



Counting Induced Subgraphs: An Algebraic Approach to $\#W[1]$ -Hardness

Julian Dörfler¹ · Marc Roth^{2,3} · Johannes Schmitt^{4,5} · Philip Wellnitz⁶

Received: 13 October 2020 / Accepted: 9 November 2021
© The Author(s) 2021

Abstract

We study the problem $\#INDSUB(\Phi)$ of counting all induced subgraphs of size k in a graph G that satisfy the property Φ . It is shown that, given *any* graph property Φ that distinguishes independent sets from bicliques, $\#INDSUB(\Phi)$ is hard for the class $\#W[1]$, i.e., the parameterized counting equivalent of NP. Under additional suitable density conditions on Φ , satisfied e.g. by non-trivial monotone properties on bipartite graphs, we strengthen $\#W[1]$ -hardness by establishing that $\#INDSUB(\Phi)$ cannot be solved in time $f(k) \cdot n^{o(k)}$ for any computable function f , unless the Exponential Time Hypothesis fails. Finally, we observe that our results remain true even if the input graph G is restricted to be bipartite and counting is done modulo a fixed prime.

Keywords Counting complexity · Edge-transitive graphs · Graph homomorphisms · Induced subgraphs · Parameterized complexity

The third author has received funding from the European Research Council (ERC) under the EU's Horizon 2020 research and innovation programme (No 786580).

✉ Marc Roth
marc.roth@cs.ox.ac.uk
Julian Dörfler
jdoerfler@cs.uni-saarland.de
Johannes Schmitt
schmitt@math.uni-bonn.de
Philip Wellnitz
wellnitz@mpi-inf.mpg.de

- ¹ Graduate School of Computer Science, Saarland Informatics Campus, Saarbrücken, Germany
- ² Department of Computer Science and Merton College, University of Oxford, Oxford, UK
- ³ Cluster of Excellence (MMCI), Saarland Informatics Campus, Saarbrücken, Germany
- ⁴ ETH Zurich, Zurich, Switzerland
- ⁵ Mathematical Institute, University of Bonn, Bonn, Germany
- ⁶ Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany

1 Introduction

The study of the computational complexity of counting problems was initiated by Valiant's seminal work about the complexity of computing the permanent [24]. In contrast to a decision problem which requires to *verify* the existence of a solution, a counting problem asks to compute the *number* of solutions. Counting complexity theory is particularly interesting for problems whose decision versions are solvable efficiently but whose counting versions are intractable. One such example is the problem of finding/counting perfect matchings, whose decision version is solvable in polynomial time [10] and whose counting version is at least as hard as every problem in the Polynomial Hierarchy PH with respect to polynomial-time Turing reductions [23,24]. In this work, we consider the following problem which was first introduced by Jerrum and Meeks [13]: Fix a graph property Φ , given a graph G and a positive integer k , compute the number of all induced subgraphs of G with k vertices that satisfy Φ . We denote this problem by $\#INDSUB(\Phi)$ and remark that, strictly speaking, $\#INDSUB(\Phi)$ is the *unlabeled* version of the problem $p\text{-}\#INDUCEDSUBGRAPHWITHPROPERTY(\Phi)$ as defined in [15, Section 1.3.1]. In particular, our properties only depend on the isomorphism type of a graph and not on any labeling of the vertices.

We study the *parameterized complexity* of $\#INDSUB(\Phi)$ depending on the property Φ . The underlying framework, known as *parameterized counting complexity theory*, was introduced independently by Flum and Grohe [11] and McCartin [18], and constitutes a hybrid of (classical) computational counting and parameterized complexity theory. Here, the method of parameterization allows us to perform a multivariate analysis of the complexity of $\#INDSUB(\Phi)$: Instead of the distinction between polynomial-time solvable and NP-hard cases, we search for properties Φ for which the problem is solvable in time $f(k) \cdot n^{O(1)}$, where n is the number of vertices of the graph and f can be any computable function. If this is the case, the problem is called *fixed-parameter tractable*. Unfortunately, the only known cases of Φ for which $\#INDSUB(\Phi)$ is fixed-parameter tractable are trivial in the sense that there are only finitely many k such that Φ is neither true nor false on the set of all graphs with k vertices. On the contrary, it is easy to see that $\#INDSUB(\Phi)$ is most likely not fixed-parameter tractable if Φ encodes a problem whose decision version is already known to be hard. An example of the latter is the property of being a complete graph. In this case, the problem $\#INDSUB(\Phi)$ is identical to the problem of counting cliques of size k , for which even the decision version, that is, *finding* a clique of size k in a graph with n vertices, cannot be done in time $f(k) \cdot n^{o(k)}$, unless the Exponential Time Hypothesis fails [5,6].

The first non-trivial hardness result of $\#INDSUB(\Phi)$ was given by Jerrum and Meeks for Φ the property of being connected [13]. Note that, in this case, the decision version of the problem can be solved efficiently as, on input G and k , one only has to decide whether there exists a connected component of G of size at least k . This result initiated a line of research in which Jerrum and Meeks proved fixed-parameter tractability of $\#INDSUB(\Phi)$ to be unlikely for the property of having an even (or odd) number of edges [15], for properties that induce low edge densities [14] and for properties that are closed under the addition of edges and whose (edge-)minimal elements have large treewidth [19]. More precisely, all of those results established hardness for the parameterized

complexity class $\#W[1]$, which can be seen as the parameterized counting equivalent of NP. In a recent breakthrough result [8], Curticapean, Dell and Marx have shown, that for every graph property Φ , the problem $\#INDSUB(\Phi)$ is either fixed-parameter tractable or hard for $\#W[1]$, that is, there are no cases of intermediate difficulty. On the downside, they did not provide an explicit criterion for $\#W[1]$ -hardness that allows to pin down the complexity of $\#INDSUB(\Phi)$, given a concrete property Φ .

However, combining the framework of [8] with tools from the “topological approach to evasiveness” by Kahn, Saks and Sturtevant [16], two of the authors of the current paper established $\#W[1]$ -hardness for a wide range of properties, including, for example, all non-trivial properties that are closed under the removal of edges and false on odd cycles [22]. Taken together, the above results suggest the following conjecture.

Conjecture 1 Let Φ be a computable graph property satisfying that there are infinitely many positive integers k such that Φ is neither true nor false on all graphs with k vertices. Then $\#INDSUB(\Phi)$ is $\#W[1]$ -hard.

Unfortunately, a proof of this conjecture seems to be a long way off. In this work however, building up on [8,22], we introduce an algebraic approach that allows us to resolve the above conjecture in case of *all* non-trivial monotone properties on bipartite graphs. In particular, we obtain a matching lower bound under the Exponential Time Hypothesis.

Results and techniques We call a graph property *monotone* if it closed under the removal of vertices and edges and *edge-monotone* if it is closed under the removal of edges only. Furthermore, we write IS_k for the graph consisting of k isolated vertices and $K_{t,t}$ for the complete bipartite graph with t vertices on each side. Our main theorems read as follows.

Theorem 1 Let Φ be a computable graph property and let \mathcal{K} be the set of all prime powers t such that $\Phi(IS_{2t}) \neq \Phi(K_{t,t})$. If \mathcal{K} is infinite then $\#INDSUB(\Phi)$ is $\#W[1]$ hard. If additionally \mathcal{K} is dense then it cannot be solved in time $f(k) \cdot n^{o(k)}$ for any computable function f unless ETH fails. This holds true even if the input graphs to $\#INDSUB(\Phi)$ are restricted to be bipartite.

In the previous theorem, a set \mathcal{K} is *dense* if there exists a constant c such that for every $m \in \mathbb{N}$, there exists a $k \in \mathcal{K}$ such that $m \leq k \leq cm$. While the hypotheses of Theorem 1 sound technical, the theorem applies in many situations. In particular, it is applicable to properties that are neither (edge-) monotone nor the complement thereof: Let Φ be the property of being Eulerian. The graph $K_{t,t}$ contains an Eulerian cycle if $t = 2^s$ for $s \geq 1$. Hence we can apply Theorem 1 with $\mathcal{K} = \{2^s \mid s \geq 1\}$, which is infinite and dense.

Corollary 1 Let Φ be the property of being Eulerian. Then $\#INDSUB(\Phi)$ is $\#W[1]$ -hard and cannot be solved in time $f(k) \cdot n^{o(k)}$ for any computable function f unless the ETH fails. This holds true even if the input graphs to $\#INDSUB(\Phi)$ are restricted to be bipartite.

In case Φ is monotone, that is, closed under the removal of vertices and edges, a short argument implies that the set \mathcal{K} from Theorem 1 is infinite if and only if Φ

and $\neg\Phi$ hold on infinitely many bipartite graphs. Thus we can state the criterion for hardness more concisely as follows.

Theorem 2 *Let Φ be a computable monotone graph property such that Φ and $\neg\Phi$ hold on infinitely many bipartite graphs. Then $\#\text{INDSUB}(\Phi)$ is $\#W[1]$ -hard and cannot be solved in time $f(k) \cdot n^{o(k)}$ for any computable function f unless the Exponential Time Hypothesis fails. This holds true even if the input graphs to $\#\text{INDSUB}(\Phi)$ are restricted to be bipartite.*

Let us illustrate further consequences of the previous theorems with respect to (edge-) monotone properties. First of all, most of the prior hardness results [13–15, 19, 22] are shown to hold in the restricted case of bipartite graphs. We provide three examples:

Corollary 2 *The problem $\#\text{INDSUB}(\Phi)$, restricted to bipartite input graphs, is $\#W[1]$ -hard and cannot be solved in time $f(k) \cdot |V(G)|^{o(k)}$ for any computable function f unless ETH fails, if Φ is one of the properties of being disconnected, planar or non-hamiltonian.*

One example of a monotone property Φ for which the complexity of $\#\text{INDSUB}(\Phi)$ was unknown, even for general graphs, is given by the following corollary of Theorem 2.

Corollary 3 *Let F be a fixed bipartite graph with at least one edge and define $\Phi(G) = 1$ if G does not contain a subgraph isomorphic to F . Then $\#\text{INDSUB}(\Phi)$ is $\#W[1]$ -hard and cannot be solved in time $f(k) \cdot |V(G)|^{o(k)}$ for any computable function f unless ETH fails. This holds true even if the input graphs of $\#\text{INDSUB}(\Phi)$ are restricted to be bipartite.*

As the number of induced subgraphs of size k that satisfy Φ equals $\binom{|V(G)|}{k}$ minus the number of induced subgraphs of size k that satisfy $\neg\Phi$, all of the previous result remain true for the complementary properties $\neg\Phi$.

