



HAL
open science

Parameterized Complexity of Independent Set in H-free graphs

Edouard Bonnet, Nicolas Bousquet, Pierre Charbit, Stéphan Thomassé, Rémi Watrigant

► **To cite this version:**

Edouard Bonnet, Nicolas Bousquet, Pierre Charbit, Stéphan Thomassé, Rémi Watrigant. Parameterized Complexity of Independent Set in H-free graphs. IPEC 2018 - 13th International Symposium on Parameterized and Exact Computation, Aug 2018, Helsinki, Finland. 10.4230/LIPIcs.CVIT.2016.23 . hal-01962369

HAL Id: hal-01962369

<https://inria.hal.science/hal-01962369v1>

Submitted on 20 Dec 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

1 Parameterized Complexity of Independent Set in 2 H-Free Graphs

3 Édouard Bonnet

4 Univ Lyon, CNRS, ENS de Lyon, Université Claude Bernard Lyon 1, LIP UMR5668, France

5 Nicolas Bousquet

6 CNRS, G-SCOP laboratory, Grenoble-INP, France

7 Pierre Charbit

8 Université Paris Diderot, IRIF, France

9 Stéphane Thomassé

10 Univ Lyon, CNRS, ENS de Lyon, Université Claude Bernard Lyon 1, LIP UMR5668, France

11 Institut Universitaire de France

12 Rémi Watrigant

13 Univ Lyon, CNRS, ENS de Lyon, Université Claude Bernard Lyon 1, LIP UMR5668, France

14 — Abstract —

15 In this paper, we investigate the complexity of MAXIMUM INDEPENDENT SET (MIS) in the class
16 of H -free graphs, that is, graphs excluding a fixed graph as an induced subgraph. Given that
17 the problem remains NP -hard for most graphs H , we study its fixed-parameter tractability and
18 make progress towards a dichotomy between FPT and $W[1]$ -hard cases. We first show that MIS
19 remains $W[1]$ -hard in graphs forbidding simultaneously $K_{1,4}$, any finite set of cycles of length at
20 least 4, and any finite set of trees with at least two branching vertices. In particular, this answers
21 an open question of Dabrowski *et al.* concerning C_4 -free graphs. Then we extend the polynomial
22 algorithm of Alekseev when H is a disjoint union of edges to an FPT algorithm when H is a
23 disjoint union of cliques. We also provide a framework for solving several other cases, which is a
24 generalization of the concept of *iterative expansion* accompanied by the extraction of a particular
25 structure using Ramsey's theorem. Iterative expansion is a maximization version of the so-called
26 *iterative compression*. We believe that our framework can be of independent interest for solving
27 other similar graph problems. Finally, we present positive and negative results on the existence
28 of polynomial (Turing) kernels for several graphs H .

29 **2012 ACM Subject Classification** Theory of computation → Fixed parameter tractability

30 **Keywords and phrases** Parameterized Algorithms, Independent Set, H-Free Graphs

31 **Digital Object Identifier** 10.4230/LIPIcs.CVIT.2016.23

32 **Funding** É. B. is supported by the LABEX MILYON (ANR-10- LABX-0070) of Université de
33 Lyon, within the program “Investissements d’Avenir” (ANR-11-IDEX-0007) operated by the
34 French National Research Agency (ANR). N. B. and P. C. are supported by the ANR Project
35 DISTANCIA (ANR-17-CE40-0015) operated by the French National Research Agency (ANR).

36 1 Introduction

37 Given a simple graph G , a set of vertices $S \subseteq V(G)$ is an *independent set* if the vertices of
38 this set are all pairwise non-adjacent. Finding an independent set with maximum cardinality
39 is a fundamental problem in algorithmic graph theory, and is known as the MIS problem
40 (MIS, for short) [11]. In general graphs, it is not only NP -hard, but also not approximable



© É. Bonnet, N. Bousquet, P. Charbit, S. Thomassé, R. Watrigant;
licensed under Creative Commons License CC-BY

42nd Conference on Very Important Topics (CVIT 2016).

Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:13

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

41 within $O(n^{1-\epsilon})$ for any $\epsilon > 0$ unless $P = NP$ [19], and $W[1]$ -hard [9] (unless otherwise
 42 stated, n always denotes the number of vertices of the input graph). Thus, it seems natural
 43 to study the complexity of MIS in restricted graph classes. One natural way to obtain such
 44 a restricted graph class is to forbid some given pattern to appear in the input. For a fixed
 45 graph H , we say that a graph is H -free if it does not contain H as an induced subgraph.
 46 Unfortunately, it turns out that for most graphs H , MIS in H -free graphs remains NP -hard,
 47 as shown by a very simple reduction first observed by Alekseev:

48 ► **Theorem 1** ([1]). *Let H be a connected graph which is neither a path nor a subdivision of*
 49 *the claw. Then MIS is NP -hard in H -free graphs.*

50 On the positive side, the case of P_t -free graphs has attracted a lot of attention during the
 51 last decade. While it is still open whether there exists $t \in \mathbb{N}$ for which MIS is NP -hard in P_t -
 52 free graphs, quite involved polynomial-time algorithms were discovered for P_5 -free graphs [16],
 53 and very recently for P_6 -free graphs [12]. In addition, we can also mention the recent following
 54 result: MIS admits a subexponential algorithm running in time $2^{O(\sqrt{tn \log n})}$ in P_t -free graphs
 55 for every $t \in \mathbb{N}$ [3]. The second open question concerns the subdivision of the claw. Let $S_{i,j,k}$
 56 be a tree with exactly three vertices of degree one, being at distance i, j and k from the
 57 unique vertex of degree three. The complexity of MIS is still open in $S_{1,2,2}$ -free graphs and
 58 $S_{1,1,3}$ -free graphs. In this direction, the only positive results concern some subcases: it is
 59 polynomial-time solvable in $(S_{1,2,2}, S_{1,1,3}, \textit{dart})$ -free graphs [14], $(S_{1,1,3}, \textit{banner})$ -free graphs
 60 and $(S_{1,1,3}, \textit{bull})$ -free graphs [15], where *dart*, *banner* and *bull* are particular graphs on five
 61 vertices. Given the large number of graphs H for which the problem remains NP -hard, it
 62 seems natural to investigate the existence of parameterized algorithms¹, that is, determining
 63 the existence of an independent set of size k in a graph with n vertices in time $O(f(k)n^c)$ for
 64 some computable function f and constant c . A very simple case concerns K_r -free graphs,
 65 that is, graphs excluding a clique of size r . In that case, Ramsey's theorem implies that
 66 every such graph G admits an independent set of size $\Omega(n^{\frac{1}{r-1}})$, where $n = |V(G)|$. In the
 67 FPT vocabulary, it implies that MIS in K_r -free graphs has a kernel with $O(k^{r-1})$ vertices.

68 To the best of our knowledge, the first step towards an extension of this observation
 69 within the FPT framework is the work of Dabrowski *et al.* [7] (see also Dabrowski's PhD
 70 manuscript [6]) who showed, among others, that for any positive integer r , MAX WEIGHTED
 71 INDEPENDENT SET is FPT in H -free graphs when H is a clique of size r minus an edge. In
 72 the same paper, they settle the parameterized complexity of MIS on almost all the remaining
 73 cases of H -free graphs when H has at most four vertices. The conclusion is that the problem
 74 is FPT on those classes, except for $H = C_4$ which is left open. We answer this question by
 75 showing that MIS remains $W[1]$ -hard in a subclass of C_4 -free graphs. On the negative side,
 76 it was proved that MIS remains $W[1]$ -hard in $K_{1,4}$ -free graphs [13].

