

COUNTING SUBGRAPHS IN SOMEWHERE DENSE GRAPHS*

MARCO BRESSAN[†], LESLIE ANN GOLDBERG[‡], KITTY MEEKS[§], AND MARC ROTH[¶]

Abstract. We study the problems of counting copies and induced copies of a small pattern graph H in a large host graph G . Recent work fully classified the complexity of those problems according to structural restrictions on the patterns H . In this work, we address the more challenging task of analysing the complexity for restricted patterns *and* restricted hosts. Specifically we ask which families of allowed patterns and hosts imply fixed-parameter tractability, i.e., the existence of an algorithm running in time $f(H) \cdot |G|^{O(1)}$ for some computable function f . Our main results present exhaustive and explicit complexity classifications for families that satisfy natural closure properties. Among others, we identify the problems of counting small matchings and independent sets in subgraph-closed graph classes \mathcal{G} as our central objects of study and establish the following crisp dichotomies as consequences of the Exponential Time Hypothesis:

- Counting k -matchings in a graph $G \in \mathcal{G}$ is fixed-parameter tractable if and only if \mathcal{G} is nowhere dense.
- Counting k -independent sets in a graph $G \in \mathcal{G}$ is fixed-parameter tractable if and only if \mathcal{G} is nowhere dense.

Moreover, we obtain almost tight conditional lower bounds if \mathcal{G} is somewhere dense, i.e., not nowhere dense. These base cases of our classifications subsume a wide variety of previous results on the matching and independent set problem, such as counting k -matchings in bipartite graphs (Curticapean, Marx; FOCS 14), in F -colourable graphs (Roth, Wellnitz; SODA 20), and in degenerate graphs (Bressan, Roth; FOCS 21), as well as counting k -independent sets in bipartite graphs (Curticapean et al.; Algorithmica 19).

At the same time our proofs are much simpler: using structural characterisations of somewhere dense graphs, we show that a colourful version of a recent breakthrough technique for analysing pattern counting problems (Curticapean, Dell, Marx; STOC 17) applies to *any* subgraph-closed somewhere dense class of graphs, yielding a unified view of our current understanding of the complexity of subgraph counting.

Key words. counting problems, somewhere dense graphs, parameterised complexity theory

MSC codes. 68Q17, 68Q25, 68R10

1. Introduction. We study the following subgraph counting problem: given two graphs H and G , compute the number of copies of H in G . For several decades this problem has received widespread attention from the theoretical community, leading to a rich algorithmic toolbox that draws from different techniques [50, 3, 10, 40] and to deep structural results in parameterised complexity theory [28, 18]. Since it was discovered that subgraph counts reveal global properties of complex networks [46, 47], subgraph counting has also found several applications in fields such as biology [2, 57] genetics [59], phylogeny [41], and data mining [60]. Unfortunately, the subgraph counting problem is in general intractable, since it contains as special cases hard problems such as CLIQUE. This does not mean however that the problem is *always* intractable; it just means that it is tractable when the pattern H is restricted to certain

*Preprint available at <https://arxiv.org/abs/2209.03402>. A preliminary version of this paper appeared in the Proceedings of the 14th Conference on Innovations in Theoretical Computer Science (ITCS 2023) [8]. This research was funded in whole, or in part, by EPSRC grant EP/V032305/1. For the purpose of Open Access, the authors have applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission. All data is provided in full in the results section of this paper.

[†]Department of Computer Science, University of Milan, Italy

[‡]Department of Computer Science, University of Oxford, United Kingdom

[§]School of Computing Science, University of Glasgow, United Kingdom

[¶]School of Electronic Engineering and Computer Science, Queen Mary University of London, United Kingdom

41 graph families. Identifying these families of patterns that are efficiently countable has
 42 been a key question for the last twenty years. A long stream of research eventually
 43 showed that, unless standard conjectures fail, subgraph counting is tractable only for
 44 very restricted families of patterns [28, 23, 14, 20, 39, 45, 18, 55, 30].

45 To circumvent this “wall of intractability”, in this work we restrict both the
 46 family of the pattern H and the family of the host G . Formally, given two classes
 47 of graphs \mathcal{H} and \mathcal{G} , we study the problems $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$, $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$, and
 48 $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$, defined as follows. For all of them, the input is a pair (H, G) with
 49 $H \in \mathcal{H}$ and $G \in \mathcal{G}$. The outputs are respectively the number of subgraphs of G
 50 isomorphic to H , denoted by $\#\text{Sub}(H \rightarrow G)$, the number of induced subgraphs of G
 51 isomorphic to H , denoted by $\#\text{IndSub}(H \rightarrow G)$, and the number of homomorphisms
 52 (edge-preserving maps) from H to G , denoted by $\#\text{Hom}(H \rightarrow G)$. Our goal is to
 53 determine for which \mathcal{H} and \mathcal{G} these three problems are tractable. To formalize what
 54 we mean by tractable, we adopt the framework of parameterized complexity [22]: we
 55 say that a problem is *fixed-parameter tractable*, or in the class FPT, if it is solvable
 56 in time $f(|H|) \cdot |G|^{O(1)}$ for some computable function f (see Section 2 for a complete
 57 introduction). For instance, we consider as tractable a running time of $2^{O(|H|)} \cdot |G|$
 58 but not one of $|G|^{O(|H|)}$. This captures the intuition that H is “small” compared
 59 to G , and is the main theoretical framework for subgraph counting [28]. Thus, the
 60 goal of this work is understanding the fixed-parameter tractability of $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$,
 61 $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$, and $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ as a function of \mathcal{H} and \mathcal{G} . Moreover, when
 62 those problems are not fixed-parameter tractable we aim to show that they are hard
 63 for the complexity class $\#\text{W}[1]$, which can be thought of as the equivalent of NP for
 64 parameterized counting.

65 We first briefly discuss which properties of \mathcal{G} are worthy of attention. When \mathcal{G}
 66 is the class of all graphs, it is well known that each of the three problems is either
 67 FPT or $\#\text{W}[1]$ -hard depending on whether certain structural parameters of \mathcal{H} (such
 68 as treewidth or vertex cover number) are bounded. Thus, when \mathcal{G} is the class of all
 69 graphs, the problem is solved. However, when \mathcal{G} is arbitrary, no such characterization
 70 is known. This is partly due to the fact that “natural” structural properties related
 71 to subgraph counting are harder to find for \mathcal{G} than for \mathcal{H} ; subgraph counting algo-
 72 rithms themselves usually exploit the structure of H but not that of G (think of tree
 73 decompositions). There is however one deep structural property that, if held by \mathcal{G} ,
 74 yields tractability: the property of being *nowhere dense*, introduced by Nešetřil and
 75 Ossona de Mendez [48]. In a nutshell \mathcal{G} is nowhere dense if, for all $r \in \mathbb{N}_0$, its members
 76 do not contain as subgraphs the r -subdivisions of arbitrarily large cliques; it can be
 77 shown that this generalizes several natural definitions of sparsity, including having
 78 bounded degree or bounded local treewidth, or excluding some topological minor. In
 79 a remarkable result, Nešetřil and Ossona de Mendez proved:¹

THEOREM 1.1 (Theorem 18.9 in [49]). *If \mathcal{G} is nowhere dense then $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$, $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$, and $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ are fixed-parameter tractable and can be solved in time $f(|H|) \cdot |V(G)|^{1+o(1)}$ for some computable function f .*

80 Thus the case of nowhere dense \mathcal{G} is closed, and we can focus on its complement —
 81 the case where \mathcal{G} is *somewhere dense*. Hence the question studied in this work is:
 82 when are $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$, $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$, and $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ fixed-parameter
 83 tractable, provided \mathcal{G} is somewhere dense?

¹In the realm of decision problems, an even more general meta-theorem is known for first-order model-checking on nowhere dense graphs [36].

84 **1.1. Our Results.** We prove dichotomies for $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$, $\#\text{INDSUB}(\mathcal{H} \rightarrow$
85 $\mathcal{G})$, and $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ into FPT and $\#\text{W}[1]$ -hard cases, assuming that \mathcal{G} is some-
86 where dense. It is known [56] that a fully general dichotomy is impossible even
87 assuming that \mathcal{G} is somewhere dense; thus we focus on the natural cases where \mathcal{H}
88 and/or \mathcal{G} are monotone (closed under taking subgraphs) or hereditary (closed under
89 taking induced subgraphs). Our dichotomies are expressed in terms of the finiteness
90 of combinatorial parameters of \mathcal{H} and \mathcal{G} , such as their clique number or their induced
91 matching number. Existing complexity dichotomies for subgraph counting are based
92 on using interpolation to evaluate linear combinations of homomorphism counts [18].
93 This technique has been exploited for families of host graphs that are closed under
94 tensoring — the closure is used to create new instances for the interpolation. The host
95 graphs in our dichotomy theorems do not have this closure property. Nevertheless, we
96 obtain a dichotomy for all somewhere dense classes using a combination of techniques
97 involving graph fractures and colourings.

98 The rest of this section presents our main conceptual contribution (Section 1.1.1),
99 gives a detailed walk-through of our complexity dichotomies (Section 1.1.2, Sec-
100 tion 1.1.3, Section 1.1.4), provides some context (Section 1.2), and overviews the
101 techniques behind our proofs (Section 1.3). For full proofs of our claims see Section 2
102 onward.

103 *Basic preliminaries.* We concisely state some necessary definitions and obser-
104 vations which are given in more detail in Section 2. We denote by \mathcal{U} the class of
105 all graphs. We denote by $\omega(G)$, $\alpha(G)$, $\beta(G)$, and $m(G)$ respectively the clique, in-
106 dependence, biclique, and matching number of a graph G . The notation extends to
107 graph classes by taking the supremum over their elements. Induced versions of those
108 quantities are identified by the subscript ind (for instance, m_{ind} denotes the induced
109 matching number). G^r denotes the r -subdivision of G , and $F \times G$ denotes the tensor
110 product of F and G . All of our lower bounds assume the Exponential Time Hypothe-
111 sis (ETH) [38]; and most of them rule out algorithms running in time $f(k) \cdot n^{o(k/\log k)}$
112 for any function f , and are therefore tight except possibly for a $O(\log k)$ factor in the
113 exponent.² All of our $\#\text{W}[1]$ -hardness results are actually $\#\text{W}[1]$ -completeness re-
114 sults; this holds because $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$, $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$, and $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ are
115 always in $\#\text{W}[1]$ due to a characterisation of $\#\text{W}[1]$ via parameterised model-counting
116 problems (see [29, Chapter 14]).

117 **1.1.1. Simpler Hardness Proofs for More Graph Families.** Our first and
118 most conceptual contribution is a novel approach to proving hardness of parameter-
119 ized subgraph counting problems for somewhere-dense families of host graphs. This
120 approach allows us to significantly generalize existing results while simultaneously
121 yielding surprisingly simpler proofs.

122 The starting point is the observation that proving intractability results for param-
123 eterized counting problems is discouragingly difficult, as it often requires tedious and
124 involved arguments. For instance, after Flum and Grohe conjectured that counting k -
125 matchings is $\#\text{W}[1]$ -hard [28], the first proof required nine years and relied on sophis-
126 ticated algebraic techniques [15]. This partially changed in 2017 when Curticapean,
127 Dell and Marx [18] showed how to express a subgraph count $\#\text{Sub}(H \rightarrow G)$ as linear
128 combination of homomorphism counts $\sum_F a_F \cdot \#\text{Hom}(F \rightarrow G)$. They showed that
129 computing this linear combination has the same complexity as computing the hardest
130 term $\#\text{Hom}(F \rightarrow G)$ such that $a_F \neq 0$. A similar claim holds for induced subgraph

²This $O(\log k)$ gap is not an artifact of our proofs, but a consequence of the well-known open problem “Can you beat treewidth?” [43, 44].

131 counts as well. Thanks to this technique one can prove intractability of several sub-
 132 graph counting problems, including for instance the problem of counting k -matchings.³
 133 These hardness results ultimately yielded complexity dichotomies for general subgraph
 134 counting problems, including notably $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ and $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ when \mathcal{G}
 135 is the class of all graphs.

136 The technique of [18] does not work for proving hardness of $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ and
 137 $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ when $\mathcal{G} \neq \mathcal{U}$. Indeed, one caveat of that technique is that the
 138 family of host graphs \mathcal{G} must satisfy certain conditions. One of those conditions is
 139 that \mathcal{G} is closed under tensoring, i.e., that $G \times G' \in \mathcal{G}$ for all $G \in \mathcal{G}$ and all $G' \in \mathcal{U}$.
 140 The reason is that the interpolation relies on evaluating, say, $\text{Sub}(H \rightarrow G \times G_i)$ for
 141 several carefully chosen graphs G_i , with the goal of constructing a certain invertible
 142 system of linear equations; for this to yield a reduction towards counting patterns
 143 in graphs from \mathcal{G} , it is crucial that $G \times G_i \in \mathcal{G}$ for all such G_i (Section 1.3 gives
 144 a concrete example using the problem of counting k -matchings). This is why the
 145 technique of [18] works smoothly for $\mathcal{G} = \mathcal{U}$; closure under tensoring holds trivially in
 146 that case. But many other natural graph families \mathcal{G} are not closed under tensoring,
 147 including somewhere dense ones (for instance, the family of d -degenerate graphs for
 148 any fixed integer $d \geq 2$). Until now, this has been the main obstacle towards proving
 149 hardness of subgraph counting for arbitrary somewhere dense graph families. The
 150 central insight of our work is that this obstacle can be circumvented in a surprisingly
 151 simple way. Using well-established results from the theory of sparsity, we prove the
 152 following claim, which we explain in detail in Section 1.3:

153 *Every monotone and somewhere dense class of graphs is closed*
 154 *under vertex-colourful tensor products of subdivided graphs.*

155 Ignoring for a moment its technicalities, this result allows us to lift the interpola-
 156 tion technique via graph tensors to *any* monotone somewhere dense class of host
 157 graphs, including for instance the aforementioned class of d -degenerate graphs. In
 158 turn this yields complexity classifications for $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$, $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$, and
 159 $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ that subsume and significantly strengthen almost all classifications
 160 known in the literature (see below). Moreover, our approach yields simple and almost
 161 self-contained proofs, helping understand the underlying causes of the hardness.

162 **1.1.2. The Complexity of $\#\text{Sub}(\mathcal{H} \rightarrow \mathcal{G})$.** This section presents our results
 163 on the fixed-parameter tractability of $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$. We start by presenting a
 164 minimal⁴ family \mathcal{H} for which hardness holds: the family of all k -matchings (or 1-
 165 regular graphs). In this case we also denote $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ as $\#\text{MATCH}(\mathcal{G})$. In the
 166 foundational work by Flum and Grohe [28], $\#\text{MATCH}(\mathcal{U})$ was identified as a central
 167 problem because of the significance of its classical counterpart (counting the number
 168 of perfect matchings); a series of works then identified $\#\text{MATCH}(\mathcal{U})$ as the minimal
 169 intractable case [15, 20, 18]. In this work, we show that $\#\text{MATCH}(\mathcal{G})$ is the minimal
 170 hard case for *every* class \mathcal{G} that is monotone and somewhere dense:

171 **THEOREM 1.2.** *Let \mathcal{G} be a monotone class of graphs⁵ and assume that ETH holds.*
 172 *Then $\#\text{MATCH}(\mathcal{G})$ is fixed-parameter tractable if and only if \mathcal{G} is nowhere dense.*

³In the field of database theory a similar technique expressing answers to unions of conjunctive queries as linear combinations of answers of conjunctive queries was independently discovered by Chen and Mengel [11].

⁴Minimal means that, for every class \mathcal{H}' , $\#\text{SUB}(\mathcal{H}' \rightarrow \mathcal{G})$ is intractable if and only if the monotone closure of \mathcal{H}' includes \mathcal{H} . The same holds for $\#\text{INDSUB}$ with “monotone” replaced by “hereditary”.

⁵We emphasize that we do not need our classes to be computable or recursively enumerable. This is due to the assumed closure properties of the classes.

173 More precisely, if \mathcal{G} is nowhere dense then $\#\text{MATCH}(\mathcal{G})$ can be solved in time $f(k) \cdot$
 174 $|V(G)|^{1+o(1)}$ for some computable function f ; otherwise $\#\text{MATCH}(\mathcal{G})$ is $\#\text{W}[1]$ -hard
 175 and cannot be solved in time $f(k) \cdot |G|^{\alpha(k/\log k)}$ for any function f .

176 Theorem 1.2 subsumes the existing intractability results for counting k -matchings
 177 in bipartite graphs [20], in F -colourable graphs [56], in bipartite graphs with one-
 178 sided degree bounds [19], and in degenerate graphs [9]. It also strengthens the latter
 179 result: while [9] establishes hardness of counting k -matchings in ℓ -degenerate graphs
 180 for $k + \ell$ as a parameter, Theorem 1.2 yields hardness for d -degenerate graphs for
 181 every fixed $d \geq 2$.⁶ Additionally, we show that Theorem 1.2 cannot be strengthened to
 182 achieve polynomial-time tractability of $\#\text{MATCH}(\mathcal{G})$ for nowhere dense and monotone
 183 \mathcal{G} , unless $\#\text{P} = \text{P}$.

184 As a consequence of Theorem 1.2 we obtain, for hereditary \mathcal{H} , an exhaustive and
 185 detailed classification of the complexity of $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ as a function of invariants
 186 of \mathcal{G} and \mathcal{H} .