In proving the previous theorems we build up on the approach in [8, 22], where it was shown that, given a graph property Φ and a positive integer k , the number of induced subgraphs of size k in a graph G that satisfy Φ can equivalently be expressed as the following sum over all (isomorphism types of) graphs H :

$$\sum_H a_\Phi(H) \cdot \#\text{Hom}(H \rightarrow G), \quad (1)$$

where a_Φ is a function from graphs to integers with finite support and, furthermore, $\#\text{Hom}(H \rightarrow G)$ is the number of graph homomorphisms from H to G . It is known that computing a linear combination of homomorphism numbers, as in the above expression, is *precisely as hard as* computing its hardest term with a non-zero coefficient ([8], also implicitly proved in [4]). We refer to this property as *complexity monotonicity*. In [22] two of the authors of the current paper used a topological approach to analyze the coefficient $a_\Phi(K_k)$ of the complete graph on k vertices. If this coefficient is non-zero then complexity monotonicity implies that computing the number

of induced subgraphs of size k in a graph G that satisfy Φ is at least as hard as computing the number $\#\text{Hom}(K_k \rightarrow G)$. This, in turn, is equivalent to computing the number of cliques of size k in G , the canonical $\#W[1]$ -complete problem [11]. While this approach led to hardness proofs for a wide range of properties Φ , it seems that resolving Conjecture 1, even restricted to monotone properties, requires a significant number of new ideas. Without going too much into the details¹ of [22], our analysis of $a_\Phi(K_k)$ is complicated by the fact that the number of edges of the complete graph on $k \geq 4$ vertices is not a prime power. In this work, we hence focus on the coefficient of $a_\Phi(H)$ for graphs H that have a prime power number of edges and for which computing $\#\text{Hom}(H \rightarrow G)$ is hard. One example of such graphs is the biclique $K_{t,t}$ for some prime power t . Here a biclique $K_{t,t}$, also called a complete bipartite graph, has t vertices on each side and contains every edge from a vertex on the left side to a vertex on the right side. Hence the number of edges is t^2 which is a prime power if t is.

In analyzing the coefficient $a_\Phi(K_{t,t})$ of the complete bipartite graph, we invoke the results of Rivest and Vuillemin [21] who considered transitive boolean functions over a domain of prime power cardinality to resolve the asymptotic version of what is known as *Karp's evasiveness conjecture* (we recommend Miller's survey [20] for an excellent overview).

Given a property Φ and a graph H , the *alternating enumerator* of Φ and H is defined to be

$$\hat{\chi}(\Phi, H) := \sum_{S \subseteq E(H)} \Phi(H[S]) \cdot (-1)^{\#S},$$

where $H[S]$ is the graph with vertices $V(H)$ and edges S . Roughly speaking, it will turn out that the value of $a_\Phi(H)$ is closely related to $\hat{\chi}(\Phi, H)$. We furthermore point out that, in case Φ is closed under the removal of edges, the alternating enumerator $\hat{\chi}(\Phi, H)$ equals what is called the reduced Euler characteristic of the simplicial complex on $E(H)$ associated to Φ [20, 22]. In Sect. 3 we study the alternating enumerator in case of edge-transitive graphs, that is, graphs whose automorphism groups act transitively on the set of edges. We give a self-contained proof of the following fact, which implicitly follows from [21].

Lemma 1 *Let Φ be a graph property and let H be an edge-transitive graph with p^ℓ edges such that p is a prime and $\Phi(H[\emptyset]) \neq \Phi(H)$. Then it holds that $\hat{\chi}(\Phi, H) = (\pm 1) \bmod p$.*

Observe that Lemma 1 provides a strong motivation for the study of edge-transitive graphs with a prime power number of edges: Following the aforementioned strategy, we will be able to show that counting the number of induced subgraphs of size k in a graph G that satisfy Φ is at least as hard as counting homomorphisms from any edge-transitive graph H with k vertices and a prime-power number of edges that satisfies

¹ Readers familiar with [22] might recall that fixed points of group actions have been used to derive a simpler formula to compute the number $a_\Phi(K_t)$ modulo a prime p for positive powers t of p . This formula would simplify greatly if the group had a p -power number of elements and acted transitively on the edges of K_t . Unfortunately, this can never happen for $t \geq 4$, since the number of edges of K_t is not itself a p -power.

$\Phi(H[\emptyset]) \neq \Phi(H)$. Ultimately, we will then obtain intractability of $\#\text{INDSUB}(\Phi)$ by relying on the complexity classification for counting homomorphisms due to Dalmau and Jonsson [9].

In the second part of Sect. 3, we will thus fully classify those graphs as subgraphs of bipartite graphs or vertex-transitive subgraphs of wreath graphs; consult Sect. 3 for the formal definitions. The proof of the following theorem, which might be of independent interest, relies on a non-trivial application of Sylow's theorems.

Theorem 3 *Let G be a connected edge-transitive graph with p^t edges for some prime p and positive integer t . Then either G is bipartite or G is vertex-transitive and can be obtained from the wreath graph W_{p^k} for $k \geq 1$ by removing edges (or both).*

With the analysis of $\hat{\chi}$ and edge-transitive graphs completed, we turn to the reduction from counting homomorphisms in Sect. 4. More precisely, given a class \mathcal{H} of edge-transitive graphs with a prime power number of edges and a graph property Φ such that for every $H \in \mathcal{H}$ we have that $\Phi(H[\emptyset]) \neq \Phi(H)$, we construct a parameterized Turing reduction from $\#\text{HOM}(\mathcal{H})$ to $\#\text{INDSUB}(\Phi)$. Here, the problem $\#\text{HOM}(\mathcal{H})$ is defined as follows: Given as input a graph $H \in \mathcal{H}$ and a graph G , compute the number of homomorphisms from H to G . For technical reasons, we cannot immediately transform the number of induced subgraphs that satisfy Φ to a linear combination of homomorphism numbers as in Eq. (1). We solve this technical issue by introducing color-prescribed variants of those problems in an intermediate step. In this context we consider H -colored graphs. Recall that a graph G is H -colored if it comes with a homomorphism c from G to H . A homomorphism from H to G is then called color-prescribed if it maps every vertex v of H to a vertex u of G satisfying that $c(u) = v$. We demonstrate that, given an H -colored graph G and oracle access to $\#\text{INDSUB}(\Phi)$, the following linear combination can be computed in time $f(|V(H)|) \cdot |V(G)|^{O(1)}$.

$$\sum_{S \subseteq E(H)} \hat{a}_\Phi(S) \cdot \#\text{cp-Hom}(H[S] \rightarrow_H G). \quad (2)$$

Here $\text{cp-Hom}(H[S] \rightarrow_H G)$ denotes the set of color-prescribed homomorphisms from $H[S]$ to G and \hat{a}_Φ is a function of finite support only depending in Φ . In particular, $\hat{a}_\Phi(E(H))$ and $\hat{\chi}(\Phi, H)$ are proved to agree up to a factor of -1 . Finally, we establish complexity monotonicity for linear combinations of color-prescribed homomorphisms as in Eq. (2), which in combination with Lemma 1 yields the desired reduction.

Combining the previous results, we invoke the reduction on graph properties that are non-trivial on bipartite graphs and prove Theorems 1 and 2, in Sect. 5. Furthermore, we illustrate in the Sect. 6 that our algebraic approach readily extends to modular counting by proving that both, Theorems 1 and 2 remain true in case counting is done modulo a fixed prime.

2 Preliminaries

Given a positive integer k , we write $[k]$ for the set $\{1, \dots, k\}$ and given a set A we write $\binom{A}{k}$ for the set of all subsets of size k of A . Furthermore, assuming that A is

finite, we write $\#A$ or $|A|$ for its cardinality. Given a function $g : A \times B \rightarrow C$ and an element $a \in A$, we write $g(a, \star)$ for the function which maps $b \in B$ to $g(a, b)$.

2.1 Graph Theory

Graphs in this work are considered simple, undirected and without self-loops. More precisely, a graph G is a pair of a finite set $V(G)$ of vertices and a symmetric and irreflexive relation $E(G) \subseteq V(G)^2$. If a graph H is obtained from G by deleting a set of edges and a set of vertices of G , including incident edges, then H is called a *subgraph* of G . Given a subset \hat{V} of $V(G)$ we write $G[\hat{V}]$ for the graph with vertices \hat{V} and edges $E \cap \hat{V}^2$. The resulting graph is called an *induced subgraph* of G . An *edge-subgraph* of a graph H is a graph obtained from H by deleting edges. Given a set $S \subseteq E(H)$ we write $H[S]$ for the edge-subgraph $(V(H), S)$ of H .

Homomorphisms and embeddings A *homomorphism* from a graph H to a graph G is a mapping $h : V(H) \rightarrow V(G)$ that preserves adjacencies. In other words, for every edge $\{u, v\} \in E(H)$ it holds that $\{h(u), h(v)\} \in E(G)$. We write $\text{Hom}(H \rightarrow G)$ for the set of all homomorphisms from H to G . A homomorphism inducing a bijection of vertices and satisfying $\{u, v\} \in E(H)$ if and only if $\{f(u), f(v)\} \in E(G)$ is called an *isomorphism* and we say that two graphs H and \hat{H} are *isomorphic* if there exists an isomorphism from H to \hat{H} . We write $\text{Sub}(H \rightarrow G)$ and $\text{IndSub}(H \rightarrow G)$ for the sets of all subgraphs and induced subgraphs of G , respectively, that are isomorphic to H .

An isomorphism from a graph to itself is called an *automorphism*. The set of automorphisms of a graph, together with the operation of functional composition constitutes a group, called the *automorphism group* of a graph. Slightly abusing notation, we will write $\text{Aut}(H)$ for both the set of automorphisms of a graph H as well as for the automorphism group of H .

An *embedding* is an injective homomorphism and we write $\text{Emb}(H \rightarrow G)$ for the set of embeddings from H to G . If an embedding h from H to G additionally satisfies that $\{h(u), h(v)\} \in E(G)$ implies $\{u, v\} \in E(H)$, we call it a *strong embedding*. We write $\text{StrEmb}(H \rightarrow G)$ for the set of strong embeddings from H to G . Observe that the images of embeddings and strong embeddings from H to G are precisely the subgraphs and induced subgraphs of G that are isomorphic to H .