77 Finally, we can also mention the case where H is the *bull* graph, which is a triangle with
 78 a pending vertex attached to two different vertices. For that case, a polynomial Turing kernel
 79 was obtained [18] then improved [10].

80 1.1 Our results

81 In Section 2, we present three reductions proving $W[1]$ -hardness of MIS in graph excluding
 82 several graphs as induced subgraphs, such as $K_{1,4}$, any fixed cycle of length at least four,

¹ For the sake of simplicity, "MIS" will denote the optimisation, decision and parameterized version of the problem (in the latter case, the parameter is the size of the solution), the correct use being clear from the context.

and any fixed tree with two branching vertices. In Section 3, we extend the polynomial algorithm of Alekseev when H is a disjoint union of edges to an *FPT* algorithm when H is a disjoint union of cliques. In Section 4, we present a general framework extending the technique of *iterative expansion*, which itself is the maximization version of the well-known iterative compression technique. We apply this framework to provide *FPT* algorithms when H is a clique minus a complete bipartite graph, or when H is a clique minus a triangle. Finally, in Section 5, we focus on the existence of polynomial (Turing) kernels. We first strengthen some results of the previous section by providing polynomial (Turing) kernels in the case where H is a clique minus a claw. Then, we prove that for many H , MIS on H -free graphs does not admit a polynomial kernel, unless $NP \subseteq coNP/poly$. Our results allows to obtain the complete dichotomy polynomial/polynomial kernel (PK)/no PK but polynomial Turing kernel/ $W[1]$ -hard for all possible graphs on four vertices, while only five graphs on five vertices remain open for the *FPT*/ $W[1]$ -hard dichotomy.

Due to space restrictions, proofs marked with a (\star) were omitted, and can be found in the long version of the paper. This long version also contains additional figures, and two variants of the reduction presented in the next section, together with a discussion.

1.2 Notation

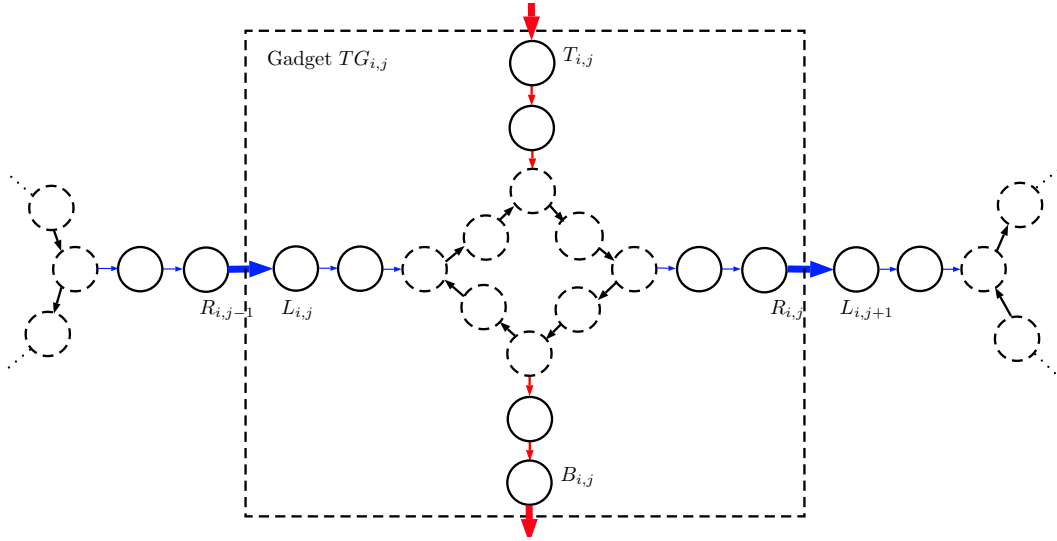
For classical notation related to graph theory or fixed-parameter tractable algorithms, we refer the reader to the monographs [8] and [9], respectively. For an integer $r \geq 2$ and a graph H with vertex set $V(H) = \{v_1, \dots, v_{n_H}\}$ with $n_H \leq r$, we denote by $K_r \setminus H$ the graph with vertex set $\{1, \dots, r\}$ and edge set $\{ab : 1 \leq a, b \leq r \text{ such that } v_a v_b \notin E(H)\}$. For $X \subseteq V(G)$, we write $G \setminus X$ to denote $G[V(G) \setminus X]$. For two graphs G and H , we denote by $G \uplus H$ the *disjoint union* operation, that is, the graph with vertex set $V(G) \cup V(H)$ and edge set $E(G) \cup E(H)$. We denote by $G + H$ the *join* operation of G and H , that is, the graph with vertex set $V(G) \cup V(H)$ and edge set $E(G) \cup E(H) \cup \{uv : u \in V(G), v \in V(H)\}$. For two integers r, k , we denote by $Ram(r, k)$ the Ramsey number of r and k , *i.e.* the minimum order of a graph to contain either a clique of size r or an independent set of size k . We write for short $Ram(k) = Ram(k, k)$. Finally, for $\ell, k > 0$, we denote by $Ram_\ell(k)$ the minimum order of a complete graph whose edges are colored with ℓ colors to contain a monochromatic clique of size k .

2 $W[1]$ -hardness

► **Theorem 2.** *For any $p_1 \geq 4$ and $p_2 \geq 1$, MIS remains $W[1]$ -hard in graphs excluding simultaneously the following graphs as induced subgraphs: $K_{1,4}$, C_4 , \dots , C_{p_1} and any tree T with two branching vertices² at distance at most p_2 .*

Proof. Let $p = \max\{p_1, p_2\}$. We reduce from GRID TILING, where the input is composed of k^2 sets $S_{i,j} \subseteq [m] \times [m]$ ($0 \leq i, j \leq k-1$), called *tiles*, each composed of n elements. The objective of GRID TILING is to find an element $s_{i,j}^* \in S_{i,j}$ for each $0 \leq i, j \leq k-1$, such that $s_{i,j}^*$ agrees in the first coordinate with $s_{i,j+1}^*$, and agrees in the second coordinate with $s_{i+1,j}^*$, for every $0 \leq i, j \leq k-1$ (incrementations of i and j are done modulo k). In such case, we say that $\{s_{i,j}^*, 0 \leq i, j \leq k-1\}$ is a *feasible solution* of the instance. It is known that GRID TILING is $W[1]$ -hard parameterized by k [5].

² A branching vertex in a tree is a vertex of degree at least 3.



■ **Figure 1** Gadget $TG_{i,j}$ representing a tile and its adjacencies with $TG_{i,j-1}$ and $TG_{i,j+1}$, for $p = 1$. Each circle is a clique on n vertices (dashed cliques are the cycle cliques). Black, blue and red arrows represent respectively type T_h , T_r and T_c edges (bold arrows are between two gadgets).

124

125 Before describing formally the reduction, let us give some definitions and ideas. Given
 126 $s = (a, b)$ and $s' = (a', b')$, we say that s is *row-compatible* (resp. *column-compatible*) with
 127 s' if $a \geq a'$ (resp. $b \geq b'$)³. Observe that a solution $\{s_{i,j}^*, 0 \leq i, j \leq k-1\}$ is feasible if
 128 and only if $s_{i,j}^*$ is row-compatible with $s_{i,j+1}^*$ and column-compatible with $s_{i+1,j}^*$ for every
 129 $0 \leq i, j \leq k-1$ (incrementations of i and j are done modulo k). Informally, the main
 130 idea of the reduction is that, when representing a tile by a clique, the row-compatibility
 131 (resp. column-compatibility) relation (as well as its complement) forms a C_4 -free graph when
 132 considering two consecutive tiles, and a claw-free graph when considering three consecutive
 133 tiles. The main difficulty is to forbid the desired graphs to appear in the “branchings” of
 134 tiles. We now describe the reduction.