187 **THEOREM 1.3.** *Let \mathcal{H} and \mathcal{G} be graph classes such that \mathcal{H} is hereditary and \mathcal{G} is*
 188 *monotone. Then the complexity of $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ is exhaustively classified by Table 1.*

	\mathcal{G} n. dense	\mathcal{G} s. dense $\omega(\mathcal{G}) = \infty$	\mathcal{G} s. dense $\omega(\mathcal{G}) < \infty$ $\beta(\mathcal{G}) = \infty$	\mathcal{G} s. dense $\omega(\mathcal{G}) < \infty$ $\beta(\mathcal{G}) < \infty$
$m(\mathcal{H}) < \infty$	P	P	P	P
$m_{\text{ind}}(\mathcal{H}) = \infty$	FPT	hard	hard	hard
$m_{\text{ind}}(\mathcal{H}) < \infty$ $\beta_{\text{ind}}(\mathcal{H}) = \infty$	P	hard [†]	hard [†]	P
Otherwise	P	hard [†]	P	P

TABLE 1

The complexity of $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ for hereditary \mathcal{H} and monotone \mathcal{G} . Here “hard” means $\#\text{W}[1]$ -hard and, unless ETH fails, without an algorithm running in time $f(|H|) \cdot |G|^{\alpha(|V(H)|/\log |V(H)|)}$; “hard[†]” means the same, but without an algorithm running in $f(|H|) \cdot |G|^{\alpha(|V(H)|)}$.

189 Note that the unique fixed-parameter tractability result in Table 1 is a “real” FPT
 190 case: we can show that, unless $\text{P} = \#\text{P}$, it is in FPT but not in P. We point out
 191 that the contributions in this work are the hardness results in the third and fourth
 192 column, that is, for the cases in which \mathcal{G} is somewhere dense, but not the class of all
 193 graphs. (For monotone \mathcal{G} , $\omega(\mathcal{G}) = \infty$ implies that \mathcal{G} is the class of all graphs.)

194 From the classification of Theorem 1.3 one can derive interesting corollaries. For
 195 example, when \mathcal{H} and \mathcal{G} are monotone one has essentially the same classification of
 196 the case $\mathcal{G} = \mathcal{U}$: only the boundedness of the matching number of \mathcal{H} (or equivalently,
 197 of its vertex-cover number) counts [20].

198 **THEOREM 1.4.** *Let \mathcal{H} and \mathcal{G} be monotone classes of graphs and assume that ETH*
 199 *holds. Then $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ is fixed-parameter tractable if $m(\mathcal{H}) < \infty$ or \mathcal{G} is nowhere*
 200 *dense; otherwise $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ is $\#\text{W}[1]$ -complete and cannot be solved in time*

⁶The class of all d -degenerate graphs is somewhere dense for all $d \geq 2$.

201 $f(|H|) \cdot |G|^{\alpha(|V(H)|/\log(|V(H)|))}$ for any function f .

202 We conclude by remarking that Table 1 and the proofs of its bounds suggest the
203 existence of three general algorithmic strategies for subgraph counting:

- 204 1. If \mathcal{G} is nowhere dense (first column of Table 1), then one can use the FPT
205 algorithm of Theorem 1.1, based on Gaifman’s locality theorem for first-order
206 formulas and the local sparsity of nowhere dense graphs (see [49]).
- 207 2. If $m(\mathcal{H}) < \infty$ (first row of Table 1), then one can use the polynomial-time
208 algorithm of Curticapean and Marx [20], based on guessing the image of a
209 maximum matching of H and counting its extensions via dynamic program-
210 ming.
- 211 3. All remaining entries marked as “P” are shown to be essentially trivial. Con-
212 cretely, we will rely on Ramsey’s theorem to prove that minor modifications
213 of the naive brute-force approach yield polynomial-time algorithms for those
214 cases.

215 **1.1.3. The Complexity of $\#\text{IndSub}(\mathcal{H} \rightarrow \mathcal{G})$.** In the previous section we
216 proved that, when \mathcal{G} is somewhere dense, k -matchings are the minimal hard family of
217 patterns for $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$. In this section we show that k -independent sets play a
218 similar role for $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$. Let $\#\text{INDSET}(\mathcal{G}) = \#\text{INDSUB}(\mathcal{I} \rightarrow \mathcal{G})$ where \mathcal{I}
219 is the set of all all independent sets (or 0-regular graphs). We prove:

220 **THEOREM 1.5.** *Let \mathcal{G} be a monotone class of graphs and assume that ETH holds.*
221 *Then $\#\text{INDSET}(\mathcal{G})$ is fixed-parameter tractable if and only if \mathcal{G} is nowhere dense.*
222 *More precisely, if \mathcal{G} is nowhere dense then $\#\text{INDSET}(\mathcal{G})$ can be solved in time $f(k) \cdot$
223 $|V(G)|^{1+o(1)}$ for some computable function f ; otherwise $\#\text{INDSET}(\mathcal{G})$ cannot be solved
224 in time $f(k) \cdot |G|^{\alpha(k/\log k)}$ for any function f .*

225 This result subsumes the intractability result for counting k -independent sets in bi-
226 partite graphs of [17]. It also strengthens the result of [9], which shows $\#\text{INDSET}(\mathcal{G})$
227 is hard when parameterized by $k + d$ where d is the degeneracy of G . More precisely, [9]
228 does not imply that $\#\text{INDSET}(\mathcal{G})$ is hard when \mathcal{G} is the class of d -degenerate graphs,
229 for any $d \geq 2$. In contrast to this, Theorem 1.5 proves such hardness for *every* $d \geq 2$.
230 Finally, we point out that the FPT case of Theorem 1.5 is not in P unless $\text{P} = \#\text{P}$.

231 As consequence of Theorem 1.5, when \mathcal{H} is hereditary (and thus in particular
232 monotone) we obtain:

233 **THEOREM 1.6.** *Let \mathcal{H} and \mathcal{G} be classes of graphs such that \mathcal{H} is hereditary and \mathcal{G}
234 is monotone. Then the complexity of $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ is exhaustively classified by
235 Table 2.*

236 **1.1.4. The Complexity of $\#\text{Hom}(\mathcal{H} \rightarrow \mathcal{G})$.** Finally, we study the parameter-
237 ized complexity of $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$. We denote by $\text{tw}(H)$ the treewidth of a graph H .
238 Informally, graphs of small treewidth admit a decomposition with small separators,
239 which allows for efficient dynamic programming. In this work we use treewidth in a
240 purely black-box fashion (e.g. via excluded-grid theorems); for its formal definition
241 see [22, Chapter 7]. We prove:

	\mathcal{G} n. dense	\mathcal{G} s. dense $\omega(\mathcal{G}) = \infty$	\mathcal{G} s. dense $\omega(\mathcal{G}) < \infty$ $\alpha(\mathcal{G}) = \infty$
$ \mathcal{H} < \infty$	P	P	P
$\alpha(\mathcal{H}) = \infty$	FPT	hard [†]	hard
Otherwise	P	hard [†]	P

TABLE 2

The complexity of $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ for hereditary \mathcal{H} and monotone \mathcal{G} . Here “hard” means $\#\text{W}[1]$ -hard and, unless ETH fails, without an algorithm running in time $f(|H|) \cdot |G|^{\mathcal{O}(|V(H)|/\log |V(H)|)}$; “hard[†]” means the same, but without an algorithm running in $f(|H|) \cdot |G|^{\mathcal{O}(|V(H)|)}$.

THEOREM 1.7. Let \mathcal{H} and \mathcal{G} be monotone classes of graphs.

1. If \mathcal{G} is nowhere dense then $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ is fixed-parameter tractable and can be solved in time $f(|H|) \cdot |V(G)|^{1+\mathcal{O}(1)}$ for some computable function f .
2. If $\text{tw}(\mathcal{H}) < \infty$ then $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ is solvable in polynomial time, and if a tree decomposition of H of width t is given, then it can be solved in time $|H|^{\mathcal{O}(1)} \cdot |V(G)|^{t+1}$.
3. If \mathcal{G} is somewhere dense and $\text{tw}(\mathcal{H}) = \infty$ then $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ is $\#\text{W}[1]$ -hard and, assuming ETH, cannot be solved in time $f(|H|) \cdot |G|^{\mathcal{O}(\text{tw}(H))}$ for any function f .

(The novel part is 3.; we included 1. and 2. to provide the complete picture.)

242 Unfortunately, in contrast to $\#\text{SUB}$ and $\#\text{INDSUB}$, we do not know how to extend
243 Theorem 1.7 to hereditary \mathcal{H} . We point out however that for hereditary \mathcal{H} the finite-
244 ness of $\text{tw}(\mathcal{H})$ cannot be the correct criterion: if \mathcal{H} is the set of all complete graphs
245 and \mathcal{G} is the set of all bipartite graphs, then \mathcal{H} is hereditary and $\text{tw}(\mathcal{H}) = \infty$, but
246 $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ is easy since $|V(H)| \leq 2$ or $\#\text{Hom}(H \rightarrow G) = 0$. More generally, the
247 complexity of $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ appears to be far from completely understood for arbitrary
248 classes \mathcal{H} . In fact, it has been recently posed as an open problem even for specific
249 monotone and somewhere dense \mathcal{G} such as the family of d -degenerate graphs [9, 4].
250 There is some evidence that the finiteness of induced grid minors is the right criterion
251 for tractability [9].

252 In what follows we provide a detailed exposition of our proof techniques, starting
253 with a brief summary of the state of the art.

254 **1.2. Related Work.** The general idea of using interpolation as a reduction
255 technique for counting problems dates back to the foundational work of Valiant [61].
256 Roughly speaking, the key to interpolation is constructing a system of linear equations
257 that is invertible and thus has a unique solution. For example, in the classic case of
258 polynomial interpolation (where one has to infer the coefficients of a univariate poly-
259 nomial given an oracle that evaluates it) the system corresponds to a Vandermonde
260 matrix, which is nonsingular and thus invertible. In the case of linear combinations
261 of homomorphism counts, an invertible system of linear equations can be construc-
262 ted via graph tensoring arguments, as proven implicitly by works of Lovász (see e.g.
263 [42, Chapters 5 and 6]). It was then discovered by Curticapean et al. in [18] that
264 these interpolation arguments could be extended to subgraph and induced subgraph

265 counts, by showing that those counts may be expressed as linear combinations of ho-
 266 momorphism counts. Using this fact, they proved that interpolation through graph
 267 tensoring applies to a wide variety of parameterised subgraph counting problems.
 268 However, their technique fails when one restrict the class of host graphs \mathcal{G} , see the
 269 discussion in Section 1.1.1; our work shows how to circumvent this obstacle.

270 The idea of using graph subdivisions for proving hardness results appeared in
 271 the context of linear-time subgraph counting in degenerate graphs [5, 6, 4]. For
 272 example, [5] observed that counting triangles in general graphs, which is conjectured
 273 not to admit a linear time algorithm, reduces in linear time to counting 6-cycles in
 274 degenerate graphs by subdividing each edge once (which always yields a 2-degenerate
 275 graph). Our work makes heavy use of graph subdivisions as well, although in a more
 276 sophisticated fashion. This is not surprising since, for each $d \geq 2$, the class of d -
 277 degenerate graphs constitutes an example of a monotone somewhere dense class of
 278 graphs.

279 **1.3. Overview of Our Techniques.** The present section expands upon Section
 280 1.1.1 and gives a detailed technical overview of our proofs of hardness for $\#\text{SUB}(\mathcal{H} \rightarrow$
 281 $\mathcal{G})$ and $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ (Section 1.3.1) and for $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ (Section 1.3.2).
 282 The main contribution of our work is these hardness proofs. The upper bounds hold
 283 from (simple adaptations of) previous work.

284 **1.3.1. Classifying Subgraph and Induced Subgraph Counting.** We start
 285 by analysing a simple case. Recall that a graph family \mathcal{G} is somewhere dense if, for
 286 some $r \in \mathbb{N}_0$, for all $k \in \mathbb{N}$ there is a $G \in \mathcal{G}$ such that K_k^r is a subgraph of G . From
 287 this characterization it is immediate that, if \mathcal{G} is somewhere dense *and monotone*,
 288 then it contains the r -subdivisions of every graph. In turn, this implies that detecting
 289 subdivisions of cliques in \mathcal{G} is at least as hard as the parameterised clique problem [27].
 290 Since the parameterised clique problem is W[1]-hard, we deduce that $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$
 291 and $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ are intractable when $\mathcal{H} = \{K_k^r : k, r \in \mathbb{N}\}$ and \mathcal{G} is monotone
 292 and somewhere dense. Unfortunately, it is unclear how to extend this approach to
 293 arbitrary \mathcal{H} , since the elements of \mathcal{H} are not necessarily r -subdivisions of graphs that
 294 are hard to count. To show how this obstacle can be overcome, we will focus on
 295 $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ when \mathcal{H} is the class of k -matchings, $\mathcal{M} = \{M_k : k \in \mathbb{N}\}$; in other
 296 words, on the problem of counting k -matchings, $\#\text{SUB}(\mathcal{M} \rightarrow \mathcal{G})$. This problem will
 297 turn out to be the minimal hard case for $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$, and its analysis will contain
 298 the key ingredients of our proof. The proof for $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ will be similar.

299 Let us start by outlining the hardness proof of $\#\text{SUB}(\mathcal{M} \rightarrow \mathcal{G})$ when $\mathcal{G} = \mathcal{U}$, by
 300 using the interpolation technique discussed in Section 1.2. From [18], we know that
 301 for every $k \in \mathbb{N}$ there is a function $a_k : \mathcal{U} \rightarrow \mathbb{Q}$ with finite support such that, for every
 302 $G \in \mathcal{U}$,

$$303 \quad (1.1) \quad \#\text{Sub}(M_k \rightarrow G) = \sum_H a_k(H) \cdot \#\text{Hom}(H \rightarrow G)$$

304 where the sum is over all isomorphism classes of all graphs. By a classic result of
 305 Dalmau and Jonsson [23], computing $\#\text{Hom}(H \rightarrow G)$ is not fixed-parameter tractable
 306 for H of unbounded treewidth, unless ETH fails. Hence, if we could use (1.1) to show
 307 that an FPT algorithm for computing $\#\text{Sub}(M_k \rightarrow G)$ yields an FPT algorithm for
 308 computing $\#\text{Hom}(H \rightarrow G)$ for some H whose treewidth grows with k , we would
 309 conclude that computing $\#\text{Sub}(M_k \rightarrow G)$ is not fixed-parameter tractable unless
 310 ETH fails. This is what [18] indeed prove. The idea is to apply (1.1) not to G ,
 311 but to a set of carefully chosen graphs $\hat{G}_1, \dots, \hat{G}_\ell$ such that the counts $\#\text{Hom}(M_k \rightarrow$

312 $\hat{G}_1), \dots, \#\text{Hom}(M_k \rightarrow \hat{G}_\ell)$ can be used to solve a linear system and infer $\#\text{Hom}(H \rightarrow$
 313 $G)$ for all H appearing on the right-hand side of (1.1).

314 Let us explain this idea in more detail. Suppose we had an oracle for $\#\text{SUB}(\mathcal{M} \rightarrow$
 315 $\mathcal{U})$, so that we could quickly compute $\#\text{Sub}(M_k \rightarrow G)$ for any desired G . Let ℓ be the
 316 size of the support of a_k , which is finite and thus a function of k , and let $\{G_i\}_{i=1, \dots, \ell}$
 317 be a set of graphs such that each G_i has size bounded by a function of k . It is a
 318 well-known fact that, for all graphs H, G, G' ,

$$319 \quad (1.2) \quad \#\text{Hom}(H \rightarrow G \times G') = \#\text{Hom}(H \rightarrow G) \cdot \#\text{Hom}(H \rightarrow G').$$

320 By combining (1.1) and (1.2), for each $i = 1, \dots, \ell$ we obtain

$$321 \quad (1.3) \quad \#\text{Sub}(M_k \rightarrow G \times G_i) = \sum_H a_k(H) \cdot \#\text{Hom}(H \rightarrow G_i) \cdot \#\text{Hom}(H \rightarrow G) = \sum_{\substack{H \\ a_k(H) \neq 0}} b_H^i \cdot X_H,$$

322

323 where $b_H^i := \#\text{Hom}(H \rightarrow G_i)$ and $X_H := a_k(H) \cdot \#\text{Hom}(H \rightarrow G)$. Now, we can
 324 compute $\#\text{Hom}(H \rightarrow G_i)$ in FPT time since $|G_i|$ is bounded by a function of k ,
 325 and we can compute $\#\text{Sub}(M_k \rightarrow G \times G_i)$ using the oracle. Therefore, in FPT
 326 time we can compute a system of ℓ linear equations with the X_H as unknowns. By
 327 applying classical results due to Lovász (see e.g. [42, Chapter 5]), Curticapean et al.
 328 [18] showed that there always exists a choice of the G_i 's such that this system has a
 329 unique solution. Hence, using those G_i 's one can compute $\#\text{Hom}(H \rightarrow G)$ in FPT
 330 time for all H with $a_k(H) \neq 0$. In particular, one can compute $\#\text{Hom}(F_k \rightarrow G)$ where
 331 F_k is any k -edge graph of maximal treewidth, since [18] also showed that $a_k(H) \neq 0$
 332 for all H with $|E(H)| \leq k$. This gives a parameterized reduction from $\#\text{HOM}(\mathcal{F} \rightarrow \mathcal{U})$
 333 to $\#\text{SUB}(\mathcal{M} \rightarrow \mathcal{U})$, where \mathcal{F} is the class of all maximal-treewidth graphs F_k . Since
 334 $\#\text{HOM}(\mathcal{F} \rightarrow \mathcal{U})$ is hard by [23], the reduction establishes hardness of $\#\text{SUB}(\mathcal{M} \rightarrow \mathcal{U})$
 335 as desired.

336 Our main question is whether this strategy can be extended from \mathcal{U} to any mono-
 337 tone somewhere dense class \mathcal{G} . This is not obvious, since the argument above relies
 338 on two crucial ingredients that may be lost when moving from \mathcal{U} to \mathcal{G} :

339 (I.1) We need to find a family of graphs $\hat{\mathcal{F}} = \{\hat{F}_k \mid k \in \mathbb{N}\}$ such that $\#\text{HOM}(\hat{\mathcal{F}} \rightarrow \mathcal{G})$
 340 is hard and, for all $k \in \mathbb{N}$, $a_k(\hat{F}_k) \neq 0$.

