_{i \in J} \{F \in \mathsf{IndSub}(\Phi, k + r \to \hat{G}) \mid z_i \notin V(F)\} \right| \\
&= \#\mathsf{IndSub}(\Phi, k + r \to \hat{G}) - \sum_{\emptyset \neq J \subseteq [r]} (-1)^{|J|+1} \#\{F \in \mathsf{IndSub}(\Phi, k + r \to \hat{G}) \mid \forall i \in J : z_i \notin V(F)\} \\
&= \#\mathsf{IndSub}(\Phi, k + r \to \hat{G}) - \sum_{\emptyset \neq J \subseteq [r]} (-1)^{|J|+1} \#\mathsf{IndSub}(\Phi, k + r \to \hat{G} \setminus J) \\
&= \sum_{J \subseteq [r]} (-1)^{|J|} \cdot \#\mathsf{IndSub}(\Phi, k + r \to \hat{G} \setminus J).
\end{aligned}$$

Consequently, the algorithm $\mathbb{A}$ requires linear time in $|G|$ and $2^r \in O(1)$ oracle calls, each query of the form $(\hat{G} \setminus J, k + r)$, to compute the number of independent sets of size k in G as shown in the previous claims. In particular, $|V(\hat{G} \setminus J)| \in O(|V(G)|)$ for each $J \subseteq [r]$ and $k + r \in O(k)$; this completes the proof.

Theorem 6.7. *Let Φ denote a computable and critical hereditary graph property. Then $\#\mathrm{INDSUB}(\Phi)$ is $\#\mathsf{W}[1]$-complete and cannot be solved in time*

$$g(k) \cdot |V(G)|^{o(k)}$$

for any function g, unless ETH fails. The same is true for the problem $\#\mathrm{INDSUB}(\neg\Phi)$.

Proof. It is known that counting independent sets of size k in bipartite graphs is $\#\mathsf{W}[1]$-hard [19] and cannot be solved in time $g(k) \cdot |V(G)|^{o(k)}$ for any function g, unless ETH fails [26]. The theorem thus follows by Lemma 6.4.

As a particular consequence we establish a complete classification for the properties of being H-free, including for instance claw-free graphs and co-graphs [60].

Theorem 6.1. *Let H denote graph that is not the trivial graph with a single vertex and let Φ denote the property $\Phi(G) = 1 :\Leftrightarrow G$ is H-free. Then $\#\text{INDSUB}(\Phi)$ is $\#\text{W}[1]$-complete and cannot be solved in time*

$$g(k) \cdot |V(G)|^{o(k)}$$

for any function g, unless ETH fails. The same is true for the problem $\#\text{INDSUB}(\neg\Phi)$.

Proof. Holds by Lemma 6.2 and Theorem 6.7. ⬛

Finally, note that the previous result is in sharp contrast to the result of Khot and Raman [43] concerning the decision version of $\#\text{INDSUB}(\Phi)$: Their result implies that *finding* an induced subgraph of size k in a graph G that satisfies a hereditary property Φ with $\Gamma(\Phi) = \{H\}$ for some graph H can be done in time $f(k) \cdot |V(G)|^{O(1)}$ for some function f if H is neither a clique nor an independent set of size at least 2.

────── **References** ──────

1   N. Alon, P. Dao, I. Hajirasouliha, F. Hormozdiari, and S. C. Sahinalp. Biomolecular network motif counting and discovery by color coding. *Bioinformatics*, 24(13):i241–i249, 07 2008. doi:10.1093/bioinformatics/btn163.

2   M. Backens. A Complete Dichotomy for Complex-Valued Holant^c. In *45th International Colloquium on Automata, Languages, and Programming, ICALP 2018, July 9-13, 2018, Prague, Czech Republic*, pages 12:1–12:14, 2018. doi:10.4230/LIPIcs.ICALP.2018.12.

3   M. Backens, A. Bulatov, L. A. Goldberg, C. McQuillan, and S. Zivny. Boolean approximate counting CSPs with weak conservativity, and implications for ferromagnetic two-spin. *J. Comput. Syst. Sci.*, 109:95–125, 2020. doi:10.1016/j.jcss.2019.12.003.

4   L. Billera and A. Björner. Face Numbers of Polytopes and Complexes. In *Handbook of discrete and computational geometry*, pages 291–310. CRC Press, Inc., 1997.

5   J. Cai, Z. Fu, H. Guo, and T. Williams. A Holant Dichotomy: Is the FKT Algorithm Universal? In *IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, USA, 17-20 October, 2015*, pages 1259–1276, 2015. doi:10.1109/FOCS.2015.81.

6   J. Cai, S. Huang, and P. Lu. From Holant to #CSP and Back: Dichotomy for Holant^c Problems. *Algorithmica*, 64(3):511–533, 2012. doi:10.1007/s00453-012-9626-6.