Colored variants Given graphs G and H , we say that G is *H -colored* if G comes with a homomorphism c from G to H , called an *H -coloring*. Note that, in particular, every edge-subgraph of H can be H -colored by the identity function on $V(H)$, which is assumed to be the given coloring whenever we consider H -colored edge-subgraphs of H in this paper. Given an edge-subgraph F of H and a homomorphism h from F to a H -colored graph G , we say that h is *color-prescribed* if for all $v \in V(F) = V(H)$ it holds that $c(h(v)) = v$. We write $\text{cp-Hom}(F \rightarrow_H G)$ for the set of all color-prescribed homomorphisms from F to G . $\text{cp-StrEmb}(F \rightarrow_H G)$ is defined similarly for color-prescribed strong embeddings. We point out that a definition of cp-Emb is obsolete as every color-prescribed homomorphism is injective by definition and hence

an embedding. Furthermore, we write $\text{cp-Sub}(F \rightarrow_H G)$ and $\text{cp-IndSub}(F \rightarrow_H G)$ for the sets of images of color-prescribed embeddings and strong embeddings from F to G , respectively. Elements of $\text{cp-Sub}(F \rightarrow_H G)$ and $\text{cp-IndSub}(F \rightarrow_H G)$ are referred to as color-prescribed subgraphs and induced subgraphs.²

Graph properties and the Alternating Enumerator A *graph property* is defined to be a function Φ from graphs to $\{0, 1\}$ such that for any pair of isomorphic graphs H and \hat{H} we have that $\Phi(H) = \Phi(\hat{H})$. Adapting the notation of Rivest and Vuillemin [21], we define the *alternating enumerator* of a property Φ and a graph H to be the function

$$\hat{\chi}(\Phi, H) := \sum_{S \subseteq E(H)} \Phi(H[S]) \cdot (-1)^{\#S}.$$

A graph property Φ is called *edge-monotone* if it is closed under the removal of edges. It is called *monotone* if it is closed under the removal of edges *as well as* vertices.³ Given a graph property Φ , a positive integer k and a graph G , we write $\text{IndSub}(\Phi, k \rightarrow G)$ for the set of all induced subgraphs of size k of G that satisfy Φ . Furthermore, given a graph property Φ and an H -colored graph G , we write $\text{cp-IndSub}(\Phi \rightarrow_H G)$ for the set of all color-prescribed induced subgraphs of size $|V(H)|$ in G that satisfy Φ . Observe that $\text{cp-IndSub}(\Phi \rightarrow_H G)$ can be expressed as a (disjoint) union over edge-subgraphs of H satisfying Φ :

$$\text{cp-IndSub}(\Phi \rightarrow_H G) = \bigcup_{\substack{S \subseteq E(H) \\ \Phi(H[S])=1}} \text{cp-IndSub}(H[S] \rightarrow_H G).$$

2.2 Parameterized Counting Complexity

The field of parameterized counting was introduced independently by McCartin [18] and Flum and Grohe [11] and constitutes a hybrid of classical computational counting and parameterized complexity theory. A *parameterized counting problem* is a pair of a function $P : \Sigma^* \rightarrow \mathbb{N}$ and a computable parameterization $\kappa : \Sigma^* \rightarrow \mathbb{N}$. It is called *fixed-parameter tractable* (FPT) if there exists a computable function f and a deterministic algorithm that computes $P(x)$ in time $f(\kappa(x)) \cdot |x|^{O(1)}$ for every $x \in \Sigma^*$. A *parameterized Turing reduction* from (P, κ) to $(\hat{P}, \hat{\kappa})$ is a deterministic FPT algorithm with respect to κ that is given oracle access to \hat{P} and that on input x computes $P(x)$ with the additional restriction that there exists a computable function g such that for any oracle query y it holds that $\hat{\kappa}(y) \leq g(\kappa(x))$. We write $(P, \kappa) \leq_T^{\text{fpt}} (\hat{P}, \hat{\kappa})$ if a parameterized Turing reduction exists.

² The reader might notice that the sets $\text{cp-Sub}(F \rightarrow_H G)$ and $\text{cp-Hom}(F \rightarrow_H G)$ as well as $\text{cp-IndSub}(F \rightarrow_H G)$ and $\text{cp-StrEmb}(F \rightarrow_H G)$ are essentially the same as a color-prescribed homomorphism is uniquely identified by its image. However, we decided to distinguish those notions in order to make the combinatorial arguments in Sect. 4 more accessible.

³ To avoid confusion, we remark that in some literature, e.g. in [19] a property is called monotone if it is closed under *addition* of vertices and edges.

Given a graph G and a positive integer k , the parameterized counting problem $\# \text{CLIQUE}$ asks to compute the number of complete subgraphs of size k in G and is parameterized by k , that is $\kappa(G, k) := k$. It is complete for the class $\#W[1]$, which can be seen as a parameterized counting equivalent of NP [11]. Evidence for the fixed-parameter intractability of $\#W[1]$ -hard problems is given by the *Exponential Time Hypothesis* (ETH), which asserts that 3-SAT cannot be solved⁴ in time $\exp(o(m))$ where m is the number of clauses of the input formula. Assuming ETH, $\# \text{CLIQUE}$ cannot be solved in time $f(k) \cdot n^{o(k)}$ for any function f [5,6] and hence $\#W[1]$ -hard problems are not fixed-parameter tractable.

Given a recursively enumerable class of graphs \mathcal{H} , the problem $\# \text{HOM}(\mathcal{H})$ asks, on input a graph $H \in \mathcal{H}$ and an arbitrary graph G , to compute the number $\# \text{Hom}(H \rightarrow G)$. Its parameterization is given by $\kappa(H, G) := |V(H)|$. The problems $\# \text{CP- HOM}(\mathcal{H})$ and $\# \text{CP- INDSub}(\mathcal{H})$ are defined similarly. Further, we define $\# \text{CP- INDSub}(\Phi)$ to be the problem of, given a graph G that is H -colored for some graph H , computing $\# \text{cp-IndSub}(\Phi \rightarrow_H G)$ and parameterize it by $\kappa(G) := |V(H)|$ —note that the H -coloring of G is part of the input and hence κ is well-defined. Finally, the problem $\# \text{INDSub}(\Phi)$ asks, given a graph G and a positive integer k , to compute $\# \text{IndSub}(\Phi, k \rightarrow G)$ and the parameterization is given by $\kappa(G, k) := k$.

3 Alternating Enumerators and p -edge-Transitive Graphs

In this part of the paper we will provide a rough exposition of the work of Rivest and Vuillemin [21] who studied transitive boolean functions to resolve the asymptotic version of Karp's evasiveness conjecture. We will then apply their result to graphs H that are both edge-transitive and have p^ℓ many edges for some prime p . This will enable us to conclude that the alternating enumerator of Φ and H is (± 1) modulo p whenever $\Phi(H[\emptyset]) \neq \Phi(H)$. We start by introducing some required notions from algebraic graph theory.

The automorphism group of a graph H induces a group action on the edges of H , given by $h\{u, v\} := \{h(u), h(v)\}$. A group action is *transitive* if there exists only one orbit and a graph H is called *edge-transitive* if the automorphism group action on the edges is transitive, that is, if for every pair of edges $\{u, v\}$ and $\{\hat{u}, \hat{v}\}$ there exists an automorphism $h \in \text{Aut}(H)$ such that $h\{u, v\} = \{\hat{u}, \hat{v}\}$. If additionally the number of edges of an edge-transitive graph is a prime power p^ℓ we call the graph *p -edge-transitive*.

Lemma 1 *Let Φ be a graph property and let H be an edge-transitive graph with p^ℓ edges such that p is a prime and $\Phi(H[\emptyset]) \neq \Phi(H)$. Then it holds that $\hat{\chi}(\Phi, H) = (\pm 1) \bmod p$.*

Lemma 1 is implicitly proven in [21, Theorem 4.3], but for completeness we will include a short and self-contained proof, demonstrating a first application of the machinery of Sylow subgroups that we will need later.

⁴ We point out that this includes deterministic and randomized algorithms.

For the proofs in this section, let us recall some key results from group theory. Given a prime number p , a finite group Γ' is called a p -group if the order $\#\Gamma'$ is a power of p . The following is a well-known and central result from the theory of finite groups.

Theorem 4 (Sylow theorems) *Let Γ be a finite group of order $\#\Gamma = p^k m$ for a prime p and an integer $m \geq 1$ coprime to p . Then Γ contains a subgroup Γ' of order p^k . Moreover, every other subgroup Γ'' of Γ of order p^k is conjugate to Γ' , that is there exists $g \in \Gamma$ with $\Gamma'' = g\Gamma'g^{-1}$. In particular, the groups Γ' , Γ'' are isomorphic (via the conjugation by g).*

Finally, every subgroup $\tilde{\Gamma} \subseteq \Gamma$ which is a p -group is actually contained in some conjugate $g\Gamma'g^{-1}$ of the group Γ' .

A subgroup $\Gamma' \subseteq \Gamma$ as above is called a p -Sylow subgroup of Γ .

The following result is a first important application of the Sylow theorems. It can be found as Exercise (E28) in [1]; we include a proof for completeness.

Lemma 2 *Let Γ be a finite group acting transitively on a set T such that $\#T = p^\ell$ for some $\ell \geq 0$. Then the induced action of any p -Sylow subgroup $\Gamma' \subseteq \Gamma$ on T is still transitive.*

Proof Let $t_0 \in T$ be any element, then T is the orbit of t_0 under Γ . Let $\text{Stab}_\Gamma(t_0) = \{g \in \Gamma : gt_0 = t_0\}$ be the stabilizer of t_0 under the action of Γ . Then by the Orbit-Stabilizer theorem, we have

$$\#\Gamma = (\#\Gamma t_0) \cdot (\#\text{Stab}_\Gamma(t_0)) = (\#T) \cdot (\#\text{Stab}_\Gamma(t_0)). \quad (3)$$

As in the Sylow theorems, let k, m be the unique nonnegative integers such that $\#\Gamma = p^k m$ with m not divisible by p and let $\Gamma' \subseteq \Gamma$ be a p -Sylow subgroup of Γ , which is of order p^k . The stabilizer of t_0 under the induced action of the subgroup $\Gamma' \subseteq \Gamma$ is given by

$$\text{Stab}_{\Gamma'}(t_0) = \{g \in \Gamma' : gt_0 = t_0\} = \text{Stab}_\Gamma(t_0) \cap \Gamma'.$$

Clearly this is a subgroup of the group Γ' and by Lagrange's theorem, the order of $\text{Stab}_{\Gamma'}(t_0)$ divides the order p^k of Γ' . Thus it is itself a power of p , say $\#\text{Stab}_{\Gamma'}(t_0) = p^n$.

On the other hand, $\text{Stab}_{\Gamma'}(t_0)$ is also a subgroup of $\text{Stab}_\Gamma(t_0)$. Inserting the order of Γ and the size of T in Eq. (3) we obtain

$$p^k m = p^\ell \cdot (\#\text{Stab}_\Gamma(t_0)), \quad (4)$$

and thus $\#\text{Stab}_\Gamma(t_0)$ can at most contain a factor of $p^{k-\ell}$. Again, by Lagrange's theorem, the order p^n of the subgroup $\text{Stab}_{\Gamma'}(t_0)$ divides the order of $\text{Stab}_\Gamma(t_0)$ and thus $n \leq k - \ell$. Finally, by the Orbit-Stabilizer theorem applied to the action of Γ' on t_0 , we have

$$p^k = \#\Gamma' = (\#\Gamma' t_0) \cdot (\#\text{Stab}_{\Gamma'}(t_0)) = (\#\Gamma' t_0) \cdot p^n. \quad (5)$$

Thus, on the one hand we obtain $\# \Gamma' t_0 = p^{k-n} \geq p^{k-(k-\ell)} = p^\ell$. On the other hand we obtain $\Gamma' t_0 \subseteq T$ and thus $\# \Gamma' t_0 \leq \# T = p^\ell$. Hence we have the equality $\# \Gamma' t_0 = p^\ell = \# T$ and thus $\Gamma' t_0 = T$. In other words, the action of Γ' on T is transitive, finishing the proof. \square

This result allows us to give a short proof of Lemma 1 above.