341 (I.2) We need to find graphs G_i such that $G \times G_i \in \mathcal{G}$. This is necessary since
 342 the argument performs a reduction to the problem of counting $\#\text{Sub}(M_k \rightarrow$
 343 $G \times G_i)$, and is not straightforward since $G \times G_i$ may not be in \mathcal{G} even when
 344 both G, G_i are.

345 It turns out that both requirements can be satisfied in a systematic way. First, we
 346 study $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ in some carefully chosen vertex-coloured and edge-coloured
 347 version. It is well-known that the coloured version of the problem is equivalent in
 348 complexity (in the FPT sense) to the uncoloured version; so, to make progress, we
 349 may consider the coloured version. Next, coloured graphs come with a canonical
 350 coloured version of the tensor product which satisfies (1.2), so we can hope to apply
 351 interpolation via tensor products in the colorful setting, too. The introduction of
 352 colours in the analysis of parameterised problems is a common tool for streamlining
 353 reductions that are otherwise unnecessarily complicated (see e.g. [20, 51, 26, 30]). The
 354 technical details of the coloured version are not hard, but cumbersome to state; since
 355 here we do not need them, we defer them to Section 2. Let us now give a high-level
 356 explanation of how we achieve (I.1) and (I.2).

357 For (I.1), we let $\hat{\mathcal{F}}$ be the class of all r -subdivisions of a family \mathcal{E} of regular ex-
 358 pander graphs. A simple construction then allows us to reduce $\#\text{HOM}(\mathcal{E} \rightarrow \mathcal{U})$, which
 359 is known to be hard, to $\#\text{HOM}(\hat{\mathcal{F}} \rightarrow \mathcal{U}^r)$, where \mathcal{U}^r is the set of all r -subdivisions of
 360 graphs. As noted above $\mathcal{U}^r \subseteq \mathcal{G}$, hence $\#\text{HOM}(\hat{\mathcal{F}} \rightarrow \mathcal{G})$ is hard. We will show in the
 361 coloured version that for each graph $F_k \in \hat{\mathcal{F}}$ with k edges, $a_k(F_k) \neq 0$ (see the proof
 362 of Lemma 4.6). Thus, (I.1) is satisfied.

363 For (I.2) we construct, for each k , a finite sequence of coloured graphs G_1, G_2, \dots
 364 satisfying the following two conditions: the system of linear equations given by (the
 365 coloured version of) (1.3) has a unique solution, and the coloured tensor product
 366 between each G_i and any coloured graph in \mathcal{U}^r is in \mathcal{G} . Concretely, we choose as G_i
 367 the so-called *fractured graphs* of the r -subdivisions of the expanders in \mathcal{E} . Fractured
 368 graphs are obtained by a splitting operation on a graph and come with a natural
 369 vertex colouring. They have been introduced in recent work on classifying subgraph
 370 counting problems [51] and we describe them in Section 2.1.

371 Together, our resolutions of (I.1) and (I.2) yield a colourful version of the frame-
 372 work of [18] that applies to *any monotone somewhere dense class of host graphs*. As a
 373 consequence we obtain that $\#\text{HOM}(\mathcal{E} \rightarrow \mathcal{U})$, the problem of counting homomorphisms
 374 from expanders in \mathcal{E} to arbitrary hosts graphs, reduces in FPT time to $\#\text{SUB}(\mathcal{M} \rightarrow \mathcal{G})$
 375 whenever \mathcal{G} is monotone and somewhere dense. Since $\#\text{HOM}(\mathcal{E} \rightarrow \mathcal{U})$ is intractable,
 376 this proves the hardness of $\#\text{SUB}(\mathcal{M} \rightarrow \mathcal{G})$ for all monotone and somewhere dense \mathcal{G} ,
 377 as stated in Theorem 1.2. From this result we will then be able to prove our general
 378 classification for $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ (Theorem 1.3) by combining existing results and
 379 Ramsey-type arguments on \mathcal{H} and \mathcal{G} .

380 This concludes our overview for $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$. The proofs for $\#\text{INDSUB}(\mathcal{H} \rightarrow$
 381 $\mathcal{G})$ are similar, but instead of $\#\text{SUB}(\mathcal{M} \rightarrow \mathcal{G})$, they use as a minimal hard case
 382 $\#\text{INDSET}(\mathcal{G})$, the problem of counting k -independent sets in host graphs from \mathcal{G} .

383 **1.3.2. Classifying Homomorphism Counting via Wall Minors.** The proof
 384 of our dichotomy for $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ for monotone \mathcal{H} and \mathcal{G} (Theorem 1.7) requires
 385 us to establish hardness when \mathcal{G} is somewhere dense and $\text{tw}(\mathcal{H}) = \infty$. Recall that our
 386 solution of (I.1) relied on a reduction from (the coloured version of) $\#\text{HOM}(\mathcal{E} \rightarrow \mathcal{U})$
 387 to (the coloured version of) $\#\text{HOM}(\hat{\mathcal{F}} \rightarrow \mathcal{U}^r)$, where \mathcal{E} is a family of regular expander
 388 graphs, $\hat{\mathcal{F}}$ is the class of all r -subdivisions of graphs in \mathcal{E} , and \mathcal{U}^r is the class of r -
 389 subdivisions of all graphs. Since for all monotone somewhere dense classes \mathcal{G} there is
 390 an r such that $\mathcal{U}^r \subseteq \mathcal{G}$, we would be done if we could make sure that every monotone
 391 class of graphs of unbounded treewidth \mathcal{H} contains $\hat{\mathcal{F}}$ as a subset. Unfortunately, this
 392 is not the case. As a trivial example, \mathcal{H} could be the class of all graphs of degree at
 393 most 3 while \mathcal{E} is a family of 4-regular expanders.

394 To circumvent this problem, we use a result of Thomassen [58] to prove that, for
 395 every positive integer r , every monotone class of graphs \mathcal{H} with unbounded treewidth,
 396 and every *wall* $W_{k,k}$, the class \mathcal{H} contains a subdivision of $W_{k,k}$ in which each edge is
 397 subdivided a positive multiple of r times. Now, the crucial property of the class of all
 398 walls $\mathcal{W} := \{W_{k,k} \mid k \in \mathbb{N}\}$ is that $\#\text{HOM}(\mathcal{W} \rightarrow \mathcal{U})$ is intractable by the classification
 399 of Dalmau and Jonsson [23]. Refining our constructions based on subdivided graphs,
 400 we are then able to show that $\#\text{HOM}(\mathcal{W} \rightarrow \mathcal{U})$ reduces to $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ whenever
 401 \mathcal{H} is monotone and of unbounded treewidth, and \mathcal{G} is monotone and somewhere dense.
 402 Theorem 1.7 will then follow as a direct consequence.

403 **2. Preliminaries.** We denote the set of non-negative integers by \mathbb{N}_0 , and the
 404 set of positive integers by \mathbb{N} . Graphs in this work are undirected and without self-
 405 loops unless stated otherwise. A *subdivision* of a graph G is obtained by subdividing

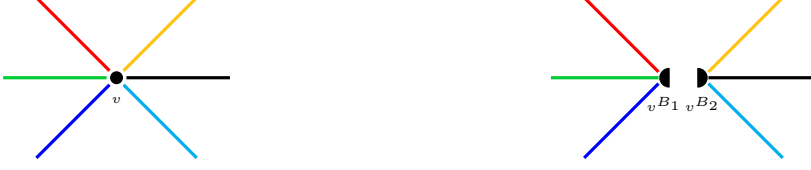


FIG. 1. A fractured graph $Q\#\sigma$ from [51]. Left: a vertex $v \in V(Q)$ with incident edges $E_Q(v) = \{\bullet, \bullet, \bullet, \bullet, \bullet\}$. Right: the splitting of v in $Q\#\sigma$ for a fracture σ where the partition σ_v of $E_Q(v)$ consists of the two blocks $B_1 = \{\bullet, \bullet, \bullet\}$, and $B_2 = \{\bullet, \bullet, \bullet\}$.

each edge of G arbitrarily often. Given a graph G and $r \in \mathbb{N}_0$, we write G^r for the r -subdivision of G , i.e., the graph obtained from G by subdividing each edge r times (so that it becomes a path of $r + 1$ edges). Note that $G^0 = G$. (The graph G^{r-1} is also called the “ r -stretch of G ” in the literature). Given a graph G and a vertex $v \in V(G)$, we write $E_G(v) := \{e \in E(G) \mid v \in e\}$ for the set of edges incident to v . Furthermore, given $A \subseteq E(G)$, we write $G[A]$ for the graph $(V(G), A)$. Given a subset of vertices $S \subseteq V(G)$, we write $G[S]$ for the subgraph of G induced by the vertices in S , that is, $G[S] := (S, \{e \in E(G) \mid e \subseteq S\})$. An “induced subgraph” of G is a subgraph induced by some $S \subseteq V(G)$.

A homomorphism from a graph H to a graph G is a mapping $\varphi : V(H) \rightarrow V(G)$ which is edge-preserving, that is, $\{u, v\} \in E(H)$ implies $\{\varphi(u), \varphi(v)\} \in E(G)$. We write:

- $\text{Hom}(H \rightarrow G)$ for the set of all homomorphisms from H to G ,
- $\text{SurHom}(H \rightarrow G)$ for the set of all surjective homomorphisms from H to G ,
- $\text{Sub}(H \rightarrow G)$ for the set of all subgraphs of G isomorphic to H , and
- $\text{IndSub}(H \rightarrow G)$ for the set of all induced subgraphs of G isomorphic to H .

2.1. Coloured Graphs and Fractures. Let H be a graph. Following standard terminology, we refer to an element of $\text{Hom}(G \rightarrow H)$ as an H -colouring of the graph G . An H -coloured graph is a pair (G, c) where G is a graph and c an H -colouring of G . We say that (G, c) is a surjectively H -coloured graph if $c \in \text{SurHom}(G \rightarrow H)$.

Given two H -coloured graphs (F, c_F) and (G, c_G) , a homomorphism from (F, c_F) to (G, c_G) is a mapping $\varphi \in \text{Hom}(F \rightarrow G)$ such that $c_G(\varphi(v)) = c_F(v)$ for each $v \in V(F)$.⁷ We write $\text{Hom}((F, c_F) \rightarrow (G, c_G))$ for the set of all homomorphisms from (F, c_F) to (G, c_G) .

Following the terminology of [51], we define a fracture of a graph H as a $|V(H)|$ -tuple $\rho = (\rho_v)_{v \in V(H)}$ where ρ_v is a partition of the set $E_H(v)$ of edges of H incident to v . Now, given a fracture ρ of H , we obtain the fractured graph $H\#\rho$ from H by splitting each vertex v according to the partition ρ_v . Formally, the graph $H\#\rho$ contains a vertex v^B for each vertex $v \in V(H)$ and block $B \in \rho_v$, and we make v^B and $u^{B'}$ adjacent if and only if $\{v, u\} \in E(H)$ and $\{u, v\} \in B \cap B'$. An illustration is provided in Figure 1.

The following H -colouring of a fractured graph is used implicitly in [51].

DEFINITION 2.1. Let H be a graph and ρ a fracture of H . We denote by $c_\rho : V(H\#\rho) \rightarrow V(H)$ the function that maps v^B to v for each $v \in V(H)$ and $B \in \rho_v$.

⁷We remark that in previous work [51], homomorphisms between H -coloured graphs are called “colour-preserving” or, if $F = H$, “colour-prescribed”. Since we will work almost exclusively in the coloured setting in this work, we will just speak of homomorphisms and always provide the H -colourings explicitly in our notation.

440 OBSERVATION 2.2. For each H and ρ , c_ρ is an H -colouring of $H\#\rho$.

441 **2.2. Graph Classes, Invariants and Minors.** We use symbols $\mathcal{F}, \mathcal{G}, \mathcal{H}$ to
 442 denote classes of graphs, and we denote by \mathcal{U} be the class of *all* graphs. A graph
 443 invariant is a function $g : \mathcal{U} \rightarrow \mathbb{N}_0$ such that $g(G) = g(H)$ whenever G and H are
 444 isomorphic. An invariant g is *bounded* on a graph family \mathcal{H} if there exists $B \in \mathbb{N}_0$
 445 such that $g(H) \leq B$ for all $H \in \mathcal{H}$, in which case we write $g(\mathcal{H}) < \infty$; otherwise we
 446 say g is unbounded on \mathcal{H} and write $g(\mathcal{H}) = \infty$. Our statements involve the following
 447 invariants.

448 DEFINITION 2.3 (Graph Invariants). For any graph G define:

- 449 • the independence number $\alpha(G)$, i.e., the size of the largest independent set of
 450 G
- 451 • the clique number $\omega(G)$, i.e., the size of the largest complete subgraph of G
- 452 • the biclique number $\beta(G)$, i.e., the largest k such that G contains a k -by- k
 453 biclique as a subgraph, and its induced version, the induced biclique number
 454 $\beta_{\text{ind}}(G)$
- 455 • the matching number $m(G)$, i.e., the size of a maximum matching of G , and
 456 its induced version, the induced matching number $m_{\text{ind}}(G)$

457 We denote by $\text{tw}(G)$ the *treewidth* of a graph G . We omit the definition of treewidth
 458 as we rely on it in a black-box manner; the interested reader can see e.g. Chapter 7
 459 of [22]. For any $k \in \mathbb{N}$ the k -by- k grid graph \boxplus_k , depicted in Figure 2, is defined by
 460 $V(\boxplus_k) = [k]^2$ and $E(\boxplus_k) = \{(i, j), (i', j')\} : i, j, i', j' \in [k], |i - i'| + |j - j'| = 1\}$. It
 461 is well known that $\text{tw}(\boxplus_k) = k$, see [22, Chapter 7.7.1].

462 We make use of the following two consequences of Ramsey's Theorem for an
 463 arbitrary class of graphs \mathcal{H} . The first one is immediate, and the second one was
 464 established by Curticapean and Marx in [20].

465 THEOREM 2.4. If $|\mathcal{H}| = \infty$ then $\max(\alpha(\mathcal{H}), \omega(\mathcal{H})) = \infty$.

466 THEOREM 2.5. If $m(\mathcal{H}) = \infty$ then $\max(\omega(\mathcal{H}), \beta_{\text{ind}}(\mathcal{H}), m_{\text{ind}}(\mathcal{H})) = \infty$.

467 A class of graphs is *hereditary* if it is closed under vertex deletion, and is *monotone*
 468 if it is hereditary and closed under edge deletion. In other words, hereditary classes
 469 are closed under taking induced subgraphs, and monotone classes are closed under
 470 taking subgraphs.

471 To present the different notions of graph minors used in this paper in a unified
 472 way, we start by introducing *contraction models*.

473 DEFINITION 2.6 (Contraction model). A contraction model of a graph H in a
 474 graph G is a partition $\{V_1, \dots, V_k\}$ of $V(G)$ such that $G[V_i]$ is connected for each
 475 $i \in [k]$ and that H is isomorphic to the graph obtained from G by contracting each
 476 $G[V_i]$ into a single vertex (and deleting multiple edges and self-loops).

477 Recall that a graph F is a *minor* of a graph G if F can be obtained from G by
 478 deletion of edges and vertices, and by contraction of edges; equivalently, F is a minor
 479 of G if F is a subgraph of a graph that has a contraction model in G . In this work,
 480 we will also require the subsequent stricter notion of minors.

481 DEFINITION 2.7 (Shallow minor [48]). A graph F is a shallow minor at depth d
 482 of a graph G if F is a subgraph of graph H that has a contraction model $\{V_1, \dots, V_k\}$
 483 in G satisfying the following additional constraint: for each $i \in [k]$ there is a vertex
 484 $x_i \in V_i$ such that each vertex in V_i has distance at most d from x_i . Given a class of
 485 graphs \mathcal{G} , we write $\mathcal{G}\nabla d$ for the set of all shallow minors at depth d of graphs in \mathcal{G} .

486 Observe that the shallow minors at depth 0 of G are exactly the subgraphs of G ,
 487 and the shallow minor of depth $|V(G)|$ are exactly the minors of G . For this reason,
 488 the notion of a shallow minor can be considered an interpolation between subgraphs
 489 and minors. Furthermore, having introduced this notion, we are now able to define
 490 somewhere dense and nowhere dense graph classes.

491 **DEFINITION 2.8** (Somewhere dense graph classes [48]). *A class of graphs \mathcal{G} is*
 492 *somewhere dense if $\omega(\mathcal{G}\nabla d) = \infty$ for some $d \in \mathbb{N}_0$, and is nowhere dense if instead*
 493 *$\omega(\mathcal{G}\nabla d) < \infty$ for all $d \in \mathbb{N}_0$.*

494 We use the following characterisation of monotone somewhere dense graph classes.⁸

495 **LEMMA 2.9** (Remark 2 in [1]). *Let \mathcal{G} be a monotone class of graphs. Then \mathcal{G} is*
 496 *somewhere dense if and only if there exists $r \in \mathbb{N}_0$ such that $G^r \in \mathcal{G}$ for all $G \in \mathcal{U}$.*

497 **2.3. Parameterised and Fine-Grained Complexity.** *A parameterized counting*
 498 *problem is a pair (P, κ) where $P : \{0, 1\}^* \rightarrow \mathbb{N}$ and $\kappa : \{0, 1\}^* \rightarrow \mathbb{N}$ is comput-*
 499 *able. For an instance x of P we call $\kappa(x)$ the *parameter* of x . An algorithm \mathbb{A} is*
 500 *fixed-parameter tractable (FPT) w.r.t. a parameterization κ if there is a computable*
 501 *function f such that \mathbb{A} runs in time $f(\kappa(x)) \cdot |x|^{O(1)}$ on every input x . A parame-*
 502 *terized counting problem (P, κ) is fixed-parameter tractable (FPT) if there is an FPT*
 503 *algorithm (w.r.t. κ) that computes P .*

504 *A parameterized Turing reduction from (P, κ) to (P', κ') is an algorithm \mathbb{A} equipped*
 505 *with oracle access to P' satisfying the following constraints:*

- 506 (A1) \mathbb{A} computes P
- 507 (A2) \mathbb{A} is FPT w.r.t. κ
- 508 (A3) there is a computable function g such that, on input x , each oracle query x'
 509 satisfies that $\kappa'(x') \leq g(\kappa(x))$.