135 For every tile $S_{i,j} = \{s_1^{i,j}, \dots, s_n^{i,j}\}$, we construct a *tile gadget* $TG_{i,j}$, depicted in Figure 1.
 136 Notice that this gadget shares some ideas with the $W[1]$ -hardness of the problem in $K_{1,4}$ -free
 137 graphs by Hermelin *et al.* [13]. To define this gadget, we first describe an oriented graph
 138 with three types of arcs (type T_h , T_r and T_c , which respectively stands for *half graph*, *row*
 139 and *column*, this meaning will become clearer later), and then explain how to represent the
 140 vertices and arcs of this graph to get the concrete gadget. Consider first a directed cycle on
 141 $4p+4$ vertices c_1, \dots, c_{4p+4} with arcs of type T_h . Then consider four oriented paths on $p+1$
 142 vertices: P_1, P_2, P_3 and P_4 . P_1 and P_3 are composed of arcs of type T_c , while P_2 and P_4 are
 143 composed of arcs of type T_r . Put an arc of type T_c between the last vertex of P_1 and c_1 , an
 144 arc of type T_c between c_{2p+3} and the first vertex of P_3 , an arc of type T_r between c_{p+2} and
 145 the first vertex of P_2 , and an arc of type T_r between the last vertex of P_4 and c_{3p+4} .

146 Now, replace every vertex of this oriented graph by a clique on n vertices, and fix an
 147 arbitrary ordering on the vertices of each clique. For each arc of type T_h between c and c' ,

³ Notice that the row-compatibility (resp. column-compatibility) relation is not symmetrical.

148 add a half graph⁴ between the corresponding cliques: connect the a^{th} vertex of the clique
 149 representing c with the b^{th} vertex of the clique representing c' iff $a > b$. For every arc of
 150 type T_r from a vertex c to a vertex c' , connect the a^{th} vertex of the clique representing
 151 c with the b^{th} vertex of the clique representing c' iff $s_a^{i,j}$ is *not* row-compatible with $s_b^{i,j}$.
 152 Similarly, for every arc of type T_c from a vertex c to a vertex c' , connect the a^{th} vertex
 153 of the clique representing C with the b^{th} vertex of the clique representing c' iff $s_a^{i,j}$ is *not*
 154 column-compatible with $s_b^{i,j}$. The cliques corresponding to vertices of this gadget are called
 155 the *main cliques* of $TG_{i,j}$, and the cliques corresponding to the central cycle on $4p+4$ vertices
 156 are called the *cycle cliques*. The main cliques which are not cycle cliques are called *path*
 157 *cliques*. The cycle cliques adjacent to one path clique are called *branching cliques*. Finally,
 158 the clique corresponding to the vertex of degree one in the path attached to c_1 (resp. c_{p+2} ,
 159 c_{2p+3} , c_{3p+4}) is called the *top* (resp. *right*, *bottom*, *left*) clique of $TG_{i,j}$, denoted by $T_{i,j}$ (resp.
 160 $R_{i,j}$, $B_{i,j}$, $L_{i,j}$). Let $T_{i,j} = \{t_1^{i,j}, \dots, t_n^{i,j}\}$, $R_{i,j} = \{r_1^{i,j}, \dots, r_n^{i,j}\}$, $B_{i,j} = \{b_1^{i,j}, \dots, b_n^{i,j}\}$, and
 161 $L_{i,j} = \{\ell_1^{i,j}, \dots, \ell_n^{i,j}\}$. For the sake of readability, we might omit the superscripts i, j when it
 162 is clear from the context.

163 ► **Lemma 3.** (\star) *Let K be an independent set of size $8(p+1)$ in $TG_{i,j}$. Then:*

- 164 (a) *K intersects all the cycle cliques on the same index x ;*
 165 (b) *if $K \cap T_{i,j} = \{t_{x_t}\}$, $K \cap R_{i,j} = \{r_{x_r}\}$, $K \cap B_{i,j} = \{b_{x_b}\}$, and $K \cap L_{i,j} = \{\ell_{x_\ell}\}$. Then:*
 166 ■ *$s_{x_\ell}^{i,j}$ is row-compatible with $s_x^{i,j}$ which is row-compatible with $s_{x_r}^{i,j}$, and*
 167 ■ *$s_{x_t}^{i,j}$ is column-compatible with $s_x^{i,j}$ which is column-compatible with $s_{x_b}^{i,j}$.*

168 For $i, j \in \{0, \dots, k-1\}$, we connect the right clique of $TG_{i,j}$ with the left clique of
 169 $TG_{i,j+1}$ in a “type T_r spirit”: for every $x, y \in [n]$, connect $r_x^{i,j} \in R_{i,j}$ with $\ell_y^{i,j+1} \in L_{i,j+1}$ iff
 170 $s_x^{i,j}$ is *not* row-compatible with $s_y^{i,j+1}$. Similarly, we connect the bottom clique of $TG_{i,j}$ with
 171 the top clique of $TG_{i+1,j}$ in a “type T_c spirit”: for every $x, y \in [n]$, connect $b_x^{i,j} \in B_{i,j}$ with
 172 $t_y^{i+1,j} \in T_{i+1,j}$ iff $s_x^{i,j}$ is *not* column-compatible with $s_y^{i+1,j}$ (all incrementations of i and j
 173 are done modulo k). This terminates the construction of the graph G .

174 ► **Lemma 4.** (\star) *The input instance of GRID TILING is positive if and only if G has an*
 175 *independent set of size $k' = 8(p+1)k^2$.*

176 Let us now prove that G does not contain the graphs mentionned in the statement as an
 177 induced subgraph:

- 178 (i) $K_{1,4}$: we first prove that for every $0 \leq i, j \leq k-1$, the graph induced by the cycle
 179 cliques of $TG_{i,j}$ is claw-free. For the sake of contradiction, suppose that there exist three
 180 consecutive cycle cliques A , B and C containing a claw. W.l.o.g. we may assume that
 181 $b_x \in B$ is the center of the claw, and $a_\alpha \in A$, $b_\beta \in B$ and $c_\gamma \in C$ are the three endpoints.
 182 By construction of the gadgets (there is a half graph between A and B and between B
 183 and C), we must have $\alpha < x < \gamma$. Now, observe that if $x < \beta$ then a_α must be adjacent
 184 to b_β , and if $\beta < x$, then b_β must be adjacent to c_γ , but both case are impossible since
 185 $\{a_\alpha, b_\beta, c_\gamma\}$ is supposed to be an independent set. Similarly, we can prove that the graph
 186 induced by each path of size $2(p+1)$ linking two consecutive gadgets is claw-free. Hence,
 187 the only way for $K_{1,4}$ to appear in G would be that the center appears in the cycle

⁴ Notice that our definition of half graph slightly differs from the usual one, in the sense that we do not put edges relying two vertices of the same index. Hence, our construction can actually be seen as the complement of a half graph (which is consistent with the fact that usually, both parts of a half graph are independent sets, while they are cliques in our gadgets).