Proof of Lemma 1 Let $\Gamma = \text{Aut}(H)$ be the automorphism group of the graph H , then by the assumption that H is edge-transitive, the action of Γ on the set $E(H)$ of edges of H is transitive. Since $\#E(H) = p^\ell$ for some $\ell \geq 0$, we see that by Lemma 2 any p -Sylow subgroup $\Gamma' \subseteq \Gamma$ still acts transitively on $E(H)$. Now consider the sum

$$\hat{\chi}(\Phi, H) = \sum_{S \subseteq E(H)} \Phi(H[S]) \cdot (-1)^{\#S}.$$

The action of Γ' on $E(H)$ induces an action of Γ' on the set of subsets $\mathcal{P}(E(H)) := \{S \subseteq E(H)\}$ of $E(H)$. Indeed, for $S \subseteq E(H)$ and $g \in \Gamma'$ we define $gS = \{gs : s \in S\}$. For this action, the set $\mathcal{P}(E(H))$ can be written as a disjoint union of the orbits $\Gamma' S_0$ of a set $S \subseteq \mathcal{P}(E(H))$ of representatives S_0 . (Recall that for a group action two orbits are either disjoint or equal.) This allows us to write the sum above as

$$\hat{\chi}(\Phi, H) = \sum_{S_0 \in \mathcal{S}} \sum_{S \in \Gamma' S_0} \Phi(H[S]) \cdot (-1)^{\#S}.$$

Until now we have just reordered the summands above, combining all summands for S in the same Γ' orbit.

Now since all elements $g \in \Gamma' \subseteq \text{Aut}(H)$ act by graph automorphisms on H , we have that the graphs $H[gS_0]$ and $H[S_0]$ are isomorphic, so in particular $\Phi(H[gS_0]) = \Phi(H[S_0])$. Applying this to the formula for $\hat{\chi}(\Phi, H)$ above, we get

$$\hat{\chi}(\Phi, H) = \sum_{S_0 \in \mathcal{S}} (\# \Gamma' S_0) \cdot \Phi(H[S_0]) \cdot (-1)^{\#S_0}. \quad (6)$$

Now by the Orbit-Stabilizer theorem, the size $\# \Gamma' S_0$ of the orbit of S_0 divides the order p^k of Γ' , so $\# \Gamma' S_0$ is itself a power of p . Further, unless $S_0 \subseteq E(H)$ is invariant under Γ' , the size of its orbit $\# \Gamma' S_0$ is a *positive* power of p and thus congruent to 0 mod p . However, the only two sets $S_0 \subseteq E(H)$ invariant under Γ' are $S_0 = \emptyset$ and $S_0 = E(H)$: Indeed, assume that S_0 is invariant under Γ' and nonempty. Then S_0 contains an element e_0 , and since S_0 is Γ' -invariant, S_0 also contains the entire orbit of e_0 under Γ' . But since Γ' acted transitively on $E(H)$, S_0 must equal the whole set $E(H)$.

To summarize, when computing $\hat{\chi}(\Phi, H)$ modulo p all but two summands in the sum in Eq. (6) are congruent to 0. Hence, we can simplify Eq. (6) to

$$\hat{\chi}(\Phi, H) = \Phi(H[\emptyset]) + \Phi(H[E(H)]) \cdot (-1)^{\#E(H)} = \Phi(H[\emptyset]) - \Phi(H) \pmod{p}.$$

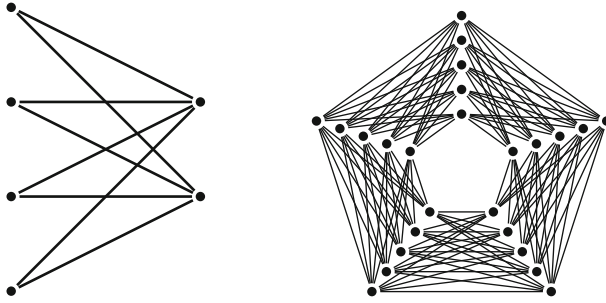


Fig. 1 The two main examples for p -edge-transitive graphs: Complete bipartite graphs K_{p^ℓ, p^m} with $\ell, m \geq 0$ (left) and wreath graphs W_{p^k} for $k \geq 1$ (right)

Note that we use the fact that for $p > 2$ we have that $\#E(H)$ is odd since it is a prime power and for $p = 2$ we have $-1 = 1$ modulo p . Now, the condition $\Phi(H[\emptyset]) \neq \Phi(H)$ exactly gives us $\Phi(H[\emptyset]) - \Phi(H) = \pm 1 \pmod{p}$. \square

There are two main examples for p -edge-transitive graphs. The first one is the class of the complete, bipartite graphs K_{p^ℓ, p^m} with $\ell, m \geq 0$. The graph K_{p^ℓ, p^m} has $p^{\ell+m}$ edges and the automorphism group clearly acts transitively on the edges of that graph. The second example is the class of wreath graphs W_{p^k} for $k \geq 1$. The graph W_{p^k} has p^k vertices that can be decomposed in disjoint sets V_0, \dots, V_{p-1} of order p^{k-1} each, and edges $\{v_i, v_{i+1}\}$ for each $i = 0, \dots, p-1$ and vertices $v_i \in V_i, v_{i+1} \in V_{i+1}$ (where it is understood that $V_p = V_0$). Thus in total, W_{p^k} has p^{2k-1} edges, except for $p = 2$ where it has 2^{2k-2} edges. The graph W_{p^k} can be seen as the lexicographical product of a p -cycle with a graph consisting of p^{k-1} disjoint vertices. For $k = 1$ we exactly obtain the p -cycle. To see that W_{p^k} is edge-transitive, we observe that on the one hand, for fixed i we can apply an arbitrary permutation on V_i leaving the graph invariant. On the other hand, there exists a “rotational action” sending V_j to V_{j+1} for $j = 0, \dots, p-1$, which also leaves the graph invariant. Using these two types of automorphisms, we can map every edge to every other edge. Consider Fig. 1 for a visualization of the two main examples of p -edge-transitive graphs.

A graph G is called *vertex-transitive* if its automorphism group $\text{Aut}(G)$ acts transitively on its set of vertices $V(G)$. The following result tells us that in a certain sense K_{p^ℓ, p^m} and W_{p^k} are the maximal p -edge-transitive graphs: note that a bipartite graph is a subgraph of a graph of the form K_{p^ℓ, p^m} for some $\ell, m \geq 0$.

Theorem 3 *Let G be a connected edge-transitive graph with p^t edges for some prime p and positive integer t . Then either G is bipartite or G is vertex-transitive and can be obtained from the wreath graph W_{p^k} for $k \geq 1$ by removing edges (or both).*

For the proof of Theorem 3, we will use the following well-known result about the relation between edge and vertex-transitivity [2, Proposition 15.1].

Lemma 3 *Let G be a connected graph and let $\Gamma \subseteq \text{Aut}(G)$ be a subgroup acting transitively on the set of edges $E(G)$. Then either Γ acts transitively on the set of vertices $V(G)$ (and thus G is vertex-transitive) or G is bipartite (or both).*

The proof from [2] carries over verbatim to the setting of the previous lemma, by replacing the full group $\text{Aut}(G)$ with the subgroup Γ .

Finally, before beginning the proof of Theorem 3 we recall some facts about the semidirect product of groups, since such products appear below. Given groups Γ_1, Γ_2 and a group homomorphism $\sigma : \Gamma_2 \rightarrow \text{Aut}(\Gamma_1)$, the semidirect product $\Gamma_1 \rtimes \Gamma_2$ is defined as the group on the underlying set $\Gamma_1 \times \Gamma_2$ with group law

$$(g_1, h_1) \cdot (g_2, h_2) = (g_1 \sigma(h_1)(g_2), h_1 h_2) \text{ for } (g_1, h_1), (g_2, h_2) \in \Gamma_1 \times \Gamma_2.$$

The group $\Gamma_1 \rtimes \Gamma_2$ naturally contains Γ_1, Γ_2 as the subgroups $\Gamma_1 \times \{e_{\Gamma_2}\}$ and $\{e_{\Gamma_1}\} \times \Gamma_2$. Moreover, given a set X and actions of Γ_1, Γ_2 on X such that

$$h \cdot (g \cdot x) = \sigma(h)(g) \cdot (h \cdot x) \text{ for } g \in \Gamma_1, h \in \Gamma_2, x \in X$$

there is a well-defined action of $\Gamma_1 \rtimes \Gamma_2$ on X given by $(g, h) \cdot x = g \cdot (h \cdot x)$.

Proof of Theorem 3 Let G be a p -edge-transitive, non-bipartite graph. Then by Lemma 2 any p -Sylow subgroup $\Gamma \subseteq \text{Aut}(G)$ still acts transitively on the edges $E(G)$ of G . By Lemma 3, since G is not bipartite, the group Γ acts transitively on the set of vertices $V(G)$ (and thus G is also vertex-transitive).

We observe that in this case, by the Orbit-Stabilizer theorem, we have $\#V(G) = p^k$ for some $k \geq 1$. We claim that, in this case, G is an edge subgraph of W_{p^k} .

To see this, let us reformulate our situation slightly: We identify the vertex set $V(G)$ with the set $[p^k] = \{1, \dots, p^k\}$. Then we can canonically identify $\text{Aut}(G)$ as a subgroup of S_{p^k} , the symmetric group on $[p^k]$ (this is because a graph automorphism is uniquely determined by its action on the vertices of a graph). Inside $\text{Aut}(G)$ we have the subgroup Γ , which is a p -group. By the Sylow theorem, there exists a p -Sylow subgroup $\Gamma' \subseteq S_{p^k}$ containing Γ .⁵ Since the action of Γ is transitive on the set of edges $E(G)$, we can obtain $E(G)$ by starting with some edge $e_0 = \{v_1, v_2\} \in E(G)$ with $v_1, v_2 \in [p^k]$ and taking its orbit $\{\{gv_1, gv_2\} : g \in \Gamma\} = E(G)$. But note that by instead taking the orbit of e_0 under $\Gamma' \subseteq S_{p^k}$ we get at least this set of edges and maybe more. Denote by G' the graph with vertices $[p^k]$ and edges $\{\{gv_1, gv_2\} : g \in \Gamma'\}$. We claim that $G' \cong W_{p^k}$.

To show this we will explicitly identify the p -Sylow subgroup $\Gamma' \subseteq S_{p^k}$ (recall that by the Sylow theorem it is unique up to conjugation, that is reordering of the elements of $[p^k]$).

First note that S_{p^k} has $(p^k)!$ elements. Inductively one sees that the highest power of p appearing in this number is $p^{e(k)}$ for $e(k) = p^{k-1} + p^{k-2} + \dots + p + 1$. We will inductively construct a subgroup $\Gamma(p, k)$ of S_{p^k} with $p^{e(k)}$ elements, which then is a p -Sylow subgroup. We note that a description of such a p -Sylow subgroup is given in [25].