510 We write $(P, \kappa) \leq^{\text{FPT}} (P', \kappa')$ if a parameterized Turing reduction from (P, κ) to
 511 (P', κ') exists.

512 The parameterized counting problem $\#\text{CLIQUE}$ asks, on input a graph G and
 513 $k \in \mathbb{N}$, to compute the number of k -cliques in G ; the parameter is k . As shown by Flum
 514 and Grohe [28], $\#\text{CLIQUE}$ is the canonical complete problem for the parameterized
 515 complexity class $\#\text{W}[1]$. In particular, a parameterized counting problem (P, κ) is
 516 called $\#\text{W}[1]$ -hard if $\#\text{CLIQUE} \leq^{\text{FPT}} (P, \kappa)$. We omit the technical definition of
 517 $\#\text{W}[1]$ via weft-1 circuits (see Chapter 14 of [29]), but we recall that $\#\text{W}[1]$ -hard
 518 problems are not FPT unless standard hardness assumptions fail (see below). We
 519 define the problems studied in this work. As usual \mathcal{H} and \mathcal{G} denote classes of graphs.

520 **DEFINITION 2.10.** $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G}), \#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G}), \#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ ask, given
 521 $H \in \mathcal{H}$ and $G \in \mathcal{G}$, to compute respectively $\#\text{Hom}(H \rightarrow G)$, $\#\text{Sub}(H \rightarrow G)$, and
 522 $\#\text{IndSub}(H \rightarrow G)$. The parameter is $|H|$.

523 For example, $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G}) = \#\text{CLIQUE}$ when \mathcal{H} is the class of all complete graphs
 524 and \mathcal{G} the class of all graphs. The following result follows immediately from an
 525 algorithm for counting answers to Boolean queries in nowhere dense graphs due to
 526 Nešetřil and Ossona de Mendez [49].

527 **THEOREM 2.11** (Theorem 18.9 in [49]). *If \mathcal{G} is nowhere dense then $\#\text{HOM}(\mathcal{H} \rightarrow$*
 528 *$\mathcal{G}), \#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G}),$ and $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ are fixed-parameter tractable and can be*
 529 *solved in time $f(|H|) \cdot |V(G)|^{1+o(1)}$ for some computable function f .*

⁸It is non-trivial to pinpoint the first statement of Lemma 2.9 in the literature: Dvorák et al. [27] attribute it to Nešetřil and de Mendez [48], who provide an implicit proof. The first explicit statement is, to the best of our knowledge, due to Adler and Adler [1].

530 In an intermediate step towards our classifications, we will rely on a coloured
531 version of homomorphism counting.

532 DEFINITION 2.12. $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{G})$ asks, given $H \in \mathcal{H}$ and a surjectively⁹ H -
533 coloured graph (G, c) with $G \in \mathcal{G}$, to compute $\#\text{Hom}((H, \text{id}_H) \rightarrow (G, c))$, where id_H
534 denotes the identity on $V(H)$. The parameter is $|H|$.

535 It is well known that $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{U})$ reduces to the uncoloured version via
536 inclusion-exclusion. The same holds for $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{G})$, too, if \mathcal{G} is monotone.
537 Formally:

538 LEMMA 2.13 (see e.g. Lemma 2.49 in [53]). *If \mathcal{G} is monotone then $\#\text{CP-HOM}(\mathcal{H} \rightarrow$
539 $\mathcal{G}) \leq^{\text{FPT}} \#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$. Moreover, on input $H \in \mathcal{H}$ and (G, c) with $G \in \mathcal{G}$, every
540 oracle query (H', G') in the reduction satisfies $H' = H$ and $G' \subseteq G$.*

541 An implicit consequence of the parameterized complexity classification for counting
542 homomorphisms due to Dalmau and Jonsson [23] establishes the following hardness
543 result for $\#\text{CP-HOM}$; an explicit argument can be found e.g. in Chapter 2 in [53].

544 THEOREM 2.14 ([23]). *If \mathcal{H} is recursively enumerable and $\text{tw}(\mathcal{H}) = \infty$ then
545 $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{U})$ is $\#\text{W}[1]$ -hard.*

546 Finally, all running-time lower bounds in this paper are conditional on ETH:

547 DEFINITION 2.15 ([38]). *The Exponential Time Hypothesis (ETH) asserts that
548 3-SAT cannot be solved in time $\exp(o(n))$ where n is the number of variables of the
549 input formula.*

550 Chen et al. [12, 13] showed that there is no function f such that $\#\text{CLIQUE}$ can be
551 solved in time $f(k) \cdot |G|^{o(k)}$ unless ETH fails. This in particular implies that $\#\text{W}[1]$ -
552 hard problems are not FPT unless ETH fails. Marx [43] strengthened Theorem 2.14
553 into:¹⁰

554 THEOREM 2.16 ([43]). *If \mathcal{H} is recursively enumerable and $\text{tw}(\mathcal{H}) = \infty$ then
555 $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{U})$ cannot be solved in time $f(|H|) \cdot |G|^{o(\frac{\text{tw}(H)}{\log \text{tw}(H)})}$ for any function f ,
556 unless ETH fails.*

557 The question of whether the $(\log \text{tw}(H))^{-1}$ factor in the above lower bound can be
558 omitted can be considered the counting version of the open problem ‘‘Can you beat
559 treewidth?’’ [43, 44].

560 **3. Counting Homomorphisms.** This section is devoted to the proof of our di-
561 chotomy theorem for $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$, Theorem 1.7. We start by showing a reduction
562 from $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{U})$ to counting colour-prescribed homomorphisms between sub-
563 divided graphs. While the proof is straightforward, the reduction will turn out useful
564 for the more involved cases of $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ and $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$. Theorem 1.7
565 will be an immediate consequence of Theorem 2.11 and Theorem 3.6 below.

566 To begin with, let $c \in \text{SurHom}(G \rightarrow H)$ and let $r \in \mathbb{N}_0$. Define the following
567 canonical homomorphism c^r from G^r to H^r . For each $u \in V(G)$, set $c^r(u) := c(u)$.
568 For any edge $e = \{u_1, u_2\} \in E(G)$, let $u_1, w_1, \dots, w_r, u_2$ be the corresponding path in
569 G^r . Let $e' = \{v_1, v_2\} = \{c(u_1), c(u_2)\}$ — note that $e' \in E(H)$ as $c \in \text{Hom}(G \rightarrow H)$

⁹In previous works (e.g. in [51]), the definition of $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{G})$ did not require the H -
colouring to be surjective. However, one can always assume surjectivity, since $\#\text{Hom}((H, \text{id}_H) \rightarrow$
 $(G, c)) = 0$ if c is not surjective. We decided to make the surjectivity condition explicit in this work.

¹⁰More precisely, Marx established the bound for the so-called partitioned subgraph problem.
However, as shown in [55], the lower bound immediately translates to $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{U})$.

570 — and let $v_1, x_1, \dots, x_r, v_2$ be the corresponding path in H^r . Then, set $c^r(w_i) := x_i$
571 for each $i \in \{1, \dots, r\}$. It is easy to see that c^r is a surjective H^r -colouring of G^r .
572 Furthermore:

573 LEMMA 3.1. *For every surjectively H -coloured graph (G, c) and every $r \in \mathbb{N}_0$,*

$$574 \quad (3.1) \quad \#\text{Hom}((H, \text{id}_H) \rightarrow (G, c)) = \#\text{Hom}((H^r, \text{id}_{H^r}) \rightarrow (G^r, c^r))$$

575 where id_H and id_{H^r} are the identities on respectively $V(H)$ and $V(H^r)$.

576 *Proof.* We define a bijection $b : \text{Hom}((H, \text{id}_H) \rightarrow (G, c)) \rightarrow \text{Hom}((H^r, \text{id}_{H^r}) \rightarrow (G^r, c^r))$.
577 Let $\varphi \in \text{Hom}((H, \text{id}_H) \rightarrow (G, c))$. For every $v \in V(H)$ let $b(\varphi)(v) = \varphi(v)$.
578 For every $\{v_1, v_2\} \in E(H)$ and every $i \in [r]$, if $u_1 = \varphi(v_1)$ and $u_2 = \varphi(v_2)$, and
579 if x_i and w_i are the i -th vertices on the paths respectively between v_1 and v_2 in
580 H^r and between u_1 and u_2 in G^r , then let $b(\varphi)(x_i) = w_i$. It is easy to see that
581 $b(\varphi) \in \text{Hom}((H^r, \text{id}_{H^r}) \rightarrow (G^r, c^r))$ and that b is injective. To see that b is surjective
582 as well, note that for every $\varphi^r \in \text{Hom}((H^r, \text{id}_{H^r}) \rightarrow (G^r, c^r))$ its restriction $\varphi^r|_{V(H)}$
583 to $V(H)$ satisfies $\varphi^r|_{V(H)} \in \text{Hom}((H, \text{id}_H) \rightarrow (G, c))$ and $b(\varphi^r|_{V(H)}) = \varphi^r$. \square

584 **3.1. Warm-up: Minor-closed Pattern Classes.** Using the characterisation
585 of somewhere dense graph classes in Lemma 2.9, and known lower bounds for counting
586 homomorphisms from grid graphs, we obtain as an easy consequence the following
587 complexity dichotomy:
588

589 THEOREM 3.2. *Let \mathcal{H} be a minor-closed class of graphs and let \mathcal{G} be a monotone
590 and somewhere dense class of graphs.*

- 591 1. *If $\text{tw}(\mathcal{H}) < \infty$ then $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G}) \in \text{P}$. Moreover, if a tree decomposition
592 of H of width t is given, then $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ can be solved in time $|H|^{O(1)} \cdot$
593 $|V(G)|^{t+1}$.*
- 594 2. *If $\text{tw}(\mathcal{H}) = \infty$, then $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ is $\#\text{W}[1]$ -hard and, assuming ETH,
595 cannot be solved in time $f(|H|) \cdot |G|^{o(\text{tw}(H))}$ for any function f .*

596 *Proof.* The tractability result is well known [24, 23], so we only need to prove the
597 hardness part. Recall that \boxplus_k denotes the k -by- k grid; see Figure 2 for a depiction of
598 \boxplus_4 . Let $\boxplus := \{\boxplus_k \mid k \in \mathbb{N}\}$. It is known that $\#\text{CP-HOM}(\boxplus \rightarrow \mathcal{U})$ is $\#\text{W}[1]$ -hard and,
599 unless ETH fails, cannot be solved in time $f(k) \cdot |G|^{o(k)}$ for any function f (see [16,
600 Lemma 1.13 and 5.7] or [53, Lemma 2.45]). As $\text{tw}(\boxplus_k) = k$, the lower bound above
601 can be written as $f(k) \cdot |G|^{o(\text{tw}(\boxplus_k))}$.

602 Let $(\boxplus_k, (G, c))$ be the input to $\#\text{CP-HOM}(\boxplus \rightarrow \mathcal{U})$. Since \mathcal{G} is somewhere dense
603 and monotone, by Lemma 2.9 there is $r \in \mathbb{N}_0$ such that \mathcal{G} contains the r -subdivision
604 of every graph and thus, in particular, G^r . Moreover, since $\text{tw}(\mathcal{H}) = \infty$ and \mathcal{H} is
605 minor-closed, by the Excluded-Grid Theorem [52] \mathcal{H} contains every planar graph and
606 thus in particular \boxplus_k^r . Clearly, \boxplus_k^r, G^r and c^r can be computed in polynomial time.
607 Moreover, by Lemma 3.1,

$$608 \quad \#\text{Hom}((\boxplus_k, \text{id}_{\boxplus_k}) \rightarrow (G, c)) = \#\text{Hom}((\boxplus_k^r, \text{id}_{\boxplus_k^r}) \rightarrow (G^r, c^r)).$$

609 Hence $\#\text{CP-HOM}(\boxplus \rightarrow \mathcal{U}) \leq^{\text{FPT}} \#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{G})$. Since $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{G}) \leq^{\text{FPT}}$
610 $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ by Lemma 2.13, we conclude that $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ is $\#\text{W}[1]$ -hard.
611 For the conditional lower bound, observe that both reductions used above preserve
612 the treewidth of the pattern (the first because treewidth is invariant under edge sub-
613 division,¹¹ the second by Lemma 2.13). \square

¹¹For example, this invariance is in Exercises 7.7 and 7.13 in [22].

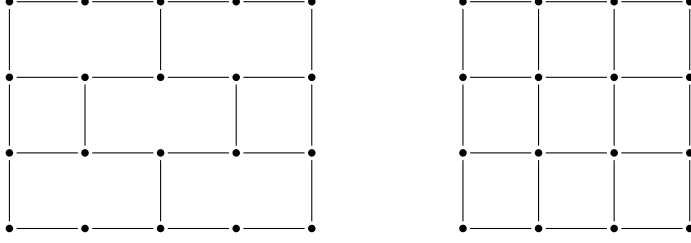


FIG. 2. The wall $W_{4,5}$ (left) and the grid \mathbb{H}_4 (right).

614 **3.2. Monotone Pattern Classes.** The strengthening of Theorem 3.2 to mono-
615 tone pattern classes can be done by reduction from counting homomorphisms from a
616 class of well-known graphs called *walls*.

617 **DEFINITION 3.3 (Walls).** Let $k, \ell \in \mathbb{N}$. The wall of height k and length ℓ , denoted
618 by $W_{k,\ell}$, is the graph whose vertex set is $\{v_{i,j} : 1 \leq i \leq k, 1 \leq j \leq \ell\}$ and whose edge
619 set contains:

- 620 • $\{v_{i,j}, v_{i,j+1}\}$ for all $1 \leq i \leq k$ and $1 \leq j \leq \ell - 1$,
- 621 • $\{v_{i,1}, v_{i+1,1}\}$ and $\{v_{i,\ell}, v_{i+1,\ell}\}$ for all $1 \leq i \leq k - 1$
- 622 • $\{v_{i,j}, v_{i+1,j}\}$ for all $1 \leq i \leq k - 1$ and $1 \leq j \leq \ell$ such that $i + j$ is even.

623 Figure 2 depicts $W_{4,5}$ as an example. We let $\mathcal{W} := \{W_{k,k} \mid k \in \mathbb{N}\}$ be the class of all
624 walls.

625 The following structural property of large walls is due to Thomassen.¹²

626 **LEMMA 3.4 (Proposition 3.2 in [58]).** For every $k, r \in \mathbb{N}$, there exists $h(k, r) \in \mathbb{N}$
627 such that every subdivision of $W_{h(k,r), h(k,r)}$ contains as a subgraph a subdivision of
628 $W_{k,k}$ in which each edge is subdivided a (positive) multiple of r times.

629 The final ingredient of our proof for the classification of monotone pattern classes
630 is given by Lemma 3.5, which is an immediate consequence of Lemma 2.9.

631 **LEMMA 3.5.** Let \mathcal{G} be a monotone and somewhere dense class of graphs. There
632 exists $r \in \mathbb{N}_0$ such that the following holds. Let G be any graph and let G' be any
633 graph obtained from G by subdividing each edge a (positive) multiple of r times. Then
634 G' is contained in \mathcal{G} .

635 *Proof.* We show that the claim holds for the $r \in \mathbb{N}_0$ given by Lemma 2.9. For this
636 r , Lemma 2.9 guarantees that for every graph H , $H' \in \mathcal{G}$. Now let G be any graph
637 and label its edges e_1, \dots, e_m . Let G' be any graph obtained from G by subdividing
638 each edge a (positive) multiple of r times. Then there exist $d_1, \dots, d_m \in \mathbb{N}$ such that,
639 for each $i \in [m]$, the edge e_i is subdivided $d_i r$ times. Now let \hat{G} be the graph obtained
640 from G by subdividing, for each $i \in [m]$, the edge e_i just d_i times. It is immediate
641 that $G' = \hat{G}^r$. Thus, by Lemma 2.9 and our choice of r , we have that $G' \in \mathcal{G}$. \square

642 We are now ready to establish the main result of this section.

643 **THEOREM 3.6.** Let \mathcal{H} be a monotone class of graphs and let \mathcal{G} be a monotone and
644 somewhere dense class of graphs.

- 645 1. If $\text{tw}(\mathcal{H}) < \infty$ then $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G}) \in \mathcal{P}$. Moreover, if a tree decomposition
646 of H of width t is given, then $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ can be solved in time $|H|^{O(1)} \cdot$
647 $|V(G)|^{t+1}$.

¹²Note that walls are called grids in [58].

648 2. If $\text{tw}(\mathcal{H}) = \infty$, then $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ is $\#\text{W}[1]$ -hard and, assuming ETH,
649 cannot be solved in time $f(k) \cdot |G|^{o(\text{tw}(H))}$ for any function f .

650 *Proof.* The tractability result is well known [24, 23], so we only need to prove
651 point 2. To this end, we will reduce from $\#\text{CP-HOM}(\mathcal{W} \rightarrow \mathcal{U})$. Walls clearly have
652 grid minors of linear size, that is, there is a function $h \in \Theta(k)$ such that $W_{k,k}$ con-
653 tains $\boxplus_{h(k)}$ as a minor. Furthermore, it is well-known that $\#\text{CP-HOM}$ is minor-
654 monotone (see e.g. [16, Lemma 5.8] or [53, Lemma 2.47]), hence $\#\text{CP-HOM}(\boxplus \rightarrow$
655 $\mathcal{U}) \leq^{\text{FPT}} \#\text{CP-HOM}(\mathcal{W} \rightarrow \mathcal{U})$. Moreover the reduction is tight, in the sense that the
656 lower bound for $\#\text{CP-HOM}(\boxplus \rightarrow \mathcal{U})$ shown in the proof of Theorem 3.2 transfers to
657 $\#\text{CP-HOM}(\mathcal{W} \rightarrow \mathcal{U})$; hence $\#\text{CP-HOM}(\mathcal{W} \rightarrow \mathcal{U})$ is $\#\text{W}[1]$ -hard and, assuming ETH,
658 it cannot be solved in time $f(k) \cdot |G|^{o(\text{tw}(W_{k,k}))}$ for any function f .