7   J. Cai and P. Lu. Holographic algorithms: From art to science. *J. Comput. Syst. Sci.*, 77(1):41–61, 2011. doi:10.1016/j.jcss.2010.06.005.

8   J. Cai, P. Lu, and M. Xia. Holographic Algorithms with Matchgates Capture Precisely Tractable Planar #CSP. *SIAM J. Comput.*, 46(3):853–889, 2017. doi:10.1137/16M1073984.

9   J. Cai, P. Lu, and M. Xia. Dichotomy for Real Holant^c Problems. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018*, pages 1802–1821, 2018. doi:10.1137/1.9781611975031.118.

10 L. S. Chandran and C. R. Subramanian. Girth and treewidth. *J. Comb. Theory, Ser. B*, 93(1):23–32, 2005. doi:10.1016/j.jctb.2004.05.004.

11 H. Chen and S. Mengel. A Trichotomy in the Complexity of Counting Answers to Conjunctive Queries. In *18th International Conference on Database Theory, ICDT 2015, March 23-27, 2015, Brussels, Belgium*, pages 110–126, 2015. doi:10.4230/LIPIcs.ICDT.2015.110.

12 H. Chen and S. Mengel. Counting Answers to Existential Positive Queries: A Complexity Classification. In *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2016, San Francisco, CA, USA, June 26 - July 01, 2016*, pages 315–326, 2016. doi:10.1145/2902251.2902279.

13 J. Chen, B. Chor, M. Fellows, X. Huang, D. W. Juedes, I. A. Kanj, and G. Xia. Tight lower bounds for certain parameterized NP-hard problems. *Inf. Comput.*, 201(2):216–231, 2005. doi:10.1016/j.ic.2005.05.001.

14 J. Chen, X. Huang, I. A. Kanj, and G. Xia. Strong computational lower bounds via parameterized complexity. *J. Comput. Syst. Sci.*, 72(8):1346–1367, 2006. doi:10.1016/j.jcss.2006.04.007.

15 Z. Chen, A. Galanis, L. A. Goldberg, W. Perkins, J. Stewart, and E. Vigoda. Fast Algorithms at Low Temperatures via Markov Chains. In D. Achlioptas and L. A. Végh, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2019, September 20-22, 2019, Massachusetts Institute of Technology, Cambridge, MA, USA*, volume 145 of *LIPIcs*, pages 41:1–41:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.APPROX-RANDOM.2019.41.

16 M. Chudnovsky, N. Robertson, P. Seymour, and R. Thomas. The Strong Perfect Graph Theorem. *Annals of Mathematics*, 164(1):51–229, 2006. URL http://www.jstor.org/stable/20159988.

17 S. A. Cook. The Complexity of Theorem-Proving Procedures. In *Proceedings of the 3rd Annual ACM Symposium on Theory of Computing, May 3-5, 1971, Shaker Heights, Ohio, USA*, pages 151–158, 1971. doi:10.1145/800157.805047.

18 R. Curticapean. *The simple, little and slow things count: On parameterized counting complexity*. PhD thesis, Saarland University, 2015. URL http://scidok.sulb.uni-saarland.de/volltexte/2015/6217/.

19 R. Curticapean, H. Dell, F. V. Fomin, L. A. Goldberg, and J. Lapinskas. A Fixed-Parameter Perspective on #BIS. *Algorithmica*, 81(10):3844–3864, 2019. doi:10.1007/s00453-019-00606-4.

20 R. Curticapean, H. Dell, and D. Marx. Homomorphisms are a good basis for counting small subgraphs. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 210–223, 2017. doi:10.1145/3055399.3055502.

21 R. Curticapean and D. Marx. Complexity of Counting Subgraphs: Only the Boundedness of the Vertex-Cover Number Counts. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 130–139, 2014. doi:10.1109/FOCS.2014.22.

22 M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms.* Springer, 2015. doi:10.1007/978-3-319-21275-3.

23 V. Dalmau and P. Jonsson. The complexity of counting homomorphisms seen from the other side. *Theor. Comput. Sci.*, 329(1-3):315–323, 2004. doi:10.1016/j.tcs.2004.08.008.

24 H. Dell, M. Roth, and P. Wellnitz. Counting Answers to Existential Questions. In *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece*, pages 113:1–113:15, 2019. doi:10.4230/LIPIcs.ICALP.2019.113.

25 J. Díaz, M. J. Serna, and D. M. Thilikos. Counting H-colorings of partial k-trees. *Theor. Comput. Sci.*, 281(1-2):291–309, 2002. doi:10.1016/S0304-3975(02)00017-8.