For $k = 1$ we have $e(k) = 1$ and a p -Sylow subgroup $\Gamma(p, 1) \subseteq S_p$ is generated by a cyclic permutation $1 \mapsto 2, 2 \mapsto 3, \dots, p \mapsto 1$ of the elements of $[p]$. The group $\Gamma(p, 1)$ is isomorphic to the cyclic group $\mathbb{Z}/p\mathbb{Z}$.

⁵ Here we remark that while Γ was chosen to be a p -Sylow subgroup of $\text{Aut}(G)$, it is not necessarily p -Sylow in the bigger group S_{p^k} .

Now assume we constructed $\Gamma(p, k - 1)$ for some $k \geq 2$, then we first note that a product of p copies $\prod_{i=0}^{p-1} \Gamma(p, k - 1)$ of $\Gamma(p, k - 1)$ acts on $[p^k]$ where the i -th factor acts by permutations on the elements

$$ip^{k-1} + 1, ip^{k-1} + 2, \dots, ip^{k-1} + p^{k-1} = (i + 1)p^{k-1}.$$

All of these actions commute, so we can see the product $\prod_{i=0}^{p-1} \Gamma(p, k - 1)$ as a subgroup of S_{p^k} . However, there is a further action of $\mathbb{Z}/p\mathbb{Z}$ on $[p^k]$ sending j to $j + p^{k-1}$ (modulo p^k). This action cyclically permutes the p blocks of p^{k-1} elements in $[p^k]$ on which the p factors of $\prod_{i=0}^{p-1} \Gamma(p, k - 1)$ act. Thus these two actions do not commute, but indeed they induce an action of the semidirect product

$$\Gamma(p, k) = \left(\prod_{i=0}^{p-1} \Gamma(p, k - 1) \right) \rtimes \mathbb{Z}/p\mathbb{Z},$$

where $\mathbb{Z}/p\mathbb{Z}$ maps to the automorphism group of $\prod_{i=0}^{p-1} \Gamma(p, k - 1)$ by permuting the factors of the product. We claim that $\Gamma(p, k)$ is the desired p -Sylow subgroup of S_{p^k} .

Indeed, as the underlying set of the semidirect product is simply the product of the two underlying sets of the groups, the number of elements of the semidirect product is

$$\#\Gamma(p, k) = (\#\Gamma(p, k - 1))^p \cdot p = (p^{e(k-1)})^p \cdot p = p^{pe(k-1)+1} = p^{e(k)},$$

so it has the correct number of elements and is indeed a subgroup of S_{p^k} .

Now recall what we want to show: for a pair $\{v_1, v_2\}$ of vertices forming an edge of our original graph G , we want to show that the graph G' with edges $\{\{gv_1, gv_2\} : g \in \Gamma' \cong \Gamma(p, k)\}$ is isomorphic to the wreath graph W_{p^k} . By relabeling the vertices (that is performing a conjugation in S_{p^k}) we may assume that $\Gamma' = \Gamma(p, k)$. Furthermore, by a translation in the group $\Gamma(p, k)$, which acts transitively on the elements of $[p^k]$, we may assume that $v_1 = 1$. Now if v_2 were in the first block $[p^{k-1}]$ of vertices, on which the first factor $\Gamma(p, k - 1)$ operates, then it is easy to see that the resulting graph G' would not be connected: the first factor $\Gamma(p, k - 1)$ would send the edge $\{1, v_2\}$ only to edges within the first block $[p^{k-1}]$ and then the cyclic permutation by the factor $\mathbb{Z}/p\mathbb{Z}$ would send this pattern of edges to the $p - 1$ other blocks, giving us a disjoint union of p graphs. This is not possible, since our original graph G is a subgraph of G' and also was assumed to be connected.

Thus we may assume that v_2 is in one of the other blocks

$$P(a) = [p^{k-1}] + ip^{k-1} \text{ for } a = 1, \dots, p - 1.$$

Now we want to argue that we can reorder these blocks, sending $P(a)$ to $P(1)$ and leaving $P(0)$ invariant, such that the group action of $\Gamma(p, k)$ is respected. And indeed, let $b \in \mathbb{Z}/p\mathbb{Z}$ be the multiplicative inverse of a (such that $ab = 1 \pmod{p}$), then there

is a permutation of $[p^k]$ sending the block $P(i)$ to the block $P(i \cdot b \bmod p)$ (where the block is just translated as a whole, not permuting the elements inside). And indeed, we see that $P(a)$ is sent to $P(1)$. The reason why this permutation respects the form of the action⁶ of $\Gamma(p, k)$ is that multiplication by b induces a group isomorphism $\mathbb{Z}/p\mathbb{Z} \rightarrow \mathbb{Z}/p\mathbb{Z}$ on the factor $\mathbb{Z}/p\mathbb{Z}$ of $\Gamma(p, k)$.

To summarize, we can assume without loss of generality that we start with an edge $\{1, v_2\}$ with v_2 in the second block of vertices. But then it is easy to see that the graph G' obtained by taking the orbit of $\{1, v_2\}$ under $\Gamma(p, k)$ is indeed the wreath graph W_{p^k} . Indeed, the group $\Gamma(p, k)$ acts transitively within each of the p blocks of vertices (since the i -th factor $\Gamma(p, k-1)$ above acts transitively there), so every edge from the first to the second block is in the orbit of $\{1, v_2\}$. Then finally the cyclic permutation action of $\mathbb{Z}/p\mathbb{Z}$ sends these edges to the set of all edges between blocks i and $i+1$, which exactly gives the set of edges of the wreath graph. This finishes the proof. \square

4 The Main Reduction: From Homomorphisms to Induced Subgraphs

In what follows we will construct a sequence of reductions, starting from $\#\text{HOM}(\mathcal{H})$ and ending in $\#\text{INDSUB}(\Phi)$. Here, \mathcal{H} is a recursively enumerable set of p -edge-transitive graphs and Φ is a graph property such that for every graph $H \in \mathcal{H}$ we have that $\Phi(H[\emptyset]) \neq \Phi(H)$.

More precisely, we will prove that

$$\#\text{HOM}(\mathcal{H}) \stackrel{\text{Lemma 4}}{\leq_{\text{T}}^{\text{fpt}}} \#\text{CP-HOM}(\mathcal{H}) \stackrel{\text{Lemma 9}}{\leq_{\text{T}}^{\text{fpt}}} \#\text{CP-INDSUB}(\Phi) \stackrel{\text{Lemma 10}}{\leq_{\text{T}}^{\text{fpt}}} \#\text{INDSUB}(\Phi) \quad (7)$$

In particular, all of those reductions will be tight in the sense that conditional lower bounds on the fine-grained complexity of $\#\text{HOM}(\mathcal{H})$ immediately transfer to $\#\text{INDSUB}(\Phi)$. For the hardness results we rely on a result of Dalmau and Jonsson [9] stating that the problem $\#\text{HOM}(\mathcal{H})$ is known to be $\#\text{W}[1]$ -hard whenever \mathcal{H} is recursively enumerable and of unbounded treewidth.⁷ Here a class of graphs is said to have unbounded treewidth if for every $b \in \mathbb{N}$ there exists a graph in the class with treewidth at least b .

4.1 Reducing Homomorphisms to Color-Prescribed Homomorphisms

In the first reduction we are given graphs H and G and the goal is to compute $\#\text{Hom}(H \rightarrow G)$ using an oracle for $\#\text{cp-Hom}(H \rightarrow_H \star)$. This can be done by taking precisely $|V(H)|$ copies of the vertices of G , that is, one for each vertex in H and then

⁶ To be precise, what happens is the following: the map sending $P(i)$ to $P(i \cdot b \bmod p)$ is a permutation of $[p^k]$, that is an element $\sigma \in S_{p^k}$. What we are claiming is that the subgroup $\Gamma(p, k) \subseteq S_{p^k}$ is stable under the conjugation by σ , that is $\Gamma(p, k) = \sigma \Gamma(p, k) \sigma^{-1}$. So σ is a relabeling of the vertices of our graph G' which leaves the graph itself invariant.

⁷ We remark that the graph parameter of treewidth is not used explicitly in this work. Hence we omit the definition and refer the interested reader e.g. to Chapter 11 in [12].

adding an edge between two vertices u and v if they are copies of adjacent vertices in G and the vertices of H corresponding to the copies of $V(G)$ that contain u and v are adjacent in H as well; this construction is also known as the Tensor product of H and G . The construction is formalized in the proof of the following lemma. In particular it is shown that the resulting graph \hat{G} is H -colored.

Lemma 4 *Let H be a graph. There exists an algorithm \mathbb{A} that is given a graph G as input and has oracle access to the function $\#\text{cp-Hom}(H \rightarrow_H \star)$ and computes $\#\text{Hom}(H \rightarrow G)$ in time $f(|V(H)|) \cdot |V(G)|$ where f is a computable function. Further, every oracle query \hat{G} satisfies $|V(\hat{G})| \leq f(|V(H)|) \cdot |V(G)|$.*

Proof Let $k = |V(H)|$. It will be convenient to assume that $V(H) = [k]$. Given a graph G , we construct a graph \hat{G} as follows. The vertex set of \hat{G} is defined to be

$$V(\hat{G}) = \bigcup_{i=1}^k V_i,$$

where $V_i = \{v_i \mid v \in V(G)\}$ is a copy of $V(G)$ identified with vertex $i \in V(H)$. We add an edge $\{u_i, v_j\}$ to \hat{G} if and only if $\{i, j\} \in E(H)$ and $\{u, v\} \in E(G)$. Now it can easily be verified that the function $c : V(\hat{G}) \rightarrow V(H)$ given by $c(v_i) := i$ is an H -coloring of \hat{G} . Furthermore it is easy to see that

$$\#\text{cp-Hom}(H \rightarrow_H \hat{G}) = \#\text{Hom}(H \rightarrow G),$$

which concludes the proof. \square

4.2 Reducing Color-prescribed Homomorphisms to Color-prescribed Induced Subgraphs

The reduction from color-prescribed homomorphisms to color-prescribed induced subgraphs requires the introduction of an H -colored variant of the framework of graph motif parameters, which was explicitly introduced in [8] and implicitly used in [4]. More precisely, given an H -colored graph G and a property Φ , we will express $\#\text{cp-IndSub}(\Phi \rightarrow_H G)$ as a linear combination of color-prescribed homomorphisms counts, that is, terms of the form $\#\text{cp-Hom}(H[S] \rightarrow_H G)$. In a first step, we show complexity monotonicity for linear combinations of color-prescribed homomorphisms. While this property allows a quite simple proof, a second step, in which we study the coefficient of $\#\text{cp-Hom}(H \rightarrow_H G)$ requires a thorough understanding of the alternating enumerator of Φ and H . In case of p -edge-transitive graphs, the latter is provided by Lemma 1.