659 Let us now construct the reduction $\#\text{CP-HOM}(\mathcal{W} \rightarrow \mathcal{U}) \leq^{\text{FPT}} \#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{G})$.
660 Let $r \in \mathbb{N}_0$ as given by Lemma 3.5. We use the fact that $\text{tw}(\mathcal{H}) = \infty$ implies that
661 \mathcal{H} contains as minors all planar graphs; that is, for every planar graph F there is a
662 graph $H \in \mathcal{H}$ such that F is a minor of H [52]. In particular, \mathcal{H} contains all walls
663 $W_{k,k}$ as minors. A graph J is said to be a “topological minor” of a graph H if there
664 is a subdivision of J that is isomorphic to a subgraph of H . Since walls have degree
665 at most 3, the fact that \mathcal{H} contains all walls as minors implies that it also contains
666 all walls as topological minors (see e.g. [25, Proposition 1.7.3]).

667 Now let $W_{k,k}$ and (G, c) be an input instance of $\#\text{CP-HOM}(\mathcal{W} \rightarrow \mathcal{U})$. Let
668 e_1, \dots, e_ℓ be the edges of $W_{k,k}$ in arbitrary order. By Lemma 3.4, every subdivi-
669 sion of $W_{h(k,r), h(k,r)}$ contains as a subgraph a subdivision of $W_{k,k}$ in which each edge
670 is subdivided a (positive) multiple of r times. Since \mathcal{H} contains $W_{k,k}$ as a topological
671 minor, there is a subdivision of $W_{k,k}$ that is isomorphic to a subgraph W' of a graph
672 in \mathcal{H} . Since \mathcal{H} is monotone, there are $W' \in \mathcal{H}$ and $d_1, \dots, d_\ell \in \mathbb{N}_0$ such that W'
673 is obtained from $W_{k,k}$ by subdividing e_i precisely $d_i r$ times for each $i \in [\ell]$.

674 We will now construct from (G, c) a graph G' and a surjective homomorphism c'
675 from G' to W' . For each edge $e = \{u, v\}$ of G we proceed as follows. Since $c \in$
676 $\text{Hom}(G \rightarrow W_{k,k})$, then $\{c(u), c(v)\} = e_i$ for some $i \in [\ell]$. By the definition of W' , e_i
677 was replaced by a path $c(u), x_1, \dots, x_{d_i r}, c(v)$. Hence, we replace the edge e in G
678 by a path $u, w_1, w_2, \dots, w_{d_i r}, v$, where the w_j are fresh vertices. Furthermore, we extend
679 the colouring c to the colouring c' by setting $c'(w_j) := x_j$ for each $j \in [d_i r]$. Since c
680 is surjective, so is c' . Also,

$$681 \quad \#\text{Hom}((W_{k,k}, \text{id}_{W_{k,k}}) \rightarrow G, c) = \#\text{Hom}((W', \text{id}_{W'}) \rightarrow (G', c')).$$

682 By querying the oracle for $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{G})$ on the instance $((W', \text{id}_{W'}), (G', c'))$
683 we can thus conclude our reduction. This immediately implies $\#\text{W}[1]$ -hardness of
684 $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{G})$. For the conditional lower bound, we observe that W' has the
685 same treewidth as $W_{k,k}$ since it is a subdivision of $W_{k,k}$, and that the size of (G', c') is
686 clearly bounded by $f(k) \cdot |G|^{O(1)}$ — note that the f depends on \mathcal{H} which is, however,
687 fixed. A reduction to the uncoloured version via Lemma 2.13 completes the proof. \square

688 Theorem 1.7 follows immediately from Theorem 2.11 and Theorem 3.6. We conclude
689 with a remark.

690 **REMARK 3.7.** *A strengthening of Theorem 3.6 to hereditary pattern classes \mathcal{H} is*
691 *not possible. Suppose for instance that \mathcal{H} contains all complete graphs and \mathcal{G} is the*
692 *class of all bipartite graphs. Although \mathcal{H} is hereditary and of unbounded treewidth, and*
693 *\mathcal{G} is monotone and somewhere dense, it is easy to see that $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ is trivial,*
694 *since we can always output zero if $H \in \mathcal{H}$ has at least 3 vertices. When it comes*

695 to a sufficient and necessary condition for tractability in case of hereditary classes
696 of patterns, we conjecture that induced grid minor size might be the right candidate.
697 However, even for very special cases, such as classes of degenerate host graphs (which
698 are somewhere dense and monotone), it is still open whether induced grid minor size
699 is the correct answer [9]. Thus, we leave the classification for hereditary classes of
700 patterns as an open problem for further research.

701 **4. Counting Subgraphs.** This section is devoted to the proofs of Theorem 1.2,
702 Theorem 1.4, and Theorem 1.3. We begin in Section 4.1 by analysing the problem of
703 counting k -matchings in somewhere dense host graphs, and proving Theorem 1.2; this
704 is the most technical part. We then move on to prove Theorem 1.4 and Theorem 1.3
705 in Section 4.2.

706 **4.1. Counting Matchings: Proof of Theorem 1.2.** A k -matching in a graph
707 G is a set $M \subseteq E(G)$ with $|M| = k$ and $e_1 \cap e_2 = \emptyset$ for all $e_1 \neq e_2$ in M . In other
708 words, a k -matching in G is a set of k pairwise non-incident edges of G . Given a
709 class of graphs \mathcal{G} , the problem $\#\text{MATCH}(\mathcal{G})$ asks, on input $k \in \mathbb{N}$ and a graph $G \in \mathcal{G}$,
710 to compute the number of k -matchings in G ; the parameter is k . We remark that
711 $\#\text{MATCH}(\mathcal{G}) = \#\text{SUB}(\mathcal{M} \rightarrow \mathcal{G})$ where \mathcal{M} is the set of all 1-regular graphs. The
712 goal of this section is to prove that $\#\text{MATCH}(\mathcal{G})$ is hard whenever \mathcal{G} is monotone and
713 somewhere dense, i.e., the hardness part of Theorem 1.2.

714 Before moving on, let us pin down some definitions and basic facts. Our analysis
715 relies on the following ‘‘coloured’’ version of the graph tensor product, as in [51]:

716 **DEFINITION 4.1.** *Let H be a graph, and let (G_1, c_1) and (G_2, c_2) be H -coloured*
717 *graphs. The tensor product $(G_1, c_1) \times (G_2, c_2)$ is the H -coloured graph (\hat{G}, \hat{c}) defined*
718 *by:*

- 719 (T1) $V(\hat{G}) = \{(v_1, v_2) \in V(G_1) \times V(G_2) \mid c_1(v_1) = c_2(v_2)\}$.
720 (T2) $\{(u_1, u_2), (v_1, v_2)\} \in E(\hat{G})$ if and only if $\{u_1, v_1\} \in E(G_1)$ and $\{u_2, v_2\} \in$
721 $E(G_2)$.
722 (T3) $\hat{c}(v_1, v_2) = c_1(v_1)$ (equivalently by (T1), $\hat{c}(v_1, v_2) = c_2(v_2)$) for all $(v_1, v_2) \in$
723 $V(\hat{G})$.

724 The crucial property of the tensor product is given by:¹³

725 **LEMMA 4.2** ([51]). *If H is a graph and $(F, c_F), (G_1, c_1), (G_2, c_2)$ are H -coloured*
726 *graphs, then*

$$727 \#\text{Hom}((F, c_F) \rightarrow (G_1, c_1) \times (G_2, c_2)) = \#\text{Hom}((F, c_F) \rightarrow (G_1, c_1)) \cdot \#\text{Hom}((F, c_F) \rightarrow (G_2, c_2)).$$

728 The final ingredient we need is the non-singularity of a certain matrix whose
729 entries count homomorphisms between fractured graphs. Formally, let H be a graph.
730 The square matrix M_H has its rows and columns indexed by the fractures of H , and
731 its entries satisfy:

$$732 (4.1) \quad M_H[\rho, \sigma] := \#\text{Hom}((H \# \rho, c_\rho) \rightarrow (H \# \sigma, c_\sigma)),$$

733 where c_ρ and c_σ are the canonical H -colourings of the fractured graphs $H \# \rho$ and $H \# \sigma$
734 (see Definition 2.1 and Observation 2.2). By ordering the columns and rows of M_H
735 along a certain lattice, the following property was established in previous work.¹⁴

¹³Proofs of Lemma 4.2 and Lemma 4.3 can also be found in Section 3.1 in an earlier version [54] of [51].

¹⁴See Footnote 13.

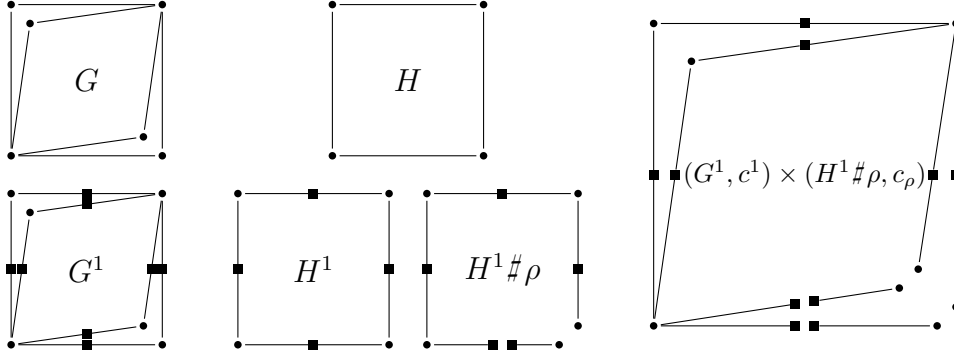


FIG. 3. the tensor product of the H^1 -coloured graphs (G^1, c^1) and $(H^1 \# \rho, c_\rho)$.

736 LEMMA 4.3 ([51]). For each graph H , the matrix M_H is nonsingular.

737 If \mathcal{G} is closed under uncoloured tensor products¹⁵, then the hardness result can
 738 be achieved by applying the reduction of [18] verbatim. However, that reduction fails
 739 if \mathcal{G} is not closed under uncoloured tensor products, and this closure property is very
 740 restrictive. Consider for example the class \mathcal{G} of square-free graphs, i.e., graphs that
 741 do not contain the 4-cycle C_4 as a subgraph. Then \mathcal{G} is clearly monotone and, since it
 742 contains the 3-subdivision of every graph, it is also somewhere dense by Lemma 2.9.
 743 However, \mathcal{G} is not closed under (uncoloured) tensor products: the path on 2 edges P_2
 744 is in \mathcal{G} , but $P_2 \times P_2 \notin \mathcal{G}$ since it contains a C_4 .

745 The main insight of this section is a weakened closure property for monotone and
 746 somewhere dense graph classes, established in the lemma below. Combined with the
 747 characterisation of somewhere dense graph classes via r -subdivisions (Lemma 2.9),
 748 this property implies that any monotone and somewhere dense class is closed under
 749 tensor products of subdivisions of coloured graphs.

750 LEMMA 4.4. Let $r \in \mathbb{N}_0$, let H be a graph without isolated vertices, and let (G, c)
 751 be an H -coloured graph on n vertices, and let ρ be a fracture of H^r . Then $(G^r, c^r) \times$
 752 $(H^r \# \rho, c_\rho)$ is a subgraph of the r -subdivision of a complete graph of order $O(kn)$,
 753 where $k = |E(H)|$ and the constants in the $O()$ notation depend only on r .

754 *Proof.* Let $T = (G^r, c^r) \times (H^r \# \rho, c_\rho)$; see Figure 3 for an example. The claim
 755 follows from Claims 1, 2, and 3 below, with Claim 3 applied to $F = T$.

756 **Claim 1.** $|V(T)| = O(kn)$. Straightforward since G^r is a subgraph of K_n^r .

757 **Claim 2:** if x and y are distinct vertices of T of degree at least 3, then the length of
 758 any simple path from x to y is a multiple of $r + 1$.

759 To prove this, recall that T is H^r -coloured by \hat{c} from Definition 4.1, and that
 760 $V(H^r)$ can be partitioned into $V(H)$ and a set S of kr fresh subdivision vertices. Let
 761 (u, v) be a vertex of T such that $\hat{c}(u, v) = s \notin V(H)$, that is, (u, v) is coloured with a
 762 subdivision vertex s . We show that (u, v) has degree at most 2 in T . Let s_1 and s_2
 763 be the two neighbours of s in H^r . By the construction of (G^r, c^r) , u has exactly two
 764 neighbours in G^r , say u_1 and u_2 . Furthermore, $c^r(u_1) = s_1$ and $c^r(u_2) = s_2$. Since s
 765 has degree 2 in H^r , there are only two cases for ρ_s .

¹⁵The adjacency matrix of the tensor product of two uncoloured graphs G and F is the Kronecker product of the adjacency matrices of G and F .

766 • Case 1: $\rho_s = \{B\}$ where $B = \{\{s, s_1\}, \{s, s_2\}\}$. In this case s^B is the only ver-
767 tex of $H\#\rho$ that is coloured by c_ρ with s . Since $\hat{c}(u, v) = s$ implies $c_\rho(v) = s$,
768 we conclude that $v = s^B$. Hence (u, v) has exactly two neighbours in T ,
769 $(u_1, s_1^{B_1})$ and $(u_2, s_2^{B_2})$, where B_1 and B_2 are the blocks of ρ_{s_1} and ρ_{s_2} con-
770 taining respectively $\{s, s_1\}$ and $\{s, s_2\}$.
771 • Case 2: $\rho_s = \{B, B'\}$ where $B = \{\{s, s_1\}\}$ and $B' = \{\{s, s_2\}\}$. In this case
772 s^B and $s^{B'}$ are the only two vertices of $H\#\rho$ that are coloured by c_ρ with s .
773 Since $\hat{c}(u, v) = s$ implies $c_\rho(v) = s$, we conclude that $v \in \{s^B, s^{B'}\}$. Assume
774 that $v = s^B$; the other case is symmetric. Then the only neighbour of (u, v)
775 in T is $(u_1, s_1^{B_1})$, where B_1 is the block of ρ_{s_1} that contains the edge $\{s, s_1\}$.
776 We conclude that the only vertices (u, v) of degree at least 3 in T satisfy $\hat{c}(u, v) \in$
777 $V(H)$, implying that $c^r(u) \in V(H)$ and thus, by the definition of c^r , that $u \in V(G)$,
778 hence u is not a subdivision vertex. The claim follows since the length of every simple
779 path between two non-subdivision vertices u_1 and u_2 in G^r is a multiple of $(r + 1)$,
780 and since T can be obtained from (G^r, c^r) by splitting vertices.

781 **Claim 3:** if F is a graph where the length of any simple path between two vertices
782 of degree at least 3 is a multiple of $(r + 1)$, then F is a subgraph of the r -subdivision
783 of a complete graph of order $O(|V(F)|)$.

784 Note first that we can deal with each connected component of F separately.
785 Furthermore, the claim is clearly true if F is just a path (of any length). For what
786 follows we can hence assume that F is connected and not isomorphic to a path. We
787 say that a path P in F is *extendable* if its internal vertices have degree 2, one endpoint
788 s_P (the “startpoint”) has degree 1, and the other endpoint has degree at least 3. If P
789 has length ℓ_P , then its *extension length* is the smallest $\ell'_P \in \mathbb{N}_0$ such that $\ell_P + \ell'_P$ is a
790 multiple of $r + 1$. Let F' be the graph formed from F by considering every extendable
791 path P and adding a new length- ℓ'_P path from s_P (adding ℓ'_P fresh vertices to make
792 up this path). Observe that, for every pair of non-isolated vertices u' and v' of F' ,
793 if both u' and v' have degree not equal to 2, then the length of every simple path
794 from u' to v' in F' is a multiple of $(r + 1)$. Therefore F' is a subgraph of the r -
795 subdivision of a complete graph of order at most $O(|V(F')|) = O(|V(F)|)$, where the
796 constants depend only on r . Moreover F is by construction a subgraph of F' , which
797 this concludes the proof of the claim. \square

798 To establish the hardness of $\#\text{MATCH}(\mathcal{G})$, we first consider an edge-coloured ver-
799 sion. Let G be a graph and $k \in \mathbb{N}$. A k -coloring of $E(G)$ is a map $c : E(G) \rightarrow$
800 $\{1, \dots, k\}$. A matching $M \subseteq E(G)$ is *edge-colorful* under if for every colour in
801 $\{1, \dots, k\}$ there is precisely one element of M with that colour.

802 **DEFINITION 4.5** ($\#\text{COLMATCH}(\mathcal{G})$). *Let \mathcal{G} be a class of graphs. The problem*
803 *$\#\text{COLMATCH}(\mathcal{G})$ asks, on input $k \in \mathbb{N}$, a graph $G \in \mathcal{G}$, and a k -coloring c of $E(G)$, to*
804 *compute the number of edge-colorful k -matchings in G . The problem is parameterised*
805 *by k .*

806 **LEMMA 4.6.** *Let \mathcal{G} be a monotone somewhere dense class of graphs. Then the*
807 *problem $\#\text{COLMATCH}(\mathcal{G})$ is $\#\text{W}[1]$ -hard and, assuming ETH , cannot be solved in*
808 *time $f(k) \cdot |G|^{\alpha(k/\log k)}$ for any function f .*

809 *Proof.* Let \mathcal{H} be a class of 3-regular expander graphs. Both the treewidth and
810 the number of edges of the elements of \mathcal{H} grow linearly in the number of vertices; that
811 is, $|E(H)| \in \Theta(|V(H)|)$ and $\text{tw}(H) \in \Theta(|V(H)|)$ for all $H \in \mathcal{H}$ (see, e.g., [37]). Hence
812 theorems 2.14 and 2.16 imply that $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{U})$ is $\#\text{W}[1]$ -hard and, assuming
813 ETH , cannot be solved in time $f(|H|) \cdot |G|^{\alpha(|H|/\log |H|)}$ for any function f . We will

814 now show that $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{U}) \leq^{\text{FPT}} \#\text{COLMATCH}(\mathcal{G})$.