188 clique attached to a path, for instance in the clique represented by the vertex c_1 in the
 189 cycle. However, it can easily be seen that in this case, a claw must lie either in the graph
 190 induced by the cycle cliques of the gadget, or in the path linking $TG_{i,j}$ with $TG_{i-1,j}$,
 191 which is impossible.

192 (ii) C_4, \dots, C_{p_1} . The main argument is that the graph induced by any two main cliques does
 193 not contain any of these cycles. Then, we show that such a cycle cannot lie entirely in
 194 the cycle cliques of a single gadget $TG_{i,j}$. Indeed, if this cycle uses at most one vertex
 195 per main clique, then it must be of length at least $4p + 4$. If it intersects a clique C on
 196 two vertices, then either it also intersect all the cycle cliques of the gadget, in which case
 197 it is of length $4p + 5$, or it intersects an adjacent clique of C on two vertices, in which
 198 case these two cliques induce a C_4 , which is impossible. Similarly, such a cycle cannot lie
 199 entirely in a path between the main cliques of two gadgets. Finally, the main cliques of
 200 two gadgets are at distance $2(p + 1)$, hence such a cycle cannot intersect the main cliques
 201 of two gadgets.

202 (iii) any tree T with two branching vertices at distance at most p_2 . Using the same argument
 203 as for the $K_{1,4}$ case, observe that the claws contained in G can only appear in the cycle
 204 cliques where the paths are attached. However, observe that these cliques are at distance
 205 $2(p + 1) > p_2$, thus, such a tree T cannot appear in G .

206

207 **3** Positive results I: disjoint union of cliques

208 For $r, q \geq 1$, let K_r^q be the disjoint union of q copies of K_r . The following proof is inspired
 209 by the case $r = 2$ by Alekseev [2].

210 **► Theorem 5.** MAXIMUM INDEPENDENT SET is FPT in K_r^q -free graphs.

211 **Proof.** We will prove by induction on q that a K_r^q -free graph has an independent set of size
 212 k or has at most $Ram(r, k)^{qk} n^{qr}$ independent sets. This will give the desired FPT-algorithm,
 213 as the proof shows how to construct this collection of independent sets. Note that the case
 214 $q = 1$ is trivial by Ramsey's theorem.

215 Let G be a K_r^q -free graph and let $<$ be any fixed total ordering of $V(G)$ such that the
 216 largest vertex in this ordering belongs to a clique of size r (the case where G is K_r -free is
 217 trivial by Ramsey's theorem). For any vertex x , define $x^+ = \{y, x < y\}$ and $x^- = V(G) \setminus x^+$.

218 Let C be a fixed clique of size r in G and let c be the largest vertex of C with respect to
 219 $<$. Let V_1 be the set of vertices of c^+ which have no neighbor in C . Note that V_1 induces a
 220 K_r^{q-1} -free graph, so by induction either it contains an independent set of size k , and so does
 221 G , or it has at most $Ram(r, k)^{(q-1)k} n^{(q-1)r}$ independent sets. In the latter case, let \mathcal{S}_1 be
 222 the set of all independent sets of $G[V_1]$.

223 Now in a second phase we define an initially empty set \mathcal{S}_C and do the following. For each
 224 independent set S_1 in \mathcal{S}_1 (including the empty set), we denote by V_2 the set of vertices in c^-
 225 that have no neighbor in S_1 . For every choice of a vertex x amongst the largest $Ram(r, k)$
 226 vertices of V_2 in the order, we add x to S_1 and modify V_2 in order to keep only vertices that
 227 are smaller than x (with respect to $<$) and non adjacent to x . We repeat this operation k
 228 times (or less if V_2 becomes empty) and, at the end, we either find an independent set of size
 229 k or add S_1 to \mathcal{S}_C . By doing so we construct a family of at most $Ram(r, k)^k$ independent
 230 sets for each S_1 , so in total we get indeed at most $Ram(r, k)^{kq} n^{(q-1)r}$ independent sets for
 231 each clique C . Finally we define \mathcal{S} as the union over all r -cliques C of the sets \mathcal{S}_C , so that \mathcal{S}
 232 has size at most the desired number.

233 We claim that if G does not contain an independent set of size k , then \mathcal{S} contains all
 234 independent sets of G . It suffices to prove that for every independent set S , there exists a
 235 clique C for which $S \in \mathcal{S}_C$. Let S be an independent set, and define C to be a clique of size
 236 r such that its largest vertex c (with respect to $<$) satisfies the conditions:

- 237 ■ no vertex of C is adjacent to a vertex of $S \cap c^+$, and
- 238 ■ c is the smallest vertex such that a clique C satisfying the first item exists.

239 First remark that such a clique always exist, since we assumed that the largest vertex c_{last}
 240 of $<$ is contained in a clique of size r , which means that $S \cap c_{last}^+$ is empty and thus the
 241 first item is vacuously satisfied. Secondly, note that several cliques C might satisfy the two
 242 previous conditions. In that case, pick one such clique arbitrarily. This definition of C and c
 243 ensures that $S \cap c^+$ is an independent set in the set V_1 defined in the construction above
 244 (it might be empty). Thus, it will be picked in the second phase as some S_1 in \mathcal{S}_1 and for
 245 this choice, each time V_2 is considered, the fact that C is chosen to minimize its largest
 246 element c guarantees that there must be a vertex of S in the $Ram(r, k)$ largest vertices in
 247 V_2 , otherwise we could find within those vertices an r -clique contradicting the choice of C .
 248 So we are insured that we will add S to the collection \mathcal{S}_C , which concludes our proof. ◀

249 4 Positive results II

250 4.1 Key ingredient: Iterative expansion and Ramsey extraction

251 In this section, we present the main idea of our algorithms. It is a generalization of iterative
 252 expansion, which itself is the maximization version of the well-known iterative compression
 253 technique. Iterative compression is a useful tool for designing parameterized algorithms for
 254 subset problems (*i.e.* problems where a solution is a subset of some set of elements: vertices
 255 of a graph, variables of a logic formula...*etc.*) [5, 17]. Although it has been mainly used for
 256 minimization problems, iterative compression has been successfully applied for maximization
 257 problems as well, under the name *iterative expansion* [4]. Roughly speaking, when the
 258 problem consists in finding a solution of size at least k , the iterative expansion technique
 259 consists in solving the problem where a solution S of size $k - 1$ is given in the input, in
 260 the hope that this solution will imply some structure in the instance. In the following, we
 261 consider an extension of this approach where, instead of a single smaller solution, one is given
 262 a set of $f(k)$ smaller solutions $S_1, \dots, S_{f(k)}$. As we will see later, we can further add more
 263 constraints on the sets $S_1, \dots, S_{f(k)}$. Notice that all the results presented in this sub-section
 264 (Lemmas 7 and 10 in particular) hold for any hereditary graph class (including the class of
 265 all graphs). The use of properties inherited from particular graphs (namely, H -free graphs in
 266 our case) will only appear in Sections 4.2 and 4.3.

267 ► **Definition 6.** For a function $f : \mathbb{N} \rightarrow \mathbb{N}$, the f -ITERATIVE EXPANSION MIS takes as
 268 input a graph G , an integer k , and a set of $f(k)$ independent sets $S_1, \dots, S_{f(k)}$, each of size
 269 $k - 1$. The objective is to find an independent set of size k in G , or to decide that such an
 270 independent set does not exist.

271 ► **Lemma 7.** (\star) Let \mathcal{G} be a hereditary graph class. MIS is FPT in \mathcal{G} iff f -ITERATIVE
 272 EXPANSION MIS is FPT in \mathcal{G} for some computable function $f : \mathbb{N} \rightarrow \mathbb{N}$.