We start by introducing a colored variant of the tensor product of graphs (see e.g. Chapter 5.4.2 in [17]). Given two H -colored graphs G and \hat{G} with colorings c and \hat{c} we define their *color-prescribed* tensor product $G \times_H \hat{G}$ as the graph with vertices $V = \{(v, \hat{v}) \in V(G) \times V(\hat{G}) \mid c(v) = \hat{c}(\hat{v})\}$ and edges between (v, \hat{v}) and (u, \hat{u}) if and only if $\{v, u\} \in E(G)$ and $\{\hat{v}, \hat{u}\} \in E(\hat{G})$. The next lemma states that $\#\text{cp-Hom}$ is linear with respect to \times_H .

Lemma 5 *Let H be a graph, let F be an edge-subgraph of H , and let G and \hat{G} be H -colored. Then we have that*

$$\#cp\text{-}Hom(F \rightarrow_H G \times_H \hat{G}) = \#cp\text{-}Hom(F \rightarrow_H G) \cdot \#cp\text{-}Hom(F \rightarrow_H \hat{G}).$$

Proof It can easily be verified that the function $b(h, \hat{h})(v) := (h(v), \hat{h}(v))$ that assigns elements in $cp\text{-}Hom(F \rightarrow_H G) \times cp\text{-}Hom(F \rightarrow_H \hat{G})$ to elements in $cp\text{-}Hom(F \rightarrow_H G \times_H \hat{G})$ is a well-defined bijection. \square

The proof of the complexity monotonicity property for color-prescribed homomorphisms (Lemma 7) will require to solve a system of linear equations. The following lemma proves that the corresponding matrix is non-singular.

Lemma 6 *Let H be a graph (which we consider H -coloured by the identity mapping) and let M be a square matrix of size $2^{|E(H)|}$ such that the rows and columns are identified by the subsets of edges of H . Furthermore assume that the entries of M are given by*

$$M(S, T) := \#cp\text{-}Hom(H[S] \rightarrow_H H[T]).$$

Then M is non-singular. This holds true even if M is considered as a matrix over \mathbb{Z}_p , that is, the field with p elements. In the latter case, the entries are taken modulo p .

Proof We fix any linear extension \lesssim of the subset inclusion relation on $E(H)$ and order the columns and rows of M accordingly. We claim that M is triangular. To see this we first observe that $M(S, S) = 1$ for every S , given by the identity homomorphism from $H[S]$ to $H[S]$ which is, of course, color-prescribed. Now consider $M(S, T)$ for some $T \neq S$ with $T \lesssim S$. It follows that there exists an edge $\{u, v\}$ in $S \setminus T$ since \lesssim linearly extends subset inclusion. Now assume that there exists a color-prescribed homomorphism h from $H[S]$ to $H[T]$. By color-prescribedness we have that $h(u) = u$ and $h(v) = v$, contradicting the fact that h is a homomorphism and $\{u, v\} \notin T$. Hence $M(S, T) = 0$ and, consequently, M is upper-triangular. \square

We are now prepared to prove the color-prescribed variant of complexity monotonicity.

Lemma 7 (Complexity monotonicity) *Let H be a graph and let a be a computable function from edge-subgraphs of H to rationals. There exists an algorithm \mathbb{A} that is given an H -colored graph G as input and has oracle access to the function*

$$\sum_{S \subseteq E(H)} a(H[S]) \cdot \#cp\text{-}Hom(H[S] \rightarrow_H \star),$$

and computes $\#cp\text{-}Hom(H[S] \rightarrow_H G)$ for all S such that $a(H[S]) \neq 0$ in time $f(|H|) \cdot |V(G)|$ where f is a computable function. Furthermore, every oracle query \hat{G} satisfies $|V(\hat{G})| \leq f(|H|) \cdot |V(G)|$.

Proof Using Lemma 5 we have that for every H -colored graph F it holds that

$$\sum_{S \subseteq E(H)} a(H[S]) \cdot \# \text{cp-Hom}(H[S] \rightarrow_H (G \times_H F)) \quad (8)$$

$$= \sum_{S \subseteq E(H)} a(H[S]) \cdot \# \text{cp-Hom}(H[S] \rightarrow_H G) \cdot \# \text{cp-Hom}(H[S] \rightarrow_H F), \quad (9)$$

which we can evaluate for $F = H[\emptyset], \dots, H[E(H)]$. This induces a system of linear equations; for each of the aforementioned F we obtain an equation

$$b_F = \sum_{S \subseteq E(H)} a_{F,S} \cdot x_S,$$

such that $b_F = \sum_{S \subseteq E(H)} a(H[S]) \cdot \# \text{cp-Hom}(H[S] \rightarrow_H (G \times_H F))$, and $a_{F,S} = \# \text{cp-Hom}(H[S] \rightarrow_H F)$, and $x_S = a(H[S]) \cdot \# \text{cp-Hom}(H[S] \rightarrow_H G)$. Now observe that we can compute b_F using the oracle, and that we can compute $a_{F,S}$ in time only depending in H (since F is an edge-subgraph of H and $S \subseteq E(H)$). Furthermore, for our choices of F , the corresponding matrix is non-singular by Lemma 6. Consequently, the numbers $a(H[S]) \cdot \# \text{cp-Hom}(H[S] \rightarrow_H G)$ are uniquely determined and can be computed by solving the system using Gaussian elimination. Finally, we obtain the numbers $\# \text{cp-Hom}(H[S] \rightarrow_H G)$ by multiplying with $a(H[S])^{-1}$ whenever $a(H[S]) \neq 0$. \square

It remains to express the number of color-prescribed induced subgraphs that satisfy a property Φ as a linear combination of color-prescribed homomorphisms.

Lemma 8 *Let H be a graph, let Φ be a graph property and let G be an H -colored graph. Then it holds that*

$$\# \text{cp-IndSub}(\Phi \rightarrow_H G) = \sum_{S \in E(H)} \Phi(H[S]) \sum_{J \subseteq E(H) \setminus S} (-1)^{\#J} \cdot \# \text{cp-Hom}(H[S \cup J] \rightarrow_H G).$$

Moreover, the absolute values of the coefficient of $\# \text{cp-Hom}(H \rightarrow_H G)$ and $\hat{\chi}(\Phi, H)$ are equal.

Proof We start by establishing the following claim.

Claim 1 *Let H be graph, let $S \subseteq E(H)$ and let G be an H -colored graph. Then*

$$\# \text{cp-IndSub}(H[S] \rightarrow_H G) = \sum_{J \subseteq E(H) \setminus S} (-1)^{\#J} \cdot \# \text{cp-Sub}(H[S \cup J] \rightarrow_H G).$$

Proof of claim It holds that

$$\begin{aligned} & \text{cp-IndSub}(H[S] \rightarrow_H G) \\ &= \text{cp-Sub}(H[S] \rightarrow_H G) \setminus \left(\bigcup_{e \in E(H) \setminus S} \text{cp-Sub}(H[S \cup \{e\}] \rightarrow_H G) \right), \quad (10) \end{aligned}$$

and hence, by inclusion-exclusion, $\#cp\text{-}IndSub(H[S] \rightarrow_H G)$ is equal to

$$\#cp\text{-}Sub(H[S] \rightarrow_H G) - \sum_{\emptyset \subsetneq J \subseteq E(H) \setminus S} (-1)^{\#J-1} \cdot \#cp\text{-}Sub(H[S \cup J] \rightarrow_H G) \quad (11)$$

$$= \sum_{J \subseteq E(H) \setminus S} (-1)^{\#J} \cdot \#cp\text{-}Sub(H[S \cup J] \rightarrow_H G), \quad (12)$$

which concludes the proof of the claim.

Now we have that

$$\#cp\text{-}IndSub(\Phi \rightarrow_H G) \quad (13)$$

$$= \sum_{S \in E(H)} \Phi(H[S]) \cdot \#cp\text{-}IndSub(H[S] \rightarrow_H G) \quad (14)$$

$$= \sum_{S \in E(H)} \Phi(H[S]) \sum_{J \subseteq E(H) \setminus S} (-1)^{\#J} \cdot \#cp\text{-}Sub(H[S \cup J] \rightarrow_H G) \quad (15)$$

$$= \sum_{S \in E(H)} \Phi(H[S]) \sum_{J \subseteq E(H) \setminus S} (-1)^{\#J} \cdot \#cp\text{-}Hom(H[S \cup J] \rightarrow_H G) \quad (16)$$

where (14) follows from the definition of $\#cp\text{-}IndSub(\Phi \rightarrow_H G)$, (15) is Claim 1 and (16) holds as color-prescribed homomorphisms are injective and a color-prescribed embedding is uniquely identified by its image. Collecting for the coefficient of $\#cp\text{-}Hom(H \rightarrow_H G)$ yields

$$\sum_{S \in E(H)} \Phi(H[S]) \cdot (-1)^{\#E(H) - \#S} = (-1)^{\#E(H)} \cdot \hat{\chi}(\Phi, H). \quad (17)$$

□

The application of the complexity monotonicity property for color-prescribed homomorphisms (Lemma 7) requires non-zero coefficients. However, this can be guaranteed for the coefficient of interest in case of p -edge-transitive graphs as shown in Sect. 3. Formally, the reduction is constructed as follows.

Lemma 9 *Let Φ be a computable graph property and let H be a p -edge-transitive graph such that $\Phi(H[\emptyset]) \neq \Phi(H)$. There exists an algorithm \mathbb{A} that is given an H -colored graph G as input and has oracle access to the function*

$$\#cp\text{-}IndSub(\Phi \rightarrow_H \star),$$

and computes $\#cp\text{-}Hom(H \rightarrow_H G)$ in time $f(|H|) \cdot |V(G)|$ where f is a computable function. Furthermore, every oracle query \hat{G} is H -colored as well and satisfies $|V(\hat{G})| \leq f(|H|) \cdot |V(G)|$.

Proof Using Lemma 8 we can express $\#cp\text{-}IndSub(\Phi \rightarrow_H \star)$ as a linear combination of color-prescribed homomorphisms. In particular, the coefficient of $\#cp\text{-}Hom(H \rightarrow_H \star)$ is $(\pm 1) \cdot \hat{\chi}(\Phi, H)$ and by Lemma 1 we have that this number is non-zero whenever H is p -edge-transitive and $\Phi(H[\emptyset]) \neq \Phi(H)$. Hence we can use the algorithm from Lemma 7 to compute $\#cp\text{-}Hom(H \rightarrow_H G)$ in the desired running time. \square

4.3 Reducing Color-Prescribed Induced Subgraphs to Uncolored Induced Subgraphs

The last part of the reduction sequence allows us to get rid of the colors. More precisely, we will reduce the problem of counting color-prescribed induced subgraphs of an H -colored graph to the problem of counting uncolored induced subgraphs of size $|V(H)|$ in a graph, both with respect to some property Φ . The proof is a straightforward application of the inclusion-exclusion principle, which is standard for reducing from the colored to the uncolored version of a parameterized counting problem (see for instance [7, Section 1.4.1]). We include a proof only for completeness.