815 Let $H \in \mathcal{H}$ and (G, c) be the input of $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{U})$. By Lemma 2.9, there is
 816 $r \in \mathbb{N}_0$ such that $G^r \in \mathcal{G}$ for all $G \in \mathcal{U}$. Construct then H^r and (G^r, c^r) , which clearly
 817 takes polynomial time. Let $k = |E(H^r)|$; clearly $k \in O(|H|)$ where the constants
 818 depend only on r . Now, by Lemma 3.1,

$$819 \quad \#\text{Hom}((H, \text{id}_H) \rightarrow (G, c)) = \#\text{Hom}((H^r, \text{id}_{H^r}) \rightarrow (G^r, c^r)).$$

820 Next, we view surjectively H^r -coloured graphs (\tilde{G}, \tilde{c}) also as edge-coloured graphs
 821 where every edge $e = \{u, v\}$ is mapped to the colour $\{\tilde{c}(u), \tilde{c}(v)\}$. This allows us to
 822 invoke the results of [51] and deduce what follows.¹⁶

823 First, there is a unique function a from fractures of H^r to rationals such that, for
 824 every surjectively H^r -coloured graph (\tilde{G}, \tilde{c}) , the number of edge-colourful k -matchings
 825 of (\tilde{G}, \tilde{c}) is:

$$826 \quad (4.2) \quad \sum_{\rho} a(\rho) \cdot \#\text{Hom}((H^r \# \rho, c_{\rho}) \rightarrow (\tilde{G}, \tilde{c})),$$

827 where the sum is over all fractures of H^r . Additionally, a satisfies:

$$828 \quad (4.3) \quad a(\top) = \prod_{v \in V(H^r)} (-1)^{\deg(v)-1} \cdot (\deg(v) - 1)!,$$

829 where \top is the coarsest fracture, that is, for each $v \in V(H^r)$ the partition \top_v only
 830 contains a singleton block (and therefore $H^r \# \top = H^r$). In particular, it is easy to see
 831 that

$$832 \quad a(\top) = \pm 2^{|V(H^r)|} \neq 0.$$

833 Now let σ be a fracture of H^r . Considering (4.2) with $(\tilde{G}, \tilde{c}) = (G^r, c^r) \times$
 834 $(H^r \# \sigma, c_{\sigma})$ and applying Lemma 4.2, the number of colourful k -matchings in $(G^r, c^r) \times$
 835 $(H^r \# \sigma, c_{\sigma})$ equals:

$$836 \quad (4.4) \quad \sum_{\rho} a(\rho) \cdot \#\text{Hom}((H^r \# \rho, c_{\rho}) \rightarrow (G^r, c^r)) \cdot \#\text{Hom}((H^r \# \rho, c_{\rho}) \rightarrow (H^r \# \sigma, c_{\sigma}))$$

837
 838 By Lemma 4.4, $(G^r, c^r) \times (H^r \# \sigma, c_{\sigma})$ is a subgraph of the r -subdivision of a complete
 839 graph, which is in \mathcal{G} by our choice of r . Since \mathcal{G} is monotone this implies $(G^r, c^r) \times$
 840 $(H^r \# \sigma, c_{\sigma}) \in \mathcal{G}$, too. Hence, if we have an oracle for $\#\text{COLMATCH}(\mathcal{G})$, then we can
 841 compute the value of (4.4), while $\#\text{Hom}((H^r \# \rho, c_{\rho}) \rightarrow (H^r \# \sigma, c_{\sigma}))$ can obviously be
 842 computed in a time that is a function of $|H|$ and r . Thus, by letting $\text{coeff}(\rho) := a(\rho) \cdot$
 843 $\#\text{Hom}((H^r \# \rho, c_{\rho}) \rightarrow (G^r, c^r))$, in FPT time we obtain a system of linear equations
 844 with unknowns $\text{coeff}(\rho)$ and whose matrix is M_{H^r} , see (4.1). By Lemma 4.3 M_{H^r} is
 845 nonsingular, hence by solving the system we can retrieve:

$$846 \quad \text{coeff}(\top) = a(\top) \cdot \#\text{Hom}((H^r \# \top, c_{\top}) \rightarrow (G^r, c^r)) = a(\top) \cdot \#\text{Hom}((H^r, \text{id}_{H^r}) \rightarrow (G^r, c^r)).$$

847 Since $a(\top) \neq 0$, we can divide by $a(\top)$ and recover $\#\text{Hom}((H^r, \text{id}_{H^r}) \rightarrow (G^r, c^r))$ as
 848 desired. This concludes the parameterized reduction to $\#\text{COLMATCH}(\mathcal{G})$ and proves
 849 the thesis. \square

¹⁶In [51], the number of edge-colourful k -matchings of G is denoted by $\#\text{ColEdgeSub}(\Phi, k \rightarrow G)$, where Φ is the graph property of being a matching. The identities (4.2) and (4.3) are immediate consequences of Lemma 4.1 and Corollary 4.3 in [51] (see also Lemma 3.1 and Corollary 3.3 in an earlier version [54] of [51]).

850 With the hardness results for $\#\text{COLMATCH}(\mathcal{G})$ above, we can finally obtain our
 851 complexity dichotomy for $\#\text{MATCH}(\mathcal{G})$. First, we prove:

852 **THEOREM 4.7.** *Let \mathcal{G} be a monotone somewhere dense class of graphs. Then*
 853 *$\#\text{MATCH}(\mathcal{G})$ is $\#\text{W}[1]$ -hard and, assuming ETH, cannot be solved in time $f(k) \cdot$*
 854 *$|G|^{o(k/\log k)}$ for any function f .*

855 *Proof.* A well-known application of inclusion-exclusion (see, e.g., [16, Lemma 1.34])
 856 yields a parameterized reduction from $\#\text{COLMATCH}(\mathcal{G})$ to $\#\text{MATCH}(\mathcal{G}')$ that pre-
 857 serves the parameter, where \mathcal{G}' is the class of all subgraphs of \mathcal{G} . By monotonicity
 858 $\mathcal{G}' = \mathcal{G}$, so the claim of Lemma 4.6 holds for $\#\text{MATCH}(\mathcal{G})$, too. \square

859 Finally, we obtain:

860 **COROLLARY 4.8** (Theorem 1.2, restated). *Let \mathcal{G} be a monotone class of graphs*
 861 *and assume that ETH holds. Then $\#\text{MATCH}(\mathcal{G})$ is fixed-parameter tractable if and*
 862 *only if \mathcal{G} is nowhere dense. In particular, if \mathcal{G} is nowhere dense then $\#\text{MATCH}(\mathcal{G})$*
 863 *can be solved in time $f(k) \cdot |V(G)|^{1+o(1)}$ for some computable function f ; otherwise*
 864 *$\#\text{MATCH}(\mathcal{G})$ cannot be solved in time $f(k) \cdot |G|^{o(k/\log k)}$ for any function f .*

865 *Proof.* Immediate from Theorem 2.11 and Theorem 4.7. \square

866 **REMARK 4.9.** *Unless $\#\text{P} = \text{P}$, Corollary 4.8 / Theorem 1.2 cannot be strength-*
 867 *ened to achieve polynomial time tractability of $\#\text{MATCH}(\mathcal{G})$ for nowhere dense and*
 868 *monotone \mathcal{G} . Let indeed \mathcal{G} be the class of all K_8 -minor-free graphs. Then \mathcal{G} is clearly*
 869 *monotone, and since it does not contain the subdivisions of cliques larger than 7, it is*
 870 *also nowhere dense by Lemma 2.9. However, as shown recently by Curticapean and*
 871 *Xia [21], counting perfect matchings (i.e., k -matchings with $k = n/2$) in K_8 -minor-*
 872 *free graphs is $\#\text{P}$ -hard.*

873 **4.2. Counting Subgraphs: Proofs of Theorems 1.3 and 1.4.** Equipped
 874 with our hardness results for counting k -matchings, we move towards proving hardness
 875 for counting subgraphs.

876 **THEOREM 4.10** (Theorem 1.3, restated). *Let \mathcal{H} and \mathcal{G} be graph classes such that*
 877 *\mathcal{H} is hereditary and \mathcal{G} is monotone. Then Table 3 exhaustively classifies the complexity*
 878 *of $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$.*

879 *Proof.* Let us first show that the cases for \mathcal{H} and \mathcal{G} in Table 3 are exhaustive and
 880 mutually exclusive. For \mathcal{G} this is straightforward. For \mathcal{H} , the first row and the rest
 881 are mutually exclusive and exhaustive, since rows 2, 3 and 4 all imply $m(\mathcal{H}) = \infty$. To
 882 see that rows 2, 3, and 4 are mutually exclusive and exhaustive for $m(\mathcal{H}) = \infty$, note
 883 that in that case Theorem 2.5 implies that at least one of $m_{\text{ind}}(\mathcal{H})$, $\beta_{\text{ind}}(\mathcal{H})$ and $\omega(\mathcal{H})$
 884 is unbounded.

885 Let us now prove the entries of Table 3. The first row is due to Curticapean and
 886 Marx [20], and the FPT result in the first column follows from Theorem 2.11. The
 887 intractability results in the second row follow from Theorem 4.7 and the fact that
 888 $m_{\text{ind}}(\mathcal{H}) = \infty$ implies that \mathcal{H} contains all matchings (since \mathcal{H} is hereditary). For the
 889 second column, note that $\omega(\mathcal{G}) = \infty$ and \mathcal{G} being monotone implies that $\mathcal{G} = \mathcal{U}$; the
 890 dichotomy of Curticapean and Marx [20] then applies again.¹⁷ Next, we prove the

¹⁷The tight conditional lower bounds in the second column follow from the fact that the respective entries subsume counting k -cliques in arbitrary graphs, and counting k -by- k bicliques in bipartite graphs. The tight bound of the former was shown in [12, 13], and the tight bound of the latter was implicitly shown in [20], and explicitly in [26]; while [26] studies *induced* subgraphs in bipartite graphs, we note that all bicliques in a bipartite graph must be induced.

	\mathcal{G} n. dense	\mathcal{G} s. dense $\omega(\mathcal{G}) = \infty$	\mathcal{G} s. dense $\omega(\mathcal{G}) < \infty$ $\beta(\mathcal{G}) = \infty$	\mathcal{G} s. dense $\omega(\mathcal{G}) < \infty$ $\beta(\mathcal{G}) < \infty$
$m(\mathcal{H}) < \infty$	P	P	P	P
$m_{\text{ind}}(\mathcal{H}) = \infty$	FPT	hard	hard	hard
$m_{\text{ind}}(\mathcal{H}) < \infty$ $\beta_{\text{ind}}(\mathcal{H}) = \infty$	P	hard [†]	hard [†]	P
$m_{\text{ind}}(\mathcal{H}) < \infty$ $\beta_{\text{ind}}(\mathcal{H}) < \infty$ $\omega(\mathcal{H}) = \infty$	P	hard [†]	P	P

TABLE 3

The complexity of $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ for hereditary \mathcal{H} and monotone \mathcal{G} (Theorem 1.3). P and FPT stand respectively for polynomial-time tractability and fixed-parameter tractability, *hard* means $\#\text{W}[1]$ -hard and without an algorithm running in time $f(|H|) \cdot |G|^{\mathcal{O}(|V(H)|/\log |V(H)|)}$ for any function f unless ETH fails, and *hard[†]* means the same but with a lower bound of $f(|H|) \cdot |G|^{\mathcal{O}(|V(H)|)}$. The FPT entry cannot be strengthened to P unless $P = \#\text{P}$, see Remark 4.12.

891 remaining entries.

892 • Row 3, Column 3: if $\beta(\mathcal{G}) = \beta_{\text{ind}}(\mathcal{H}) = \infty$ then $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ is hard. Since \mathcal{H}
893 is hereditary, it contains all bicliques. Since \mathcal{G} is monotone, it contains all bipartite
894 graphs. Hence $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ is at least as hard as counting k -by- k bicliques in
895 bipartite graphs, which is known to be hard [20].¹⁸

896 • Row 3, Column 4: if $m_{\text{ind}}(\mathcal{H}), \omega(\mathcal{G}), \beta(\mathcal{G}) < \infty$ then $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ is in polynomial
897 time. Let (H, G) be the input of $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$. If $\omega(H) > \omega(\mathcal{G})$ or $\beta_{\text{ind}}(H) >$
898 $\beta(\mathcal{G})$, then we can output 0. We can thus restrict the problem to those H such that
899 $\omega(H) \leq \omega(\mathcal{G})$ and $\beta_{\text{ind}}(H) \leq \beta(\mathcal{G})$. Recall that $m_{\text{ind}}(H) \leq m_{\text{ind}}(\mathcal{H}) < \infty$. By the
900 contrapositive of Theorem 2.5, there is a monotonically increasing function R such
901 that:

$$902 \quad m(H) \leq R(m_{\text{ind}}(H), \omega(H), \beta_{\text{ind}}(H)) \leq R(m_{\text{ind}}(\mathcal{H}), \omega(\mathcal{G}), \beta(\mathcal{G})) < \infty,$$

903 where the second inequality holds by monotonicity of R and the third one by the
904 boundedness of all three arguments. We therefore obtain polynomial time as in the
905 first row.

906 • Row 4, Columns 3 and 4: if $m_{\text{ind}}(\mathcal{H}), \beta_{\text{ind}}(\mathcal{H}), \omega(\mathcal{G}) < \infty$, then $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ is in
907 polynomial time. Let (H, G) be the input of $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$. If $\omega(H) > \omega(\mathcal{G})$ then
908 we output 0, hence we can assume that $\omega(H) \leq \omega(\mathcal{G})$. Similarly to the previous case,
909 we then obtain polynomial time since

$$910 \quad m(H) \leq R(m_{\text{ind}}(H), \omega(H), \beta_{\text{ind}}(H)) \leq R(m_{\text{ind}}(\mathcal{H}), \omega(\mathcal{G}), \beta_{\text{ind}}(\mathcal{H})) < \infty.$$

911 • Rows 3 and 4, Column 1: $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ is in polynomial time. We show that
912 $\omega(\mathcal{G}), \beta(\mathcal{G}) < \infty$; then the same arguments used for Rows 3 and 4 of Column 4

¹⁸See Footnote 17 for the tight conditional lower bound.

913 apply. Suppose by contradiction that $\max(\omega(\mathcal{G}), \beta(\mathcal{G})) = \infty$. Since \mathcal{G} is monotone, if
 914 $\omega(\mathcal{G}) = \infty$ then \mathcal{G} contains (the 0-subdivision of) every clique, and if $\beta(\mathcal{G}) = \infty$ then
 915 \mathcal{G} contains all bipartite graphs and thus the 1-subdivision of every clique. In any case
 916 Lemma 2.9 implies that \mathcal{G} is somewhere dense, contradicting the assumptions. \square

917 Theorem 1.4 follows immediately.

918 COROLLARY 4.11 (Theorem 1.4, restated). *Let \mathcal{H} and \mathcal{G} be monotone graph*
 919 *classes and assume that ETH holds. Then $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ is fixed-parameter tractable*
 920 *if $m(\mathcal{H}) < \infty$ or \mathcal{G} is nowhere dense; otherwise $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ is $\#\text{W}[1]$ -complete*
 921 *and cannot be solved in time $f(|H|) \cdot |G|^{\circ(|V(H)|/\log(|V(H)|))}$ for any function f .*

922 *Proof.* If \mathcal{H} is monotone then \mathcal{H} is hereditary and Theorem 1.3 applies. The union
 923 of the first row and the first column of Table 3 yield the tractable case; the union of
 924 the remaining entries yield the intractable case and the lower bounds. \square

925 We conclude this section with a remark.

926 REMARK 4.12. *Let \mathcal{H} and \mathcal{G} be the classes of graphs of degrees bounded by 2 and*
 927 *3, respectively. Then $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ subsumes the $\#\text{P}$ -hard problem of counting*
 928 *Hamiltonian cycles in 3-regular graphs. Since both classes are monotone (and thus*
 929 *also hereditary), since $m_{\text{ind}}(\mathcal{H}) = \infty$, and since classes of bounded degree graphs are*
 930 *nowhere dense (see e.g. [36]), this shows that the FPT entry in Table 3 cannot be*
 931 *strengthened to P unless $\#\text{P} = \text{P}$.*

932 **5. Counting Induced Subgraphs.** This section is devoted to the proofs of
 933 Theorem 1.5 and Theorem 1.6. We begin in Section 5.1 by analysing the problem of
 934 counting independent sets and proving Theorem 1.5; this is the most technical part.
 935 We then prove Theorem 1.6 in Section 5.2.

936 **5.1. Counting Independent Sets: Proof of Theorem 1.5.** Given a class
 937 of graphs \mathcal{G} , the problem $\#\text{INDSET}(\mathcal{G})$ asks, on input $k \in \mathbb{N}$ and a graph $G \in \mathcal{G}$, to
 938 compute the number of independent sets of size k (also called *k-independent sets*) in
 939 G . In this section we prove hardness results for $\#\text{INDSET}(\mathcal{G})$ and leverage them to
 940 $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$. To this end we will rely on subgraphs induced by sets of edges;
 941 they play a role similar to that of fractured graphs in Section 4. Given a graph F
 942 and a set $A \subseteq E(F)$, we denote the subgraph $(V(F), A)$ by $F[A]$. For what follows
 943 observe that, for any $A \subseteq E(F)$, the identity function on $V(F)$, which we denote
 944 by id_F , is a surjective F -colouring of $F[A]$. Now recall Definition 4.1. We start with
 945 the following simple variation of Lemma 4.4.