273 We will actually prove a stronger version of this result, by adding more constraints on
 274 the input sets $S_1, \dots, S_{f(k)}$, and show that solving the expansion version on this particular
 275 kind of input is enough to obtain the result for MIS.

276 ► **Definition 8.** Given a graph G and a set of $k - 1$ vertex-disjoint cliques of G , $\mathcal{C} =$
 277 $\{C_1, \dots, C_{k-1}\}$, each of size q , we say that \mathcal{C} is a set of *Ramsey-extracted cliques of size q* if
 278 the conditions below hold. Let $C_r = \{c_j^r : j \in \{1, \dots, q\}\}$ for every $r \in \{1, \dots, k - 1\}$.

- 279 ■ For every $j \in [q]$, the set $\{c_j^r : r \in \{1, \dots, k - 1\}\}$ is an independent set of G of size $k - 1$.
- 280 ■ For any $r \neq r' \in \{1, \dots, k - 1\}$, one of the four following case can happen:
 - 281 (i) for every $j, j' \in [q]$, $c_j^r c_{j'}^{r'} \notin E(G)$
 - 282 (ii) for every $j, j' \in [q]$, $c_j^r c_{j'}^{r'} \in E(G)$ iff $j \neq j'$
 - 283 (iii) for every $j, j' \in [q]$, $c_j^r c_{j'}^{r'} \in E(G)$ iff $j < j'$
 - 284 (iv) for every $j, j' \in [q]$, $c_j^r c_{j'}^{r'} \in E(G)$ iff $j > j'$

285 In the case (i) (resp. (ii)), we say that the relation between C_r and $C_{r'}$ is *empty* (resp.
 286 *full*⁵). In case (iii) or (iv), we say the relation is *semi-full*.

287 Observe, in particular, that a set \mathcal{C} of $k - 1$ Ramsey-extracted cliques of size q can
 288 be partitionned into q independent sets of size $k - 1$. As we will see later, these cliques
 289 will allow us to obtain more structure with the remaining vertices if the graph is H -free.
 290 Roughly speaking, if q is large, we will be able to extract from \mathcal{C} another set \mathcal{C}' of $k - 1$
 291 Ramsey-extracted cliques of size $q' < q$, such that every clique is a module⁶ with respect to
 292 the solution x_1^*, \dots, x_k^* we are looking for. Then, by guessing the structure of the adjacencies
 293 between \mathcal{C}' and the solution, we will be able to identify from the remaining vertices k sets
 294 X_1, \dots, X_k , where each X_i has the same neighborhood as x_i^* w.r.t. \mathcal{C}' , and plays the role of
 295 “candidates” for this vertex. For a function $f : \mathbb{N} \rightarrow \mathbb{N}$, we define the following problem:

296 ► **Definition 9.** The f -RAMSEY-EXTRACTED ITERATIVE EXPANSION MIS problem takes
 297 as input an integer k and a graph G whose vertices are partitionned into non-empty sets
 298 $X_1 \cup \dots \cup X_k \cup C_1 \cup \dots \cup C_{k-1}$, where:

- 299 ■ $\{C_1, \dots, C_{k-1}\}$ is a set of $k - 1$ Ramsey-extracted cliques of size $f(k)$
- 300 ■ any independent set of size k in G is contained in $X_1 \cup \dots \cup X_k$
- 301 ■ $\forall i \in \{1, \dots, k\}$, $\forall v, w \in X_i$ and $\forall j \in \{1, \dots, k - 1\}$, $N(v) \cap C_j = N(w) \cap C_j = \emptyset$ or
 302 $N(v) \cap C_j = N(w) \cap C_j = C_j$
- 303 ■ the following bipartite graph \mathcal{B} is connected: $V(\mathcal{B}) = B_1 \cup B_2$, $B_1 = \{b_1^1, \dots, b_k^1\}$,
 304 $B_2 = \{b_1^2, \dots, b_{k-1}^2\}$ and $b_j^1 b_r^2 \in E(\mathcal{B})$ iff X_j and C_r are adjacent.

305 The objective is to find an independent set S in G of size at least k , or to decide that G does
 306 not contain an independent set S such that $S \cap X_i \neq \emptyset$ for all $i \in \{1, \dots, k\}$.

307 ► **Lemma 10.** Let \mathcal{G} be a hereditary graph class. If there exists a computable function
 308 $f : \mathbb{N} \rightarrow \mathbb{N}$ such that f -RAMSEY-EXTRACTED ITERATIVE EXPANSION MIS is FPT in \mathcal{G} ,
 309 then g -ITERATIVE EXPANSION MIS is FPT in \mathcal{G} , where $g(x) = \text{Ram}_\ell(f(x)2^{x(x-1)}) \forall x \in \mathbb{N}$,
 310 with $\ell_x = 2^{(x-1)^2}$.

311 **Proof.** Let $f : \mathbb{N} \rightarrow \mathbb{N}$ be such a function, and let G , k and $\mathcal{S} = \{S_1, \dots, S_{g(k)}\}$ be an input
 312 of g -ITERATIVE EXPANSION MIS. Recall that the objective is to find an independent set
 313 of size k in G , or to decide that such an independent set does not exist. If G contains
 314 an independent set of size k , then either there is one intersecting some set of \mathcal{S} , or every
 315 independent set of size k avoids the sets in \mathcal{S} . In order to capture the first case, we branch

⁵ Remark that in this case, the graph induced by $C_r \cup C_{r'}$ is the complement of a perfect matching.

⁶ A set of vertices M is a module if every vertex $v \notin M$ is adjacent to either all vertices of M , or none.

316 on every vertex v of the sets in \mathcal{S} , and make a recursive call with parameter $G \setminus N[v]$, $k - 1$.
 317 In the remainder of the algorithm, we thus assume that any independent set of size k in G
 318 avoids every set of \mathcal{S} .

319 We choose an arbitrary ordering of the vertices of each S_j . Let us denote by s_j^r the r^{th}
 320 vertex of S_j . Notice that given an ordered pair of sets of $k - 1$ vertices (A, B) , there are
 321 $\ell_k = 2^{\binom{k-1}{2}}$ possible sets of edges between these two sets. Let us denote by $c_1, \dots, c_{2^{\binom{k-1}{2}}}$
 322 the possible sets of edges, called *types*. We define an auxiliary edge-colored graph H whose
 323 vertices are in one-to-one correspondence with $S_1, \dots, S_{g(k)}$, and, for $i < j$, there is an
 324 edge between S_i and S_j of color γ iff the type of (S_i, S_j) is γ . By Ramsey's theorem, since
 325 H has $\text{Ram}_{\ell_k}(f(k)2^{k(k-1)})$ vertices, it must admit a monochromatic clique of size at least
 326 $h(k) = f(k)2^{k(k-1)}$. *W.l.o.g.*, the vertex set of this clique corresponds to $S_1, \dots, S_{h(k)}$. For
 327 $p \in \{1, \dots, k - 1\}$, let $C_p = \{s_j^p, \dots, s_{h(k)}^p\}$. Observe that the Ramsey extraction ensures
 328 that each C_p is either a clique or an independent set. If C_p is an independent set for some r ,
 329 then we can immediately conclude, since $h(k) \geq k$. Hence, we suppose that C_p is a clique for
 330 every $p \in \{1, \dots, k - 1\}$. We now prove that C_1, \dots, C_{k-1} are Ramsey-extracted cliques of
 331 size $h(k)$. First, by construction, for every $j \in \{1, \dots, h(k)\}$, the set $\{s_j^p : p = 1, \dots, k - 1\}$ is
 332 an independent set. Then, let c be the type of the clique obtained previously, represented by
 333 the adjacencies between two sets (A, B) , each of size $k - 1$. For every $p \in \{1, \dots, k - 1\}$, let
 334 a_p (resp. b_p) be the a^{th} vertex of A (resp. B). Let $p, q \in \{1, \dots, k - 1\}$, $p \neq q$. If any of $a_p b_q$ and
 335 $a_q b_p$ are edges in type c , then there is no edge between C_p and C_q , and their relation is thus
 336 empty. If both edges $a_p b_q$ and $a_q b_p$ exist in c , then the relation between C_p and C_q is full.
 337 Finally if exactly one edge among $a_p b_q$ and $a_q b_p$ exists in c , then the relation between C_p
 338 and C_q is semi-full. This concludes the fact that $\mathcal{C} = \{C_1, \dots, C_{k-1}\}$ are Ramsey-extracted
 339 cliques of size $h(k)$.