Lemma 10 *Let Φ be a graph property and let H be a graph with k vertices. There exists an algorithm \mathbb{A} that is given an H -colored graph G as input and has oracle access to the function $\#IndSub(\Phi, k \rightarrow \star)$ and computes $\#cp\text{-}IndSub(\Phi \rightarrow_H G)$ in time $f(k) \cdot |V(G)|$ where f is a computable function. Furthermore, every oracle query \hat{G} satisfies $|V(\hat{G})| \leq |V(G)|$ and, in particular, \hat{G} allows an H -coloring as well.*

Proof It will be convenient to assume that $V(H) = [k]$. We first check whether the H -coloring c of G is surjective. If this is not the case then there exists some vertex $i \in V(H)$ such that $i \notin \text{im}(c)$ and hence there is no color-prescribed induced subgraph of G , so \mathbb{A} can just output 0. Otherwise, the H -coloring of G induces a partition of $V(G)$ in k many non-empty and pairwise disjoint subsets, each associated with some “color” $i \in V(H)$. This allows us to equivalently express $cp\text{-}IndSub(\Phi \rightarrow_H G)$ in terms of vertex-colorful induced subgraphs:

$$cp\text{-}IndSub(\Phi \rightarrow_H G) = \left\{ S \subseteq \binom{V(G)}{k} \mid c(S) = [k] \wedge \Phi(G[S]) = 1 \right\} \quad (18)$$

By the principle of inclusion and exclusion we obtain that

$$\#cp\text{-}IndSub(\Phi \rightarrow_H G) = \sum_{J \subseteq [k]} (-1)^{\#J} \cdot \#IndSub(\Phi, k \rightarrow G_J), \quad (19)$$

where G_J is the graph obtained from G by deleting all vertices that are colored with some color in J . Hence we can compute $\#cp\text{-}IndSub(\Phi \rightarrow_H G)$ using 2^k oracle calls. Finally, we observe that H -colored graphs are closed under the removal of vertices and therefore every oracle query G_J allows an H -coloring. \square

5 Non-trivial Monotone Properties on Bipartite Graphs

In what follows, we apply the algebraic approach which was laid out in the preceding sections to bipartite graph properties. This will allow us to prove our main result. To this end, we say that a set $\mathcal{K} \subseteq \mathbb{N}$ is *dense* if there exists a constant c such that for every $k' \in \mathbb{N}$ there exists $k \in \mathcal{K}$ such that $k' \leq k \leq ck'$. Furthermore, we write IS_k for the graph with k isolated vertices. The following theorem is obtained by invoking the reduction sequence (7) to complete bipartite graphs $K_{t,t}$ for prime powers $t = p^k$, which are p -edge-transitive (see Sect. 3). The extension to modular counting is presented in the next section.

Theorem 1 *Let Φ be a computable graph property and let \mathcal{K} be the set of all prime powers t such that $\Phi(IS_{2t}) \neq \Phi(K_{t,t})$. If \mathcal{K} is infinite then $\#INDSUB(\Phi)$ is $\#W[1]$ hard. If additionally \mathcal{K} is dense then it cannot be solved in time $f(k) \cdot n^{o(k)}$ for any computable function f unless ETH fails. This holds true even if the input graphs to $\#INDSUB(\Phi)$ are restricted to be bipartite.*

While $\#W[1]$ -hardness will follow by the classification of Dalmau and Jonsson [9], hardness under ETH requires a tight reduction from counting cliques, which we will present first. In particular we use a trick inspired by Lemma 1.11 in [7] to make the reduction parsimonious which is required for the extension to modular counting in the subsequent section.

Lemma 11 *There exists an algorithm that, given a positive integer $\ell > 1$ and a graph G with n vertices, computes in time $O(\ell n)$ a $K_{\ell,\ell}$ -colored graph G' with at most $O(\ell n)$ vertices such that the number of cliques of size ℓ in G equals $\#cp\text{-Hom}(K_{\ell,\ell} \rightarrow_{K_{\ell,\ell}} G')$.*

Proof Let the vertex set of G be $\{v_i \mid 1 \leq i \leq n\}$ and let that of $K_{\ell,\ell}$ be $\{a_i, b_i \mid 1 \leq i \leq \ell\}$. We now construct the graph G' on the vertex set $\{u_{i,j}, w_{i,j} \mid 1 \leq i \leq \ell, 1 \leq j \leq n\}$ with a $K_{\ell,\ell}$ -coloring given by $c(u_{i,j}) = a_i$ and $c(w_{i,j}) = b_i$. We add an edge between $u_{i,j}$ and $w_{i',j'}$ if and only if

- either $(i, j) = (i', j')$,
- or $i < i', j < j'$ and the vertices v_j and $v_{j'}$ are adjacent,
- or $i > i', j > j'$ and the vertices v_j and $v_{j'}$ are adjacent.

Let $\{v_{j_1}, \dots, v_{j_\ell}\}$ be an ℓ -clique in G . Assume w.l.o.g. that $j_k < j_{k'}$ for $k < k'$. Then the set $\{u_{1,j_1}, \dots, u_{\ell,j_\ell}, w_{1,j_1}, \dots, w_{\ell,j_\ell}\}$ forms a colorful biclique in G' , so it gives rise to a color-prescribed homomorphism $h \in cp\text{-Hom}(K_{\ell,\ell} \rightarrow_{K_{\ell,\ell}} G')$. Now let $h' \in cp\text{-Hom}(K_{\ell,\ell} \rightarrow_{K_{\ell,\ell}} G')$ be a color-prescribed homomorphism. Then there has to be the following colorful biclique in G' :

$$\{u_{1,\alpha_1}, \dots, u_{\ell,\alpha_\ell}, w_{1,\beta_1}, \dots, w_{\ell,\beta_\ell}\}.$$

We first see that for every i we have $\alpha_i = \beta_i$ since there has to be an edge between u_{i,α_i} and w_{i,β_i} . All other edges now directly imply that $\{v_{\alpha_1}, \dots, v_{\alpha_\ell}\}$ is a clique of size ℓ in G . Furthermore the edges enforce $\alpha_i < \beta_{i'} = \alpha_{i'}$ for every $i < i'$, i.e. that the α_i are sorted in increasing order and thus there is a one-to-one correspondence between homomorphisms in $cp\text{-Hom}(K_{\ell,\ell} \rightarrow_{K_{\ell,\ell}} G')$ and ℓ -cliques in G . \square

Proof of Theorem 1 Let Φ and \mathcal{K} be as given in Theorem 1. We define a class of graphs \mathcal{H} as follows:

$$\mathcal{H} = \{K_{t,t} \mid t \in \mathcal{K}\}.$$

By the reductions sequence (7), given by Lemmas 4, 9 and 10, we obtain that $\#\text{HOM}(\mathcal{H}) \leq_{\text{T}}^{\text{fpt}} \#\text{INDSUB}(\Phi)$. As Φ is computable, \mathcal{H} is recursively enumerable. Furthermore, as \mathcal{K} is infinite, we have that there are arbitrary large bicliques in \mathcal{H} and, in particular, the treewidth of \mathcal{H} is unbounded. Therefore $\#\text{HOM}(\mathcal{H})$, and hence $\#\text{INDSUB}(\Phi)$, are $\#W[1]$ -hard by the classification of counting homomorphisms due to Dalmau and Jonsson [9]. For the tight bound under ETH, we reduce from the *decision problem* CLIQUE which asks, given G and k , to *decide* whether G contains a clique of size k and which cannot be solved in time $f(k) \cdot n^{o(k)}$ for any computable function f , unless ETH fails [5,6]. Now assume that \mathcal{K} is dense and let (G, k) be an instance of CLIQUE. By density of \mathcal{K} , there exists $\ell \in \mathcal{K}$ such that $k \leq \ell \leq ck$ for some overall constant c independent of k . We construct the graph \tilde{G} from G by adding $\ell - k$ further vertices and adding edges between all new vertices as well as between every pair of an old and a new vertex. It can then easily be verified that G contains a clique of size k if and only if \tilde{G} contains a clique of size ℓ .

Next we apply Lemma 11 to \tilde{G} and ℓ , and obtain an $K_{\ell,\ell}$ -colored graph G' satisfying that the number of ℓ -cliques in \tilde{G} is equal to

$$\#\text{cp-Hom}(K_{\ell,\ell} \rightarrow_{K_{\ell,\ell}} G').$$

Now we invoke Lemmas 9 and 10 to conclude the reduction. In particular, all reductions are tight in the sense that every oracle call for $\#\text{INDSUB}(\Phi)$ in the final part of the reduction is a pair $(\tilde{G}, 2\ell)$ where the number of vertices of \tilde{G} is bounded by $O(\ell \cdot |V(G)|)$. As $\ell \leq ck$ we conclude that every algorithm that solves $\#\text{INDSUB}(\Phi)$ in time $f(k) \cdot n^{o(k)}$ can be used to solve CLIQUE in time $f(k) \cdot n^{o(k)}$ —just check in the end whether the output is a number greater than zero.

Finally, we point out that for both ($\#W[1]$ and ETH) hardness results, the last part of the reduction, that is, Lemma 10 only queries for graphs that are $K_{t,t}$ -colorable and hence bipartite. \square

Note that, in case Φ or its complement is edge-monotone, we only have to find infinitely many prime powers t for which Φ is neither true nor false on the set of all edge-subgraphs of $K_{t,t}$, which is the case for all natural, non-trivial properties that do not rely on the number of vertices in some way.⁸ If Φ (or its complement) is monotone, that is, not only closed under the removal of edges, but also under the removal of vertices, then such artificial properties do not exist and we can state the result more clearly as follows.

Theorem 2 *Let Φ be a computable monotone graph property such that Φ and $\neg\Phi$ hold on infinitely many bipartite graphs. Then $\#\text{INDSUB}(\Phi)$ is $\#W[1]$ -hard and cannot*

⁸ An example of such a “non-natural” case would be the (edge-monotone) property which is equivalent to planarity for all graphs whose number of vertices is contained in the image of the Ackermann function, and which is trivially true for all other vertex counts.

be solved in time $f(k) \cdot n^{o(k)}$ for any computable function f unless the Exponential Time Hypothesis fails. This holds true even if the input graphs to $\# \text{INDSUB}(\Phi)$ are restricted to be bipartite.

Proof Since Φ holds for infinitely many (bipartite) graphs, we have that for each positive integer k , there exists a graph H with at least k vertices such that $\Phi(H) = 1$. Since Φ is monotone, that is, closed under removal of vertices and edges, and since IS_k is a subgraph of H , we have that $\Phi(\text{IS}_k) = 1$.

Next, we claim that $\Phi(K_{t,t}) = 0$ for all but finitely many t . Assume for contradiction that the latter is false, that is, for each t there exists a $t' \geq t$ such that $\Phi(K_{t',t'}) = 1$. However, this implies that for every bipartite graph H , the property Φ is true for a supergraph of H . Since Φ is monotone, we conclude that $\Phi(H) = 1$, contradicting the fact that $\neg\Phi$ holds for infinitely many bipartite graphs.