946 LEMMA 5.1. *Let $r \in \mathbb{N}_0$, let H be a graph without isolated vertices, let G be an*
 947 *H -coloured graph, and let $A \subseteq E(H^r)$. Then $(G^r, c^r) \times (H^r[A], \text{id}_{H^r})$ is a subgraph of*
 948 *$K_{|V(G)|}^r$.*

949 *Proof.* Let $n = |V(G)|$. First, note that $(G^r, c^r) \times (H^r, \text{id}_{H^r}) = (G^r, c^r)$, and by
 950 construction (G^r, c^r) is a subgraph of K_n^r . Next, for every $A \subseteq E(H^r)$ the graph
 951 $(G^r, c^r) \times (H^r[A], \text{id}_{H^r})$ is obtained from (G^r, c^r) by deleting edges — specifically, for
 952 every $e = \{u, v\} \in E(H^r) \setminus A$, delete from G^r all edges between vertices coloured with
 953 u and vertices coloured with v . Thus $(G^r, c^r) \times (H^r[A], \text{id}_{H^r})$ is a subgraph of K_n^r
 954 too. \square

955 Recall that $\#\text{COLMATCH}(\mathcal{G})$, the problem of counting edge-colourful k -matchings,
 956 was the key subproblem in the hardness proofs for $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$ — see Section 4.1.
 957 In the case of $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$, the key subproblem turns out to be that of counting
 958 vertex-colourful independent sets. Let G be a graph and let $c : V(G) \rightarrow \{1, \dots, k\}$ be

959 a coloring of $V(G)$. A set $U \subseteq V(G)$ is *vertex-colourful* if for every colour in $\{1, \dots, k\}$
 960 there is precisely one element of U with that colour.

961 **DEFINITION 5.2** ($\#\text{COLINDSET}(\mathcal{G})$). *Let \mathcal{G} be a class of graphs. The problem*
 962 $\#\text{COLINDSET}(\mathcal{G})$ *asks, on input $k \in \mathbb{N}$, a graph $G \in \mathcal{G}$, and a k -coloring c of $V(G)$,*
 963 *to compute the number of vertex-colourful k -independent sets in G . The problem is*
 964 *parameterised by k .*

965 Our goal is to show that $\#\text{COLINDSET}(\mathcal{G})$ is intractable whenever \mathcal{G} is monotone
 966 and somewhere dense. As for $\#\text{COLMATCH}(\mathcal{G})$ in Section 4.1, the reduction relies on
 967 solving a system of linear equations. Let H be a graph. The square matrix N_H has
 968 its rows and columns indexed by the subsets of $E[H]$, and its entries satisfy

$$969 \quad (5.1) \quad N_H[A, B] = \#\text{Hom}((H[A], \text{id}_H) \rightarrow (H[B], \text{id}_H)).$$

970 Similarly to the matrix M_H in Section 4.1, the following was established in prior work:

971 **LEMMA 5.3** ([26]). *For each graph H , the matrix N_H is nonsingular.*

972 We are now able to establish intractability of $\#\text{COLINDSET}(\mathcal{G})$.

973 **LEMMA 5.4.** *Let \mathcal{G} be a monotone somewhere dense class of graphs. Then the*
 974 *problem $\#\text{COLINDSET}(\mathcal{G})$ is $\#\text{W}[1]$ -complete and, assuming ETH, cannot be solved*
 975 *in time $f(k) \cdot |G|^{\sigma(k/\log k)}$ for any function f .*

976 *Proof.* The proof is similar to that of Lemma 4.6. First, since \mathcal{G} is monotone
 977 and somewhere dense, by Lemma 2.9 there exists $r \in \mathbb{N}_0$ such that $G^r \in \mathcal{G}$ for every
 978 $G \in \mathcal{U}$. Second, let \mathcal{H} be a class of 3-regular expander graphs. By theorems 2.14
 979 and 2.16, $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{U})$ is $\#\text{W}[1]$ -hard and assuming ETH cannot be solved in
 980 time $f(|H|) \cdot |G|^{\sigma(|H|/\log |H|)}$ for any function f . We show a parameterized reduction
 981 from $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{U})$ to $\#\text{COLINDSET}(\mathcal{G})$.

982 Let $(H, (G, c))$ be the input to $\#\text{CP-HOM}(\mathcal{H} \rightarrow \mathcal{U})$. Our reduction starts by
 983 constructing H^r and (G^r, c^r) , which by Lemma 3.1 satisfy

$$984 \quad \#\text{Hom}((H, \text{id}_H) \rightarrow (G, c)) = \#\text{Hom}((H^r, \text{id}_{H^r}) \rightarrow (G^r, c^r)).$$

985 Let $k = |V(H^r)|$; clearly $k \in O(|H|)$ since r is a constant independent of H . Our goal
 986 is to use the oracle for $\#\text{COLINDSET}(\mathcal{G})$ to compute $\#\text{Hom}((H^r, \text{id}_{H^r}) \rightarrow (G^r, c^r))$.
 987 From now on we view surjectively H^r -coloured graphs (\tilde{G}, \tilde{c}) also as vertex-coloured
 988 graphs with colouring \tilde{c} . This allows us to invoke [26, Lemma 8] and obtain what
 989 follows.¹⁹

990 First, there is a unique function \hat{a} from subsets of $E[H^r]$ to rationals such that,
 991 for every surjectively H^r -coloured graph (\tilde{G}, \tilde{c}) , the number of vertex-colourful k -
 992 independent sets in (\tilde{G}, \tilde{c}) equals

$$993 \quad (5.2) \quad \sum_A \hat{a}(A) \cdot \#\text{Hom}((H^r[A], \text{id}_{H^r}) \rightarrow (\tilde{G}, \tilde{c})),$$

994 where the sum is over all subsets of $E[H^r]$. Additionally,

$$995 \quad (5.3) \quad \hat{a}(E(H^r)) = \pm \chi \neq 0,$$

¹⁹In [26] the number of colourful k -independent sets in a surjectively H^r -coloured graph \tilde{G} is denoted by $\#\text{cp-IndSub}(\Phi \rightarrow_{H^r} \tilde{G})$, where Φ is the graph property of being an independent set.

996 where $\hat{\chi}$ is the so-called alternating enumerator for the graph property of being an
 997 independent set — we omit the definition since the only property needed for $\hat{\chi}$ is it
 998 being easily computable and non-zero (see [26]).

999 Now consider (5.2) with $(\tilde{G}, \tilde{c}) = (G^r, c^r) \times (H^r[B], \text{id}_{H^r})$ and apply Lemma 4.2.
 1000 We deduce that the number of vertex-colourful k -independent sets in (\tilde{G}, \tilde{c}) is

$$1001 \sum_A \hat{a}(A) \cdot \#\text{Hom}((H^r[A], \text{id}_{H^r}) \rightarrow (G^r, c^r)) \cdot \#\text{Hom}((H^r[A], \text{id}_{H^r}) \rightarrow (H^r[B], \text{id}_{H^r})).$$

1003 By Lemma 5.1, for every $B \subseteq E(H^r)$ of H^r the graph $(G^r, c^r) \times (H^r[B], \text{id}_{H^r})$ is a
 1004 subgraph of the r -subdivision of a complete graph; by the monotonicity of \mathcal{G} and by
 1005 the choice of r this implies $(G^r, c^r) \times (H^r[B], \text{id}_{H^r}) \in \mathcal{G}$, see Lemma 2.9. Thus, as in the
 1006 proof of Lemma 4.6, by using an oracle for $\#\text{COLINDSET}(\mathcal{G})$ we can construct in FPT
 1007 time a system of linear equations whose matrix N_{H^r} is nonsingular by Lemma 5.3.
 1008 Since $\hat{a}(E(H^r)) \neq 0$ by (5.3), solving this system enables us to compute

$$1009 \#\text{Hom}((H^r[E(H^r)], \text{id}_{H^r}) \rightarrow (G^r, c^r)) = \#\text{Hom}((H^r, \text{id}_{H^r}) \rightarrow (G^r, c^r)),$$

1010 concluding the proof. \square

1011 With the above hardness results for $\#\text{COLINDSET}(\mathcal{G})$, we can finally prove complex-
 1012 ity dichotomies for its non-coloured counterpart $\#\text{INDSET}(\mathcal{G})$. We start by porting
 1013 Lemma 5.4 from $\#\text{COLINDSET}(\mathcal{G})$ to $\#\text{INDSET}(\mathcal{G})$.

1014 **THEOREM 5.5.** *Let \mathcal{G} be a monotone somewhere dense class of graphs. Then*
 1015 *$\#\text{INDSET}(\mathcal{G})$ is $\#\text{W}[1]$ -hard and, assuming ETH, cannot be solved in time $f(k) \cdot$*
 1016 *$|G|^{o(k/\log k)}$ for any function f .*

1017 *Proof.* Almost identical to the proof of Theorem 4.7: when \mathcal{G} is monotone,
 1018 $\#\text{COLINDSET}(\mathcal{G})$ can be reduced in FPT time to $\#\text{INDSET}(\mathcal{G})$ via inclusion-exclusion
 1019 while preserving the parameter (see, for instance, [16, Lemma 1.34]), and the claim
 1020 then follows by Lemma 5.4. \square

1021 We can finally prove Theorem 1.5 as a simple corollary.

1022 **COROLLARY 5.6** (Theorem 1.5, restated). *Let \mathcal{G} be a monotone class of graphs*
 1023 *and assume that ETH holds. Then $\#\text{INDSET}(\mathcal{G})$ is fixed-parameter tractable if and*
 1024 *only if \mathcal{G} is nowhere dense. In particular, if \mathcal{G} is nowhere dense then $\#\text{INDSET}(\mathcal{G})$*
 1025 *can be solved in time $f(k) \cdot |V(G)|^{1+o(1)}$ for some computable function f ; otherwise*
 1026 *$\#\text{INDSET}(\mathcal{G})$ cannot be solved in time $f(k) \cdot |G|^{o(k/\log k)}$ for any function f .*

1027 *Proof.* Immediate by Theorem 2.11 and Theorem 5.5. \square

1028 We conclude with a remark.

1029 **REMARK 5.7.** *Corollary 5.6 cannot be strengthened to polynomial-time tractability*
 1030 *of $\#\text{INDSET}(\mathcal{G})$ when \mathcal{G} is nowhere dense and monotone, unless $\#\text{P} = \text{P}$: graphs of*
 1031 *degree at most 3 form such a class, yet counting independent sets in them is $\#\text{P}$ -*
 1032 *hard [35].*

1033 **5.2. Counting Induced Subgraphs: Proof of Theorem 1.6.** Equipped with
 1034 our complexity dichotomy for $\#\text{INDSET}(\mathcal{G})$, we can now prove our complexity di-
 1035 chotomies for $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$. First, we consider the case that \mathcal{H} is monotone.

1036 **COROLLARY 5.8.** *Let \mathcal{H} and \mathcal{G} be monotone graph classes and assume that ETH*
 1037 *holds. Then $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ is fixed-parameter tractable if \mathcal{H} is finite or \mathcal{G} is*
 1038 *nowhere dense; otherwise $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ is $\#\text{W}[1]$ -complete and cannot be solved*
 1039 *in time $f(|H|) \cdot |G|^{o(|V(H)|/\log(|V(H)|))}$ for any function f .*

	\mathcal{G} nowhere dense	\mathcal{G} somewhere dense $\omega(\mathcal{G}) = \infty$	\mathcal{G} somewhere dense $\omega(\mathcal{G}) < \infty$ $\alpha(\mathcal{G}) = \infty$
\mathcal{H} finite	P	P	P
$\alpha(\mathcal{H}) = \infty$	FPT	#W[1]-hard not in $f(k) \cdot n^{o(k)}$	#W[1]-hard not in $f(k) \cdot n^{o(k/\log k)}$
$\alpha(\mathcal{H}) < \infty$ $\omega(\mathcal{H}) = \infty$	P	#W[1]-hard not in $f(k) \cdot n^{o(k)}$	P

TABLE 4

The complexity of $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ for hereditary \mathcal{H} and monotone \mathcal{G} . P and FPT stand respectively for polynomial-time tractability and fixed-parameter tractability, and hard means #W[1]-hard and without an algorithm running in time $f(k) \cdot n^{o(k/\log(k))}$ for any function f unless ETH fails, where $k = |V(H)|$ and $n = |V(G)|$. The FPT entry cannot be strengthened to P unless $\text{P} = \#\text{P}$, see Remark 5.7.

1040 *Proof.* If \mathcal{H} is finite then $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ is clearly in polynomial time (and
1041 thus fixed-parameter tractable) since the brute-force algorithm runs in time $O(|G|^{|H|})$.
1042 If \mathcal{G} is nowhere dense then the fixed-parameter tractability follows by Theorem 2.11.
1043 Finally, if \mathcal{H} is monotone and infinite then it contains all independent sets, and thus
1044 $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ subsumes $\#\text{INDSET}(\mathcal{G})$; in which case Theorem 5.5 yields the
1045 lower bound for somewhere dense \mathcal{G} . \square

1046 Next, we consider the case that \mathcal{H} is hereditary. We obtain a refined complexity
1047 classification that subsumes the one of Corollary 5.8 and yields Theorem 1.6.

1048 THEOREM 5.9 (Theorem 1.6, restated). *Let \mathcal{H} and \mathcal{G} be graph classes such that \mathcal{H}*
1049 *is hereditary and \mathcal{G} is monotone. Then Table 4 exhaustively classifies the complexity*
1050 *of $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$.*

1051 *Proof.* The cases for \mathcal{G} and \mathcal{H} in Table 4 are mutually exclusive and exhaustive
1052 by Ramsey's Theorem (Theorem 2.4). Let us then prove the entries of Table 4.

1053 The first row holds since for finite \mathcal{H} the brute-force algorithm runs in polynomial
1054 time, and the FPT result follows from Theorem 2.11. For the intractability results in
1055 the second column, note that since \mathcal{G} is monotone and infinite then $\mathcal{G} = \mathcal{U}$, and since \mathcal{H}
1056 is hereditary, the cases $\alpha(\mathcal{H}) = \infty$ and $\omega(\mathcal{H}) = \infty$ subsume respectively $\#\text{INDSET}(\mathcal{U})$
1057 and $\#\text{CLIQUE}(\mathcal{U})$. Both are canonical #W[1]-hard problems and cannot be solved in
1058 time $f(k) \cdot n^{o(k)}$ unless ETH fails [12, 13].²⁰ The intractability results in the third
1059 column follows from Theorem 5.5 since \mathcal{H} being hereditary and $\alpha(\mathcal{H}) = \infty$ imply that
1060 $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$ subsumes $\#\text{INDSET}(\mathcal{G})$.

1061 It remains to prove the first and the third entry of the third row. Note that both
1062 entries assume $\omega(\mathcal{G}) < \infty$ and $\alpha(\mathcal{H}) < \infty$. Let then (H, G) be the input. If $\omega(H) >$
1063 $\omega(\mathcal{G})$ then we can immediately return 0. Otherwise $|V(H)| \leq R(\omega(\mathcal{G}), \alpha(\mathcal{H})) < \infty$,
1064 where R is Ramsey's function (see Theorem 2.4), and the brute-force algorithm runs
1065 in polynomial time. \square

²⁰The lower bound in [12, 13] applies to counting k -cliques, and we note that counting k -cliques and counting k -independent sets are interreducible by taking the complement of the host.

1066 **6. Outlook.** Due to the absence of a general dichotomy [56], the following two
1067 directions are evident candidates for future analysis.

1068 *Hereditary Host Graphs.* Is there a way to refine our classifications to hereditary
1069 \mathcal{G} ? While such results would naturally be much stronger, we point out that a classi-
1070 fication of general first-order (FO) model-checking and model-counting in hereditary
1071 graphs is wide open. Concretely, even if $\mathcal{H} = \mathcal{U}$, it currently seems elusive to ob-
1072 tain criteria for hereditary \mathcal{G} which, if satisfied, yield fixed-parameter tractability of
1073 $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$, $\#\text{INDSUB}(\mathcal{H} \rightarrow \mathcal{G})$, and $\#\text{HOM}(\mathcal{H} \rightarrow \mathcal{G})$ and which, if not satisfied,
1074 yield $\#\text{W}[1]$ -hardness of those problems. In a nutshell, the problem is that there are
1075 arbitrarily dense hereditary classes of host graphs for which those problems, and even
1076 the much more general FO-model counting problem, become tractable; a trivial ex-
1077 ample is given by \mathcal{G} being the class of all complete graphs. See [31, 33, 34] for recent
1078 work on specific hereditary hosts and [32, 7] for general approaches to understand FO
1079 model checking on dense graphs.

1080 *Arbitrary Pattern Graphs.* Can we refine our classifications to arbitrary classes of
1081 patterns \mathcal{H} , given that we stay in the realm of monotone classes of hosts \mathcal{G} ? We believe
1082 this question is the most promising direction for future research. While a sufficient
1083 and necessary criterion for the fixed-parameter tractability of, say $\#\text{SUB}(\mathcal{H} \rightarrow \mathcal{G})$,
1084 must depend on the set of forbidden subgraphs of \mathcal{G} , we conjecture that the structure
1085 of monotone somewhere dense graph classes is rich enough to allow for an explicit
1086 combinatorial description of such a criterion. In fact, such criteria have already been
1087 established for some specific classes of host graphs, e.g. bipartite graphs [20] and
1088 degenerate graphs [9].