26 J. Dörfler, M. Roth, J. Schmitt, and P. Wellnitz. Counting Induced Subgraphs: An Algebraic Approach to #W[1]-hardness. In P. Rossmanith, P. Heggernes, and J. Katoen, editors, *44th International Symposium on Mathematical Foundations of Computer Science, MFCS 2019, August 26-30, 2019, Aachen, Germany*, volume 138 of *LIPIcs*, pages 26:1–26:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.MFCS.2019.26.

27 A. Durand and S. Mengel. Structural Tractability of Counting of Solutions to Conjunctive Queries. *Theory Comput. Syst.*, 57(4):1202–1249, 2015. doi:10.1007/s00224-014-9543-y.

28 J. Edmonds. Paths, Trees, and Flowers. *Canadian Journal of Mathematics*, 17:449–467, 1965. doi:10.4153/CJM-1965-045-4.

29 P. Erdös and G. Szckeres. A combinatorial problem in geometry. In I. Gessel and G.-C. Rota, editors, *Classic Papers in Combinatorics*, pages 49–56, Boston, MA, 1987. Birkhäuser Boston. doi:10.1007/978-0-8176-4842-8_3.

30 J. Flum and M. Grohe. The Parameterized Complexity of Counting Problems. *SIAM J. Comput.*, 33(4):892–922, 2004. doi:10.1137/S0097539703427203.

31 S. Földes and P. Hammer. Split graphs. In *Proceedings of the Eighth Southeastern Conference on Combinatorics, Graph Theory, and Computing, Floridat Atlantic University, Boca Raton, February 28-March 3, 1977*, pages 311–315, 1977. URL https://books.google.co.uk/books?id=PNAtPwAACAAJ.

32 L. A. Goldberg and M. Jerrum. Approximating Pairwise Correlations in the Ising Model. *TOCT*, 11(4):23:1–23:20, 2019. doi:10.1145/3337785.

33 J. A. Grochow and M. Kellis. Network Motif Discovery Using Subgraph Enumeration and Symmetry-Breaking. In T. Speed and H. Huang, editors, *Research in Computational Molecular Biology*, pages 92–106, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

34 M. Grohe, T. Schwentick, and L. Segoufin. When is the evaluation of conjunctive queries tractable? In *Proceedings on 33rd Annual ACM Symposium on Theory of Computing, July 6-8, 2001, Heraklion, Crete, Greece*, pages 657–666, 2001. doi:10.1145/380752.380867.

35 S. Huang and P. Lu. A Dichotomy for Real Weighted Holant Problems. *Computational Complexity*, 25(1):255–304, 2016. doi:10.1007/s00037-015-0118-3.

36 R. Impagliazzo and R. Paturi. On the Complexity of k-SAT. *J. Comput. Syst. Sci.*, 62(2):367–375, 2001. doi:10.1006/jcss.2000.1727.

37 M. Jerrum and K. Meeks. The parameterised complexity of counting connected subgraphs and graph motifs. *J. Comput. Syst. Sci.*, 81(4):702–716, 2015. doi:10.1016/j.jcss.2014.11.015.

38 M. Jerrum and K. Meeks. Some Hard Families of Parameterized Counting Problems. *TOCT*, 7(3):11:1–11:18, 2015. doi:10.1145/2786017.

39 M. Jerrum and K. Meeks. The parameterised complexity of counting even and odd induced subgraphs. *Combinatorica*, 37(5):965–990, 2017. doi:10.1007/s00493-016-3338-5.

40   J. Kahn, M. E. Saks, and D. Sturtevant. A topological approach to evasiveness. *Combinatorica*, 4(4):297–306, 1984. doi:10.1007/BF02579140.

41   P. W. Kasteleyn. The statistics of dimers on a lattice: I. The number of dimer arrangements on a quadratic lattice. *Physica*, 27(12):1209–1225, 1961. doi:10.1016/0031-8914(61)90063-5.

42   P. W. Kasteleyn. Dimer Statistics and Phase Transitions. *Journal of Mathematical Physics*, 4(2):287–293, 1963. doi:10.1063/1.1703953.

43   S. Khot and V. Raman. Parameterized complexity of finding subgraphs with hereditary properties. *Theor. Comput. Sci.*, 289(2):997–1008, 2002. doi:10.1016/S0304-3975(01)00414-5.

44   A. V. Kostochka. Lower bound of the Hadwiger number of graphs by their average degree. *Combinatorica*, 4(4):307–316, 1984. doi:10.1007/BF02579141.

45   D. Lokshtanov, D. Marx, and S. Saurabh. Lower bounds based on the Exponential Time Hypothesis. *Bulletin of the EATCS*, 105:41–72, 2011. URL http://eatcs.org/beatcs/index.php/beatcs/article/view/92.

46   L. Lovász. *Large Networks and Graph Limits*, volume 60 of *Colloquium Publications*. American Mathematical Society, 2012. URL http://www.ams.org/bookstore-getitem/item=COLL-60.