Consequently, we have shown that $\Phi(\text{IS}_k) = 1$ for all positive integers k and $\Phi(K_{t,t}) = 0$ for all but finitely many t . Hence we can apply Theorem 1 and, in particular, the set \mathcal{K} will contain all but finitely many prime powers and is therefore dense. \square

6 Extension to Modular Counting

In the last part of the paper, we show that our main result (Theorem 1) can easily be extended to counting modulo a fixed prime:

Theorem 5 *Let p be a prime number, let Φ be a computable graph property and let \mathcal{K} be the set of all prime powers $t = p^k$ such that $\Phi(\text{IS}_{2t}) \neq \Phi(K_{t,t})$. If \mathcal{K} is infinite then $\text{Mod}_p \text{INDSUB}(\Phi)$ is $\text{Mod}_p W[1]$ hard. If additionally \mathcal{K} is dense, then it cannot be solved in time $f(k) \cdot n^{o(k)}$ for any computable function f unless ETH fails. This holds true even if the input graphs to $\text{Mod}_p \text{INDSUB}(\Phi)$ are restricted to be bipartite.*

Here $\text{Mod}_p \text{INDSUB}(\Phi)$ asks, given G and k , to compute the number of induced subgraphs with k vertices in G that satisfy Φ modulo p . The parameterized complexity class $\text{Mod}_p W[1]$ is defined by the problem of, given G and k , deciding whether the number of k -cliques in G is 0 modulo p , which is complete for the class (see [3] for $p = 2$ and Chapter 1.2.2 in [7] for the general case).

First of all, we point out that the modular counting version of Theorem 2 follows as corollary from the above theorem in the same way Theorem 2 follows from Theorem 1. For the proof of Theorem 5 we rely on the following fact stating that all required reductions in Sect. 4 work as well in the case of counting modular a prime number.

Fact 1 *Let p be a fixed prime. Then Lemmas 9 and 10 remain true when counting is done modulo p if the graph H is restricted to be $K_{t,t}$ for some prime power $t = p^k$.*

The only two non-trivial observations required to verify Fact 1 are, first, that $\hat{\chi}(\Phi, K_{t,t}) \not\equiv 0 \pmod{p}$ whenever $\Phi(K_{t,t}[\emptyset]) \neq \Phi(K_{t,t})$ (Lemma 1) and, second, that complexity monotonicity (Lemma 7) holds for computation modulo p as well, since non-singularity of the matrix M in the proof is given by Lemma 6 even in case the entries of M are considered to be elements of \mathbb{Z}_p . The last ingredient for the proof

of Theorem 5, in particular for hardness under ETH, requires a method of isolating cliques that works in the parameterized setting. This is given by the following result of Williams et al.

Lemma 12 (Lemma 2.1 in [26]) *Let $p \geq 2$ be an integer, G, H be undirected graphs. Let G' be a random induced subgraph of G such that each vertex is taken with probability $1/2$, independently. If there is at least one induced H in G , the number of induced H in G' is not a multiple of p with probability at least $2^{-|H|}$.*

Proof of Theorem 5 The proof is most similar to the proof of the tight lower bound under ETH in Theorem 1. We start our reduction from the problem of *finding* a clique of size k . In case \mathcal{K} is dense and we aim to establish the ETH hardness result, we perform the following two initial steps before the main reduction:

1. Given G and k , we construct a graph \hat{G} such that G contains a clique of size k if and only if \hat{G} contains a clique of size ℓ where $k \leq \ell \leq ck$ for some overall constant c . The details of the construction are given in the proof of Theorem 1.
2. We use Lemma 12 to isolate an ℓ -clique in \hat{G} , assuming there is any, with high probability.

For the main part of the reduction we then first apply the reduction from counting cliques to counting color-prescribed homomorphisms from the biclique as given by Lemma 11. In particular, this reduction is parsimonious. Finally, we proceed from this point on precisely as in the proof of Theorem 1, the correctness of which follows by Fact 1.

We conclude by pointing out that, in case the randomized construction of Lemma 12 was used, we can perform probability amplification by repeating the final algorithm 2^k times to end up in a constant success probability. \square

7 Conclusion

We have established hardness for $\#INDSUB(\Phi)$ for *any* (edge-)monotone property Φ that is non-trivial on bipartite graphs. In particular, this holds true even if we count modulo a prime and restrict the input graphs to be bipartite as well. Hence, we did not only significantly extend the set of graph properties Φ for which the (parameterized) complexity of $\#INDSUB(\Phi)$ is understood, but we also generalized many of the prior results, such as [13], [19] and parts of [22] to the cases of bipartite input graphs and modular counting.

As a next step towards a proof of Conjecture 1, we suggest the study of properties that are defined by forbidden induced subgraphs, for which the complexity of $\#INDSUB(\Phi)$ is only partially resolved at this point.

Acknowledgements We are very grateful to Radu Curticapean and Holger Dell for fruitful discussions and valuable feedback on early drafts of this work.

Declaration

Conflict of interest The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Abhyankar, S.S.: Lectures on algebra, vol. I. World Scientific Publishing Co. Pte. Ltd., Hackensack, NJ (2006). <https://doi.org/10.1142/9789812773449>
2. Biggs, N.: Algebraic graph theory, 2nd (edn.). Cambridge Mathematical Library. Cambridge University Press, Cambridge (1993). <https://doi.org/10.1017/CBO9780511608704>
3. Björklund, A., Dell, H., Husfeldt, T.: The parity of set systems under random restrictions with applications to exponential time problems. In: Automata, Languages, and Programming—42nd International Colloquium, ICALP 2015, Kyoto, Japan, 6–10 July, 2015, Proceedings, Part I, pp. 231–242 (2015). https://doi.org/10.1007/978-3-662-47672-7_19
4. Chen, H., Mengel, S.: 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, pp. 315–326 (2016). <https://doi.org/10.1145/2902251.2902279>
5. Chen, J., Chor, B., Fellows, M., Huang, X., Juedes, D.W., Kanj, I.A., Xia, G.: Tight lower bounds for certain parameterized NP-hard problems. *Inf. Comput.* **201**(2), 216–231 (2005). <https://doi.org/10.1016/j.ic.2005.05.001>
6. Chen, J., Huang, X., Kanj, I.A., Xia, G.: Strong computational lower bounds via parameterized complexity. *J. Comput. Syst. Sci.* **72**(8), 1346–1367 (2006). <https://doi.org/10.1016/j.jcss.2006.04.007>
7. Curticapean, R.: The simple, little and slow things count: On parameterized counting complexity. Ph.D. thesis, Saarland University (2015). <http://scidok.sulb.uni-saarland.de/volltexte/2015/6217/>
8. Curticapean, R., Dell, H., Marx, D.: 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, 19–23 June, 2017, pp. 210–223 (2017). <https://doi.org/10.1145/3055399.3055502>
9. Dalmau, V., Jonsson, P.: The complexity of counting homomorphisms seen from the other side. *Theor. Comput. Sci.* **329**(1–3), 315–323 (2004). <https://doi.org/10.1016/j.tcs.2004.08.008>
10. Edmonds, J.: Paths, trees, and flowers. *Can. J. Math.* **17**, 449–467 (1965). <https://doi.org/10.4153/CJM-1965-045-4>
11. Flum, J., Grohe, M.: The parameterized complexity of counting problems. *SIAM J. Comput.* **33**(4), 892–922 (2004). <https://doi.org/10.1137/S0097539703427203>
12. Flum, J., Grohe, M.: Parameterized complexity theory. Texts in Theoretical Computer Science. An EATCS Series. Springer (2006). <https://doi.org/10.1007/3-540-29953-X>
13. Jerrum, M., Meeks, K.: The parameterised complexity of counting connected subgraphs and graph motifs. *J. Comput. Syst. Sci.* **81**(4), 702–716 (2015). <https://doi.org/10.1016/j.jcss.2014.11.015>
14. Jerrum, M., Meeks, K.: Some hard families of parameterized counting problems. *TOCT* **7**(3), 11:1–11:18 (2015). <https://doi.org/10.1145/2786017>
15. Jerrum, M., Meeks, K.: The parameterised complexity of counting even and odd induced subgraphs. *Combinatorica* **37**(5), 965–990 (2017). <https://doi.org/10.1007/s00493-016-3338-5>
16. Kahn, J., Saks, M.E., Sturtevant, D.: A topological approach to evasiveness. *Combinatorica* **4**(4), 297–306 (1984). <https://doi.org/10.1007/BF02579140>
17. Lovász, L.: Large networks and graph limits. *Colloquium Publications*, vol. 60. American Mathematical Society (2012). <http://www.ams.org/bookstore-getitem/item=COLL-60>
18. McCartin, C.: Parameterized counting problems. *Ann. Pure Appl. Logic* **138**(1–3), 147–182 (2006). <https://doi.org/10.1016/j.apal.2005.06.010>

19. Meeks, K.: The challenges of unbounded treewidth in parameterised subgraph counting problems. *Discrete Appl. Math.* **198**, 170–194 (2016). <https://doi.org/10.1016/j.dam.2015.06.019>
20. Miller, C.A.: Evasiveness of graph properties and topological fixed-point theorems. *Found. Trends Theor. Comput. Sci.* **7**(4), 337–415 (2013). <https://doi.org/10.1561/04000000055>
21. Rivest, R.L., Vuillemin, J.: On recognizing graph properties from adjacency matrices. *Theoret. Comput. Sci.* **3**(3), 371–384 (1976/77). [https://doi.org/10.1016/0304-3975\(76\)90053-0](https://doi.org/10.1016/0304-3975(76)90053-0)
22. Roth, M., Schmitt, J.: Counting induced subgraphs: A topological approach to $\#W[1]$ -hardness. In: 13th International Symposium on Parameterized and Exact Computation, IPEC 2018, 20–24 August, 2018, Helsinki, Finland, pp. 24:1–24:14 (2018). <https://doi.org/10.4230/LIPIcs.IPEC.2018.24>
23. Toda, S.: PP is as hard as the polynomial-time hierarchy. *SIAM J. Comput.* **20**(5), 865–877 (1991). <https://doi.org/10.1137/0220053>
24. Valiant, L.G.: The complexity of computing the permanent. *Theor. Comput. Sci.* **8**, 189–201 (1979). [https://doi.org/10.1016/0304-3975\(79\)90044-6](https://doi.org/10.1016/0304-3975(79)90044-6)
25. Weir, A.J.: The Sylow subgroups of the symmetric groups. *Proc. Amer. Math. Soc.* **6**, 534–541 (1955). <https://doi.org/10.2307/2033425>
26. Williams, V.V., Wang, J.R., Williams, R.R., Yu, H.: Finding four-node subgraphs in triangle time. In: Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, 4–6 January, 2015, pp. 1671–1680 (2015). <https://doi.org/10.1137/1.9781611973730.111>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.