- 1090 [1] Hans Adler and Isolde Adler. Interpreting nowhere dense graph classes as a classical notion of
1091 model theory. *European J. Combin.*, 36:322–330, 2014. doi:10.1016/j.ejc.2013.06.048.
- 1092 [2] Noga Alon, Phuong Dao, Iman Hajirasouliha, Fereydoun Hormozdiari, and S. Cenk Sahin-
1093 nalp. Biomolecular network motif counting and discovery by color coding. *Bioinformatics*,
1094 24(13):i241–i249, 2008. doi:10.1093/bioinformatics/btn163.
- 1095 [3] Noga Alon, Raphael Yuster, and Uri Zwick. Color-Coding. *J. ACM*, 42(4):844–856, 1995.
1096 doi:10.1145/210332.210337.
- 1097 [4] Suman K. Bera, Lior Gishboliner, Yevgeny Levanzov, C. Seshadhri, and Asaf Shapira. Counting
1098 subgraphs in degenerate graphs. *J. ACM*, 69(3):23:1–23:21, 2022. doi:10.1145/3520240.
- 1099 [5] Suman K. Bera, Noujan Pashanasangi, and C. Seshadhri. Linear time subgraph counting,
1100 graph degeneracy, and the chasm at size six. In *Proceedings of the 11th conference on*
1101 *Innovations in Theoretical Computer Science (ITCS)*, pages 38:1–38:20. Schloss Dagstuhl
1102 – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.ITCS.2020.38.
- 1103 [6] Suman K. Bera, Noujan Pashanasangi, and C. Seshadhri. Near-linear time homomorphism
1104 counting in bounded degeneracy graphs: The barrier of long induced cycles. In *Proceedings*
1105 *of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2315–2332.
1106 SIAM, 2021. doi:10.1137/1.9781611976465.138.
- 1107 [7] Édouard Bonnet, Eun Jung Kim, Stéphan Thomassé, and Rémi Watrigant. Twin-width I:
1108 tractable FO model checking. *J. ACM*, 69(1):3:1–3:46, 2022. doi:10.1145/3486655.
- 1109 [8] Marco Bressan, Leslie Ann Goldberg, Kitty Meeks, and Marc Roth. Counting subgraphs in
1110 somewhere dense graphs. In Yael Tauman Kalai, editor, *14th Innovations in Theoretical*
1111 *Computer Science Conference, ITCS 2023, January 10-13, 2023, MIT, Cambridge,*
1112 *Massachusetts, USA*, volume 251 of *LIPIcs*, pages 27:1–27:14. Schloss Dagstuhl - Leibniz-
1113 Zentrum für Informatik, 2023. URL: <https://doi.org/10.4230/LIPIcs.ITCS.2023.27>, doi:
1114 10.4230/LIPIcs.ITCS.2023.27.
- 1115 [9] Marco Bressan and Marc Roth. Exact and approximate pattern counting in degenerate graphs:
1116 New algorithms, hardness results, and complexity dichotomies. In *Proceedings of the 62nd*
1117 *IEEE Annual Symposium on Foundations of Computer Science (FOCS)*, pages 276–285.
1118 IEEE, 2021. doi:10.1109/FOCS52979.2021.00036.
- 1119 [10] Russ Bubley and Martin E. Dyer. Path coupling: A technique for proving rapid mixing in
1120 markov chains. In *Proceedings of the 38th Annual Symposium on Foundations of Computer*
1121 *Science (FOCS)*, pages 223–231. IEEE Computer Society, 1997. doi:10.1109/SFCS.1997.
1122 646111.
- 1123 [11] Hubie Chen and Stefan Mengel. Counting Answers to Existential Positive Queries: A
1124 Complexity Classification. In *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI*
1125 *Symposium on Principles of Database Systems (PODS)*, pages 315–326. ACM, 2016.
1126 doi:10.1145/2902251.2902279.
- 1127 [12] Jianer Chen, Benny Chor, Mike Fellows, Xiuzhen Huang, David W. Juedes, Iyad A. Kanj, and
1128 Ge Xia. Tight lower bounds for certain parameterized NP-hard problems. *Inform. and*
1129 *Comput.*, 201(2):216–231, 2005. doi:10.1016/j.ic.2005.05.001.
- 1130 [13] Jianer Chen, Xiuzhen Huang, Iyad A. Kanj, and Ge Xia. Strong computational lower bounds
1131 via parameterized complexity. *J. Comput. System Sci.*, 72(8):1346–1367, 2006. doi:10.
1132 1016/j.jcss.2006.04.007.
- 1133 [14] Yijia Chen, Marc Thurley, and Mark Weyer. Understanding the Complexity of Induced
1134 Subgraph Isomorphisms. In *Proceedings of the 35th International Colloquium on Au-*
1135 *tomata, Languages and Programming (ICALP)*, pages 587–596. Springer, 2008. doi:
1136 10.1007/978-3-540-70575-8_48.
- 1137 [15] Radu Curticapean. Counting matchings of size k is $W[1]$ -hard. In *Proceedings of the 40th*
1138 *International Colloquium on Automata, Languages, and Programming (ICALP)*, pages
1139 352–363. Springer, 2013. doi:10.1007/978-3-642-39206-1_30.
- 1140 [16] Radu Curticapean. *The simple, little and slow things count: On parameterized counting com-*
1141 *plexity*. PhD thesis, Saarland University, 2015. URL: [http://scidok.sulb.uni-saarland.de/](http://scidok.sulb.uni-saarland.de/volltexte/2015/6217/)
1142 [volltexte/2015/6217/](http://scidok.sulb.uni-saarland.de/volltexte/2015/6217/).
- 1143 [17] Radu Curticapean, Holger Dell, Fedor V. Fomin, Leslie Ann Goldberg, and John Lapinskas.
1144 A fixed-parameter perspective on $\#BIS$. *Algorithmica*, 81(10):3844–3864, 2019. doi:10.
1145 1007/s00453-019-00606-4.
- 1146 [18] Radu Curticapean, Holger Dell, and Dániel Marx. Homomorphisms are a good basis for count-
1147 ing small subgraphs. In *Proceedings of the 49th Annual ACM SIGACT Symposium on The-*
1148 *ory of Computing (STOC)*, pages 210–223. ACM, 2017. doi:10.1145/3055399.3055502.
- 1149 [19] Radu Curticapean, Holger Dell, and Marc Roth. Counting Edge-Injective Homomorphisms

- 1150 and Matchings on Restricted Graph Classes. In *Proceedings of the 34th Symposium on*
1151 *Theoretical Aspects of Computer Science (STACS)*, pages 25:1–25:15. Schloss Dagstuhl –
1152 Leibniz-Zentrum für Informatik, 2017. doi:10.4230/LIPIcs.STACS.2017.25.
- 1153 [20] Radu Curticapean and Dániel Marx. Complexity of Counting Subgraphs: Only the Bound-
1154 edness of the Vertex-Cover Number Counts. In *Proceedings of the 55th IEEE Annual*
1155 *Symposium on Foundations of Computer Science (FOCS)*, pages 130–139. IEEE, 2014.
1156 doi:10.1109/FOCS.2014.22.
- 1157 [21] Radu Curticapean and Mingji Xia. Parameterizing the permanent: Hardness for fixed ex-
1158 cluded minors. In *Proceedings of the 5th Symposium on Simplicity in Algorithms,*
1159 *(SOSA@SODA)*, pages 297–307. SIAM, 2022. doi:10.1137/1.9781611977066.23.
- 1160 [22] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin
1161 Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer,
1162 2015. doi:10.1007/978-3-319-21275-3.
- 1163 [23] Víctor Dalmau and Peter Jonsson. The complexity of counting homomorphisms seen from the
1164 other side. *Theoret. Comput. Sci.*, 329(1-3):315–323, 2004. doi:10.1016/j.tcs.2004.08.
1165 008.
- 1166 [24] Josep Díaz, Maria J. Serna, and Dimitrios M. Thilikos. Counting H-colorings of partial k-trees.
1167 *Theoret. Comput. Sci.*, 281(1-2):291–309, 2002. doi:10.1016/S0304-3975(02)00017-8.
- 1168 [25] Reinhard Diestel. *Graph Theory (5th Edition)*. Springer, 2017. doi:10.1007/
1169 978-3-662-53622-3.
- 1170 [26] Julian Dörfler, Marc Roth, Johannes Schmitt, and Philip Wellnitz. Counting Induced Sub-
1171 graphs: An Algebraic Approach to #W[1]-Hardness. *Algorithmica*, 84(2):379–404, 2022.
1172 doi:10.1007/s00453-021-00894-9.
- 1173 [27] Zdenek Dvorák, Daniel Král, and Robin Thomas. Testing first-order properties for subclasses
1174 of sparse graphs. *J. ACM*, 60(5):36:1–36:24, 2013. doi:10.1145/2499483.
- 1175 [28] Jörg Flum and Martin Grohe. The Parameterized Complexity of Counting Problems. *SIAM*
1176 *J. Comput.*, 33(4):892–922, 2004. doi:10.1137/S0097539703427203.
- 1177 [29] Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Texts in Theoretical Com-
1178 puter Science. An EATCS Series. Springer, 2006. doi:10.1007/3-540-29953-X.
- 1179 [30] Jacob Focke and Marc Roth. Counting small induced subgraphs with hereditary properties.
1180 In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*
1181 *(STOC)*, pages 1543–1551. ACM, 2022. doi:10.1145/3519935.3520008.
- 1182 [31] Jakub Gajarský, Petr Hliněný, Daniel Lokshtanov, Jan Obdržálek, Sebastian Ordyniak, M. S.
1183 Ramanujan, and Saket Saurabh. FO model checking on posets of bounded width. In
1184 *Proceedings of the 56th IEEE Annual Symposium on Foundations of Computer Science*
1185 *(FOCS)*, pages 963–974. IEEE Computer Society, 2015. doi:10.1109/FOCS.2015.63.
- 1186 [32] Jakub Gajarský, Petr Hliněný, Jan Obdržálek, Daniel Lokshtanov, and M. S. Ramanujan. A
1187 new perspective on FO model checking of dense graph classes. *ACM Trans. Comput. Log.*,
1188 21(4):28:1–28:23, 2020. doi:10.1145/3383206.
- 1189 [33] Robert Ganian, Petr Hliněný, Daniel Král, Jan Obdržálek, Jarett Schwartz, and Jakub Teska.
1190 FO model checking of interval graphs. *Log. Methods Comput. Sci.*, 11(4), 2015. doi:
1191 10.2168/LMCS-11(4:11)2015.
- 1192 [34] Colin Geniet and Stéphane Thomassé. First order logic and twin-width in tournaments and
1193 dense oriented graphs. *CoRR*, abs/2207.07683, 2022. arXiv:2207.07683.
- 1194 [35] Catherine S. Greenhill. The complexity of counting colourings and independent sets in sparse
1195 graphs and hypergraphs. *Comput. Complexity*, 9(1):52–72, 2000. doi:10.1007/PL00001601.
- 1196 [36] Martin Grohe, Stephan Kreutzer, and Sebastian Siebertz. Deciding first-order properties of
1197 nowhere dense graphs. *J. ACM*, 64(3):17:1–17:32, 2017. doi:10.1145/3051095.
- 1198 [37] Martin Grohe and Dániel Marx. On tree width, bramble size, and expansion. *J. Combin.*
1199 *Theory Ser. B*, 99(1):218–228, 2009. doi:10.1016/j.jctb.2008.06.004.
- 1200 [38] Russell Impagliazzo and Ramamohan Paturi. On the Complexity of k-SAT. *J. Comput. System*
1201 *Sci.*, 62(2):367–375, 2001. doi:10.1006/jcss.2000.1727.
- 1202 [39] Mark Jerrum and Kitty Meeks. The parameterised complexity of counting connected subgraphs
1203 and graph motifs. *J. Comput. System Sci.*, 81(4):702–716, 2015. doi:10.1016/j.jcss.
1204 2014.11.015.
- 1205 [40] Mark Jerrum, Alistair Sinclair, and Eric Vigoda. A polynomial-time approximation algorithm
1206 for the permanent of a matrix with nonnegative entries. *J. ACM*, 51(4):671–697, 2004.
1207 doi:10.1145/1008731.1008738.
- 1208 [41] Oleksii Kuchaiev, Tijana Milenković, Vesna Memišević, Wayne Hayes, and Nataša Pržulj. Topo-
1209 logical network alignment uncovers biological function and phylogeny. *Journal of the Royal*
1210 *Society Interface*, 7(50):1341–1354, 2010. doi:10.1098/rsif.2010.0063.
- 1211 [42] László Lovász. *Large Networks and Graph Limits*, volume 60 of *Colloquium Publications*. Amer-

- 1212 ican Mathematical Society, 2012. URL: [http://www.ams.org/bookstore-getitem/item=](http://www.ams.org/bookstore-getitem/item=COLL-60)
1213 [COLL-60](http://www.ams.org/bookstore-getitem/item=COLL-60).
- 1214 [43] Dániel Marx. Can You Beat Treewidth? *Theory Comput.*, 6(1):85–112, 2010. doi:10.4086/
1215 [toc.2010.v006a005](https://doi.org/10.4086/toc.2010.v006a005).
- 1216 [44] Dániel Marx. Tractable Hypergraph Properties for Constraint Satisfaction and Conjunctive
1217 Queries. *J. ACM*, 60(6):42:1–42:51, 2013. doi:10.1145/2535926.
- 1218 [45] Kitty Meeks. The challenges of unbounded treewidth in parameterised subgraph counting
1219 problems. *Discrete Appl. Math.*, 198:170–194, 2016. doi:10.1016/j.dam.2015.06.019.
- 1220 [46] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network Motifs:
1221 Simple Building Blocks of Complex Networks. *Science*, 298(5594):824–827, 2002. doi:
1222 [10.1126/science.298.5594.824](https://doi.org/10.1126/science.298.5594.824).
- 1223 [47] Ron Milo, Shalev Itzkovitz, Nadav Kashtan, Reuven Levitt, Shai Shen-Orr, Inbal Ayzenshtat,
1224 Michal Sheffer, and Uri Alon. Superfamilies of evolved and designed networks. *Science*,
1225 303(5663):1538–1542, 2004. doi:10.1126/science.1089167.
- 1226 [48] Jaroslav Nešetřil and Patrice Ossona de Mendez. On nowhere dense graphs. *Eur. J. Comb.*,
1227 32(4):600–617, 2011. doi:10.1016/j.ejc.2011.01.006.
- 1228 [49] Jaroslav Nešetřil and Patrice Ossona de Mendez. *Sparsity – Graphs, Structures, and Al-*
1229 *gorithms*, volume 28 of *Algorithms and combinatorics*. Springer, 2012. doi:10.1007/
1230 [978-3-642-27875-4](https://doi.org/10.1007/978-3-642-27875-4).
- 1231 [50] Jaroslav Nešetřil and Svatopluk Poljak. On the complexity of the subgraph problem. *Comment.*
1232 *Math. Univ. Carolin.*, 26(2):415–419, 1985. URL: <https://eudml.org/doc/17394>.
- 1233 [51] Norbert Peyerimhoff, Marc Roth, Johannes Schmitt, Jakob Stix, Alina Vdovina, and Philip
1234 Wellnitz. Parameterized Counting and Cayley Graph Expanders. *SIAM J. Discrete Math.*,
1235 to appear.
- 1236 [52] Neil Robertson and Paul D. Seymour. Graph minors. V. Excluding a planar graph. *J. Combin.*
1237 *Theory Ser. B*, 41(1):92–114, 1986. doi:10.1016/0095-8956(86)90030-4.
- 1238 [53] Marc Roth. *Counting Problems on Quantum Graphs: Parameterized and Exact Complex-*
1239 *ity Classifications*. PhD thesis, Saarland University, 2019. URL: [https://scidok.sulb.](https://scidok.sulb.uni-saarland.de/bitstream/20.500.11880/27575/1/thesis.pdf)
1240 [uni-saarland.de/bitstream/20.500.11880/27575/1/thesis.pdf](https://scidok.sulb.uni-saarland.de/bitstream/20.500.11880/27575/1/thesis.pdf).
- 1241 [54] Marc Roth, Johannes Schmitt, and Philip Wellnitz. Detecting and counting small subgraphs,
1242 and evaluating a parameterized tutte polynomial: Lower bounds via toroidal grids and
1243 cayley graph expanders. *CoRR*, abs/2011.03433, 2020. arXiv:2011.03433.
- 1244 [55] Marc Roth, Johannes Schmitt, and Philip Wellnitz. Counting Small Induced Subgraphs Sat-
1245 isfying Monotone Properties. *SIAM J. Comput.*, 0(0):FOCS20–139–FOCS20–174, 2022.
1246 doi:10.1137/20M1365624.
- 1247 [56] Marc Roth and Philip Wellnitz. Counting and finding homomorphisms is universal for parame-
1248 terized complexity theory. In *Proceedings of the 2020 ACM-SIAM Symposium on Discrete*
1249 *Algorithms (SODA)*, pages 2161–2180. SIAM, 2020. doi:10.1137/1.9781611975994.133.
- 1250 [57] Benjamin Schiller, Sven Jager, Kay Hamacher, and Thorsten Strufe. Stream – A Stream-
1251 Based Algorithm for Counting Motifs in Dynamic Graphs. In *Proceedings of the 2nd*
1252 *International Conference on Algorithms for Computational Biology (AlCoB)*, pages 53–
1253 67. Springer International Publishing, 2015. doi:10.1007/978-3-319-21233-3_5.
- 1254 [58] Carsten Thomassen. On the presence of disjoint subgraphs of a specified type. *J. Graph Theory*,
1255 12(1):101–111, 1988. doi:10.1002/jgt.3190120111.
- 1256 [59] Ngoc Hieu Tran, Kwok Pui Choi, and Louxin Zhang. Counting motifs in the human interactome.
1257 *Nature communications*, 4(1):1–8, 2013. doi:10.1038/ncomms3241.
- 1258 [60] Charalampos E. Tsourakakis, Jakub Pachocki, and Michael Mitzenmacher. Scalable motif-
1259 aware graph clustering. In *Proceedings of the 26th International Conference on World*
1260 *Wide Web (WWW)*, page 1451–1460, 2017. doi:10.1145/3038912.3052653.
- 1261 [61] Leslie G. Valiant. The Complexity of Enumeration and Reliability Problems. *SIAM J. Comput.*,
1262 8(3):410–421, 1979. doi:10.1137/0208032.