47   D. Marx. Can You Beat Treewidth? *Theory of Computing*, 6(1):85–112, 2010. doi:10.4086/toc.2010.v006a005.

48   K. Meeks. The challenges of unbounded treewidth in parameterised subgraph counting problems. *Discrete Applied Mathematics*, 198:170–194, 2016. doi:10.1016/j.dam.2015.06.019.

49   C. A. Miller. Evasiveness of graph properties and topological fixed-point theorems. *Foundations and Trends in Theoretical Computer Science*, 7(4):337–415, 2013. doi:10.1561/0400000055.

50   R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network Motifs: Simple Building Blocks of Complex Networks. *Science*, 298(5594):824–827, 2002. doi:10.1126/science.298.5594.824.

51   G. Pólya. Bemerkung zur Interpolation und zur Näherungstheorie der Balkenbiegung. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 11(6):445–449, 1931.

52   J. S. Provan and M. O. Ball. The Complexity of Counting Cuts and of Computing the Probability that a Graph is Connected. *SIAM J. Comput.*, 12(4):777–788, 1983. doi:10.1137/0212053.

53   F. P. Ramsey. On a Problem of Formal Logic. *Proceedings of the London Mathematical Society*, s2-30(1):264–286, 1930. doi:10.1112/plms/s2-30.1.264.

54   A. L. Rosenberg. On the Time Required to Recognize Properties of Graphs: A Problem. *SIGACT News*, 5(4):15–16, Oct. 1973. doi:10.1145/1008299.1008302.

55   M. Roth. *Counting Problems on Quantum Graphs: Parameterized and Exact Complexity Classifications*. PhD thesis, Saarland University, 2019. URL https://scidok.sulb.uni-saarland.de/bitstream/20.500.11880/27575/1/thesis.pdf.

56   M. Roth and J. Schmitt. Counting induced subgraphs: A Topological Approach to #W[1]-hardness. In *13th International Symposium on Parameterized and Exact Computation, IPEC 2018, August 20-24, 2018, Helsinki, Finland*, pages 24:1–24:14, 2018. doi:10.4230/LIPIcs.IPEC.2018.24.

57   B. Schiller, S. Jager, K. Hamacher, and T. Strufe. StreaM - A Stream-Based Algorithm for Counting Motifs in Dynamic Graphs. In A.-H. Dediu, F. Hernández-Quiroz, C. Martín-Vide, and D. A. Rosenblueth, editors, *Algorithms for Computational Biology*, pages 53–67, Cham, 2015. Springer International Publishing.

58   I. J. Schoenberg. On Hermite-Birkhoff Interpolation. *Journal of Mathematical Analysis and Applications*, 16(3):538 – 543, 1966. doi:10.1016/0022-247X(66)90160-0.

59   F. Schreiber and H. Schwöbbermeyer. Frequency Concepts and Pattern Detection for the Analysis of Motifs in Networks. In C. Priami, E. Merelli, P. Gonzalez, and A. Omicini, editors, *Transactions on Computational Systems Biology III*, pages 89–104, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

60   D. Seinsche. On a property of the class of n-colorable graphs. *Journal of Combinatorial Theory, Series B*, 16(2):191 – 193, 1974. doi:10.1016/0095-8956(74)90063-X.

61   J. Spencer. Ramsey's theorem—a new lower bound. *Journal of Combinatorial Theory, Series A*, 18(1):108 – 115, 1975. doi:10.1016/0097-3165(75)90071-0.

62   H. N. V. Temperley and M. E. Fisher. Dimer problem in statistical mechanics-an exact result. *The Philosophical Magazine: A Journal of Theoretical Experimental and Applied Physics*, 6(68):1061–1063, 1961. doi:10.1080/14786436108243366.

63 A. Thomason. The Extremal Function for Complete Minors. *J. Comb. Theory, Ser. B*, 81(2):318–338, 2001. doi:10.1006/jctb.2000.2013.

64 S. Toda. PP is as Hard as the Polynomial-Time Hierarchy. *SIAM J. Comput.*, 20(5):865–877, 1991. doi:10.1137/0220053.

65 J. R. Ullmann. An algorithm for subgraph isomorphism. *J. ACM*, 23(1):31–42, 1976. doi:10.1145/321921.321925.

66 L. G. Valiant. The Complexity of Computing the Permanent. *Theor. Comput. Sci.*, 8:189–201, 1979. doi:10.1016/0304-3975(79)90044-6.

67 L. G. Valiant. The Complexity of Enumeration and Reliability Problems. *SIAM J. Comput.*, 8(3):410–421, 1979. doi:10.1137/0208032.

68 L. G. Valiant. Holographic Algorithms. *SIAM J. Comput.*, 37(5):1565–1594, 2008. doi:10.1137/070682575.