340 Suppose that G has an independent set $X^* = \{x_1^*, \dots, x_k^*\}$. Recall that we assumed
 341 previously that X^* is contained in $V(G) \setminus (C_1 \cup \dots \cup C_{k-1})$. The next step of the algorithm
 342 consists in branching on every subset of $f(k)$ indices $J \subseteq \{1, \dots, h(k)\}$, and restrict every set
 343 C_p to $\{s_j^p : j \in J\}$. For the sake of readability, we keep the notation C_p to denote $\{s_j^p : j \in J\}$
 344 (the non-selected vertices are put back in the set of remaining vertices of the graph, *i.e.*
 345 we do not delete them). Since $h(k) = f(k)2^{k(k-1)}$, there must exist a branching where the
 346 chosen indices are such that for every $i \in \{1, \dots, k\}$ and every $p \in \{1, \dots, k - 1\}$, x_i^* is either
 347 adjacent to all vertices of C_p or none of them. In the remainder, we may thus assume that
 348 such a branching has been made, with respect to the considered solution $X^* = \{x_1^*, \dots, x_k^*\}$.
 349 Now, for every $v \in V(G) \setminus (C_1, \dots, C_{k-1})$, if there exists $p \in \{1, \dots, k - 1\}$ such that
 350 $N(v) \cap C_p \neq \emptyset$ and $N(v) \cap C_p \neq C_p$, then we can remove this vertex, as we know that it
 351 cannot correspond to any x_i^* . Thus, we know that all the remaining vertices v are such that
 352 for every $p \in \{1, \dots, k - 1\}$, v is either adjacent to all vertices of C_p , or none of them.

353 In the following, we perform a color coding-based step on the remaining vertices. Informally,
 354 this color coding will allow us to identify, for every vertex x_i^* of the optimal solution, a
 355 set X_i of candidates, with the property that all vertices in X_i have the same neighborhood
 356 with respect to sets C_1, \dots, C_{k-1} . We thus color uniformly at random the remaining vertices
 357 $V(G) \setminus (C_1, \dots, C_{k-1})$ using k colors. The probability that the elements of X^* are colored
 358 with pairwise distinct colors is at least e^{-k} . We are thus reduced to the case of finding
 359 a *colorful*⁷ independent set of size k . For every $i \in \{1, \dots, k\}$, let X_i be the vertices of
 360 $V(G) \setminus (C_1, \dots, C_{k-1})$ colored with color i . We now partition every set X_i into at most
 361 2^{k-1} subsets $X_i^1, \dots, X_i^{2^{k-1}}$, such that for every $j \in \{1, \dots, 2^{k-1}\}$, all vertices of X_i^j have

⁷ A set of vertices is called *colorful* if it is colored with pairwise distinct colors.

362 the same neighborhood with respect to the sets C_1, \dots, C_{k-1} (recall that every vertex of
363 $V(G) \setminus (C_1, \dots, C_{k-1})$ is adjacent to all vertices of C_p or none, for each $p \in \{1, \dots, k-1\}$).
364 We branch on every tuple $(j_1, \dots, j_k) \in \{1, \dots, 2^{k-1}\}$. Clearly the number of branchings
365 is bounded by a function of k only and, moreover, one branching (j_1, \dots, j_k) is such that
366 x_i^* has the same neighborhood in $C_1 \cup \dots \cup C_{k-1}$ as vertices of $X_i^{j_i}$ for every $i \in \{1, \dots, k\}$.
367 We assume in the following that such a branching has been made. For every $i \in \{1, \dots, k\}$,
368 we can thus remove vertices of X_i^j for every $j \neq j_i$. For the sake of readability, we rename
369 $X_i^{j_i}$ as X_i . Let \mathcal{B} be the bipartite graph with vertex bipartition (B_1, B_2) , $B_1 = \{b_1^1, \dots, b_k^1\}$,
370 $B_2 = \{b_1^2, \dots, b_{k-1}^2\}$, and $b_i^1 b_p^2 \in E(\mathcal{B})$ iff x_i^* is adjacent to C_p . Since every x_i^* has the same
371 neighborhood as X_i with respect to C_1, \dots, C_{k-1} , this bipartite graph actually corresponds
372 to the one described in Definition 9 representing the adjacencies between X_i 's and C_p 's. We
373 now prove that it is connected. Suppose it is not. Then, since $|B_1| = k$ and $|B_2| = k-1$,
374 there must be a component with as many vertices from B_1 as vertices from B_2 . However,
375 in this case, using the fixed solution X^* on one side and an independent set of size $k-1$
376 in $C_1 \cup \dots \cup C_{k-1}$ on the other side, it implies that there is an independent set of size k
377 intersecting $\cup_{p=1}^{k-1} C_p$, a contradiction.

378 Hence, all conditions of Definition 9 are now fulfilled. It now remains to find an independent
379 set of size k disjoint from the sets \mathcal{C} , and having a non-empty intersection with X_i , for
380 every $i \in \{1, \dots, k\}$. We thus run an algorithm solving f -RAMSEY-EXTRACTED ITERATIVE
381 EXPANSION MIS on this input, which concludes the algorithm. \blacktriangleleft

382 The proof of the following result is immediate, by using successively Lemmas 7 and 10.

383 **► Theorem 11.** *Let \mathcal{G} be a hereditary graph class. If f -RAMSEY-EXTRACTED ITERATIVE*
384 *EXPANSION MIS is FPT in \mathcal{G} for some computable function f , then MIS is FPT in \mathcal{G} .*

385 We now apply this framework to two families of graphs H .

386 4.2 Clique minus a smaller clique

387 **► Theorem 12.** *(\star) For any $r \geq 2$ and $s < r$, MIS in $(K_r \setminus K_s)$ -free graphs is FPT if $s \leq 3$,*
388 *and $W[1]$ -hard otherwise.*

389 4.3 Clique minus a complete bipartite graph

390 For every three positive integers r, s_1, s_2 with $s_1 + s_2 < r$, we consider the graph $K_r \setminus K_{s_1, s_2}$.
391 Another way to see $K_r \setminus K_{s_1, s_2}$ is as a P_3 of cliques of size $s_1, r - s_1 - s_2$, and s_2 . More
392 formally, every graph $K_r \setminus K_{s_1, s_2}$ can be obtained from a P_3 by adding $s_1 - 1$ false twins of
393 the first vertex, $r - s_1 - s_2 - 1$, for the second, and $s_2 - 1$, for the third.

394 **► Theorem 13.** *$\forall r \geq 2$ and $s_1 \leq s_2$ s.t. $s_1 + s_2 < r$, MIS in $K_r \setminus K_{s_1, s_2}$ -free graphs is FPT.*

395 **Proof.** It is more convenient to prove the result for $K_{3r} \setminus K_{r,r}$ -free graphs, for any positive
396 integer r . It implies the theorem by choosing this new r to be larger than s_1, s_2 , and
397 $r - s_1 - s_2$. We will show that for $f(x) := 3r$ for every $x \in \mathbb{N}$, f -RAMSEY-EXTRACTED
398 ITERATIVE EXPANSION MIS in $K_{3r} \setminus K_{r,r}$ -free graphs is FPT. By Theorem 11, this implies
399 that MIS is FPT in this class. Let C_1, \dots, C_{k-1} (whose union is denoted by \mathcal{C}) be the
400 Ramsey-extracted cliques of size $3r$, which can be partitioned, as in Definition 9, into $3r$
401 independent sets S_1, \dots, S_{3r} , each of size $k-1$. Let $\mathcal{X} = \bigcup_{i=1}^k X_i$ be the set in which we are
402 looking for an independent set of size k . We recall that between any X_i and any C_j there are
403 either all the edges or none. Hence, the whole interaction between \mathcal{X} and \mathcal{C} can be described

404 by the bipartite graph \mathcal{B} described in Definition 9. Firstly, we can assume that each X_i is of
 405 size at least $Ram(r, k)$, otherwise we can branch on $Ram(r, k)$ choices to find one vertex in
 406 an optimum solution. By Ramsey's theorem, we can assume that each X_i contains a clique
 407 of size r (if it contains an independent set of size k , we are done). Our general strategy is
 408 to leverage the fact that the input graph is $(K_{3r} \setminus K_{r,r})$ -free to describe the structure of \mathcal{X} .
 409 Hopefully, this structure will be sufficient to solve our problem in FPT time.

410 We define an auxiliary graph Y with $k - 1$ vertices. The vertices y_1, \dots, y_{k-1} of Y
 411 represent the Ramsey-extracted cliques of \mathcal{C} and two vertices y_i and y_j are adjacent iff the
 412 relation between C_i and C_j is not empty (equivalently the relation is full or semi-full). It
 413 might seem peculiar that we concentrate the structure of \mathcal{C} , when we will eventually discard
 414 it from the graph. It is an indirect move: the simple structure of \mathcal{C} will imply that the
 415 interaction between \mathcal{X} and \mathcal{C} is simple, which in turn, will severely restrict the subgraph
 416 induced by \mathcal{X} . More concretely, in the rest of the proof, we will (1) show that Y is a clique,
 417 (2) deduce that \mathcal{B} is a complete bipartite graph, (3) conclude that \mathcal{X} cannot contain an
 418 induced $K_r^2 = K_r \uplus K_r$ and run the algorithm of Theorem 5.

419 Suppose that there is $y_{i_1}y_{i_2}y_{i_3}$ an induced P_3 in Y , and consider $C_{i_1}, C_{i_2}, C_{i_3}$ the
 420 corresponding Ramsey-extracted cliques. For $s < t \in [3r]$, let $C_i^{s \rightarrow t} := C_i \cap \bigcup_{s \leq j \leq t} S_j$.
 421 In other words, $C_i^{s \rightarrow t}$ contains the elements of C_i having indices between s and t . Since
 422 $|C_i| = 3r$, each C_i can be partitionned into three sets, of r elements each: $C_i^{1 \rightarrow r}$, $C_i^{r+1 \rightarrow 2r}$
 423 and $C_i^{2r+1 \rightarrow 3r}$. Recall that the relation between C_{i_1} and C_{i_2} (resp. C_{i_2} and C_{i_3}) is either
 424 full or semi-full, while the relation between C_{i_1} and C_{i_3} is empty. This implies that at least
 425 one of the four following sets induces a graph isomorphic to $K_{3r} \setminus K_{r,r}$:

- 426 ■ $C_{i_1}^{1 \rightarrow r} \cup C_{i_2}^{r+1 \rightarrow 2r} \cup C_{i_3}^{1 \rightarrow r}$
- 427 ■ $C_{i_1}^{1 \rightarrow r} \cup C_{i_2}^{r+1 \rightarrow 2r} \cup C_{i_3}^{2r+1 \rightarrow 3r}$
- 428 ■ $C_{i_1}^{2r+1 \rightarrow 3r} \cup C_{i_2}^{r+1 \rightarrow 2r} \cup C_{i_3}^{1 \rightarrow r}$
- 429 ■ $C_{i_1}^{2r+1 \rightarrow 3r} \cup C_{i_2}^{r+1 \rightarrow 2r} \cup C_{i_3}^{2r+1 \rightarrow 3r}$

430 Hence, Y is a disjoint union of cliques. Let us assume that Y is the union of at least two
 431 (maximal) cliques.

432 Recall that the bipartite graph \mathcal{B} is connected. Thus there is $b_h^1 \in B_1$ (corresponding to
 433 X_h) adjacent to $b_i^2 \in B_2$ and $b_j^2 \in B_2$ (corresponding to C_i and C_j , respectively), such that
 434 y_i and y_j lie in two different connected components of Y (in particular, the relation between
 435 C_i and C_j is empty). Recall that X_h contains a clique of size at least r . This clique induces,
 436 together with any r vertices in C_i and any r vertices in C_j , a graph isomorphic to $K_{3r} \setminus K_{r,r}$;
 437 a contradiction. Hence, Y is a clique.

438 Now, we can show that \mathcal{B} is a complete bipartite graph. Each X_h has to be adjacent to
 439 at least one C_i (otherwise this trivially contradicts the connectedness of \mathcal{B}). If X_h is not
 440 linked to C_j for some $j \in \{1, \dots, k - 1\}$, then a clique of size r in X_h (which always exists)
 441 induces, together with $C_i^{1 \rightarrow r} \cup C_j^{2r+1 \rightarrow 3r}$ or with $C_i^{2r+1 \rightarrow 3r} \cup C_j^{1 \rightarrow r}$, a graph isomorphic to
 442 $K_{3r} \setminus K_{r,r}$.

443 Since \mathcal{B} is a complete bipartite graph, every vertex of C_1 dominates all vertices of \mathcal{X} In
 444 particular, \mathcal{X} is in the intersection of the neighborhood of the vertices of some clique of size
 445 r . This implies that the subgraph induced by \mathcal{X} is $(K_r \uplus K_r)$ -free. Hence, we can run the
 446 FPT algorithm of Theorem 5 on this graph. ◀

447 5 Polynomial (Turing) kernels

448 In this section we investigate some special cases of Section 4.3, in particular when H is a
 449 clique of size r minus a claw with s branches, for $s < r$. Although Theorem 13 proves that

450 MIS is FPT for every possible values of r and s , we show that when $s \geq r - 2$, the problem
 451 admits a polynomial Turing kernel, while for $s \leq 2$, it admits a polynomial kernel. Notice
 452 that the latter result is somehow tight, as Corollary 18 shows that MIS cannot admit a
 453 polynomial kernel in $(K_r \setminus K_{1,s})$ -free graphs whenever $s \geq 3$.

454 ► **Theorem 14.** $(\star) \forall r \geq 2$, MIS in $(K_r \setminus K_{1,r-2})$ -free graphs has a polynomial Turing
 455 kernel.

456 ► **Theorem 15.** $(\star) \forall r \geq 3$, MIS in $(K_r \setminus K_{1,2})$ -free graphs has a kernel with $O(k^{r-1})$
 457 vertices.

458 Observe that a $(K_r \setminus K_2)$ -free graph is $(K_{r+1} \setminus K_{1,2})$ -free, hence, thus the previous result
 459 also applies to $(K_r \setminus K_2)$ -free graphs, which answers a question of [7].

460 We now focus on kernel lower bounds.

461 ► **Definition 16.** Given the graphs H, H_1, \dots, H_p , we say that (H_1, \dots, H_p) is a multipartite
 462 decomposition of H if H is isomorphic to $H_1 + \dots + H_p$. We say that (H_1, \dots, H_p) is maximal
 463 if, for every multipartite decomposition (H'_1, \dots, H'_q) of H , we have $p > q$.

464 It can easily be seen that for every graph H , a maximal multipartite decomposition of H
 465 is unique. We have the following:

466 ► **Theorem 17.** (\star) Let H be any fixed graph, and let $H = H_1 + \dots + H_p$ be the maximal
 467 multipartite decomposition of H . If, for some $i \in [p]$, MIS is NP-hard in H_i -free graphs,
 468 then MIS does not admit a polynomial kernel in H -free graphs unless $NP \subseteq coNP/poly$.

469 The next results shows that the polynomial kernel obtained in the previous section for
 470 $(K_r \setminus K_{1,s})$ -free graphs, $s \leq 2$, is somehow tight.

471 ► **Corollary 18.** (\star) For $r \geq 4$, and every $3 \leq s \leq r - 1$, MIS in $(K_r \setminus K_{1,s})$ -free graphs
 472 does not admit a polynomial kernel unless $NP \subseteq coNP/poly$.

473 We conjecture that Theorem 17 actually captures all possible negative cases concerning
 474 the kernelization of the problem. Informally speaking, our intuition is the natural idea that
 475 the join operation between graphs seems the only way to obtain $\alpha(G) = O(\max_{i=1, \dots, t} \alpha(G_i))$,
 476 which is the main ingredient of OR-compositions.

477 6 Conclusion and open problems

478 We started to unravel the FPT/ $W[1]$ -hard dichotomy for MIS in H -free graphs, for a fixed
 479 graph H . At the cost of one reduction, we showed that it is $W[1]$ -hard as soon as H is not
 480 chordal, even if we simultaneously forbid induced $K_{1,4}$ and trees with at least two branching
 481 vertices. Tuning this construction, it is also possible to show that if a connected H is not
 482 roughly a "path of cliques" or a "subdivided claw of cliques", then MIS is $W[1]$ -hard.

483 An interesting open problem is the case when H is the *cricket*, that is a triangle with
 484 two pending vertices, each attached to a different vertex

485 For disconnected graphs H , we obtained an FPT algorithm when H is a cluster (*i.e.*, a
 486 disjoint union of cliques). We conjecture that, more generally, the disjoint union of two easy
 487 cases is an easy case; formally, *if MIS is FPT in G -free graphs and in H -free graphs, then it*
 488 *is FPT in $G \uplus H$ -free graphs.*

489 A natural question regarding our two FPT algorithms of Section 4 concerns the existence
 490 of polynomial kernels. In particular, we even do not know whether the problem admits a
 491 kernel for very simple cases, such as when $H = K_5 \setminus K_3$ or $H = K_5 \setminus K_{2,2}$.

492 A more anecdotal conclusion is the fact that the parameterized complexity of the problem
 493 on H -free graphs is now complete for every graph H on four vertices, including concerning
 494 the polynomial kernel question, whereas the $FPT/W[1]$ -hard question remains open for only
 495 five graphs H on five vertices.

 496 ——— **References** ———

- 497 **1** V. Alekseev. The effect of local constraints on the complexity of determination of the
 498 graph independence number. *Combinatorial-Algebraic Methods in Applied Mathematics*,
 499 pages 3–13, 1982. in Russian.
- 500 **2** V. E. Alekseev. On the number of maximal independent sets in graphs from hereditary
 501 classes. *Combinatorial-Algebraic Methods in Discrete Optimization*, pages 5–8, 1991. (In
 502 Russian).
- 503 **3** G. Bacsó, D. Lokshtanov, D. Marx, M. Pilipczuk, Z. Tuza, and E. Jan van Leeuwen.
 504 Subexponential-time algorithms for maximum independent set in P_t -free and broom-
 505 free graphs. *CoRR*, abs/1804.04077, 2018.
- 506 **4** J. Chen, Y. L., S. Lu, S. S., and F. Zhang. Iterative expansion and color coding: An
 507 improved algorithm for 3d-matching. *ACM Trans. Algorithms*, 8(1):6:1–6:22, 2012.
- 508 **5** M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk,
 509 and S. Saurabh. *Parameterized Algorithms*. Springer, 2015.
- 510 **6** K. Dabrowski. *Structural Solutions to Maximum Independent Set and Related Problems*.
 511 PhD thesis, University of Warwick, 2012.
- 512 **7** K. Dabrowski, V. V. Lozin, H. Müller, and D. Rautenbach. Parameterized complexity
 513 of the weighted independent set problem beyond graphs of bounded clique number. *J.*
 514 *Discrete Algorithms*, 14:207–213, 2012.
- 515 **8** R. Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics*.
 516 Springer, 2012.
- 517 **9** R. G. Downey and M. R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in
 518 Computer Science. Springer, 2013.
- 519 **10** H. Perret du Cray and I. Sau. Improved FPT algorithms for weighted independent set in
 520 bull-free graphs. *Discrete Mathematics*, 341(2):451–462, 2018.
- 521 **11** M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of*
 522 *NP-Completeness*. W. H. Freeman, 1979.
- 523 **12** A. Grzesik, T. Klimosova, M. Pilipczuk, and M. Pilipczuk. Polynomial-time algorithm for
 524 maximum weight independent set on P_6 -free graphs. *CoRR*, abs/1707.05491, 2017.
- 525 **13** D. Hermelin, M. Mnich, and E. J. van Leeuwen. Parameterized complexity of induced
 526 graph matching on claw-free graphs. *Algorithmica*, 70(3):513–560, 2014.
- 527 **14** T. Karthick. Independent sets in some classes of $S_{i,j,k}$ -free graphs. *J. Comb. Optim.*,
 528 34(2):612–630, August 2017.
- 529 **15** T. Karthick and F. Maffray. Maximum weight independent sets in classes related to claw-
 530 free graphs. *Discrete Applied Mathematics*, 216:233 – 239, 2017.
- 531 **16** D. Lokshtanov, M. Vatshelle, and Y. Villanger. Independent set in P_5 -free graphs in
 532 polynomial time. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on*
 533 *Discrete Algorithms, SODA 2014*, pages 570–581, 2014.
- 534 **17** B. Reed, K. Smith, and A. Vetta. Finding odd cycle transversals. *Operations Research*
 535 *Letters*, 32(4):299 – 301, 2004.
- 536 **18** S. Thomassé, N. Trotignon, and K. Vuskovic. A polynomial turing-kernel for weighted
 537 independent set in bull-free graphs. *Algorithmica*, 77(3):619–641, 2017.
- 538 **19** D. Zuckerman. Linear degree extractors and the inapproximability of max clique and
 539 chromatic number. *Theory of Computing*, 3(1):103–128, 2007.