Parameterized Lower Bounds and Dichotomy Results for the NP-completeness of *H*-free Edge Modification Problems

N. R. Aravind[1], R. B. Sandeep[1⋆], and Naveen Sivadasan[2]

[1] Department of Computer Science & Engineering
Indian Institute of Technology Hyderabad, India
{aravind,cs12p0001}@iith.ac.in
[2] TCS Innovation Labs, Hyderabad, India
naveen@atc.tcs.com

Abstract. For a graph H, the H-free Edge Deletion problem asks whether there exist at most k edges whose deletion from the input graph G results in a graph without any induced copy of H. H-free Edge Completion and H-free Edge Editing are defined similarly where only completion (addition) of edges are allowed in the former and both completion and deletion are allowed in the latter. We completely settle the classical complexities of these problems by proving that H-free Edge Deletion is NP-complete if and only if H is a graph with at least two edges, H-free Edge Completion is NP-complete if and only if H is a graph with at least two non-edges and H-free Edge Editing is NP-complete if and only if H is a graph with at least three vertices. Additionally, we prove that, these NP-complete problems cannot be solved in parameterized subexponential time, i.e., in time $2^{o(k)} \cdot |G|^{O(1)}$, unless Exponential Time Hypothesis fails. Furthermore, we obtain implications on the incompressibility of these problems.

1 Introduction

Edge modification problems are to test whether modifying at most k edges makes the input graph satisfy certain properties. The three major edge modification problems are edge deletion, edge completion and edge editing problems. In edge deletion problems we are allowed to delete at most k edges from the input graph. Similarly, in completion problems, it is allowed to complete (add) at most k edges and in editing problems at most k editing (deletion or completion) are allowed. Edge modification problems comes under the broader category of graph modification problems which have found applications in DNA physical mapping [11], numerical algebra [14], circuit design [9] and machine learning [2].

The focus of this paper is on H-free edge modification problems, in which we are allowed to modify at most k edges to make the input graph devoid of any induced copy of H, where H is any fixed graph. Though these problems

⋆ supported by TCS Research Scholarship

have been studied for four decades, a complete dichotomy result on the classical complexities of these problems are not yet found. We settle this by proving that H-free Edge Deletion is NP-complete if and only if H is a graph with at least two edges, H-free Edge Completion is NP-complete if and only if H is a graph with at least two non-edges and H-free Edge Editing is NP-complete if and only if H is a graph with at least three vertices. As a bonus, we obtain the parameterized lower bounds for these NP-complete problems. We obtain that these NP-complete problems cannot be solved in parameterized subexponential time (i.e., in time $2^{o(k)} \cdot |G|^{O(1)}$), unless Exponential Time Hypothesis (ETH) fails, where ETH is a widely believed complexity theoretic assumption. Furthermore, we obtain implications on the incompressibility (non-existence of polynomial kernels) of these problems.

We build on our recent paper [1], in which we proved that H-free Edge Deletion is NP-complete if H has at least two edges and has a component with maximum number of vertices which is a tree or a regular graph. We also proved that these problems cannot be solved in parameterized subexponential time, unless ETH fails.

Related Work: In 1981, Yannakakis proved that H-free Edge Deletion is NP-complete if H is a cycle [16]. Later in 1988, El-Mallah and Colbourn proved that the problem is NP-complete if H is a path of at least two edges [9]. Addressing the fixed parameter tractability of a generalized version of these problems, Cai proved that [4] H-free Edge Deletion, Completion and Editing are fixed parameter tractable, i.e., they can be solved in time $f(k) \cdot |G|^{O(1)}$, for some function f. Polynomial kernelizability of these problems have been studied widely. Given an instance (G, k) of the problem the objective is to obtain in polynomial time an equivalent instance of size polynomial in k. Kratsch and Wahlström gave the first result on the incompressibility of H-free edge modification problems. They proved that [13] for a certain graph H on seven vertices, H-free Edge Deletion and H-free Edge Editing do not admit polynomial kernels, unless NP $\subseteq$ coNP/poly. They use polynomial parameter transformation from an NP-complete problem and hence their results imply the NP-completeness of these problems. Later, Cai and Cai proved that H-free Edge Editing, Deletion and Completion do not admit polynomial kernels if H is a path or a cycle with at least four edges, unless NP $\subseteq$ coNP/poly [5]. Further, they proved that H-free Edge Editing and Deletion are incompressible if H is 3-connected but not complete, and H-free Edge Completion is incompressible if H is 3-connected and has at least two non-edges, unless NP $\subseteq$ coNP/poly [5]. Under the same assumption, it is proved that H-free Edge Deletion and H-free Edge Completion are incompressible if H is a tree on at least 7 vertices, which is not a star graph and H-free Edge Deletion is incompressible if H is the star graph $K_{1,s}$, where $s \geq 10$ [6]. They also use polynomial parameter transformations and hence these problems are NP-complete.

Outline of the Paper: Section 2 gives the notations and terminology used in the paper. It also introduces a construction which is a modified version of the main construction used in [1]. Section 3 settles the case of H-FREE EDGE EDITING. Section 4 obtains results for H-FREE EDGE DELETION and COMPLETION. In the concluding section, we discuss the implications of our results on the incompressibility of H-free edge modification problems.

2 Preliminaries and Basic Tools

Graphs: For a graph G, $V(G)$ denotes the vertex set and $E(G)$ denotes the edge set. We denote the symmetric difference operator by $\triangle$, i.e., for two sets F and F', $F \triangle F' = (F \setminus F') \cup (F' \setminus F)$. For a graph G and a set $F \subseteq [V(G)]^2$, $G \triangle F$ denotes the graph $(V(G), E(G) \triangle F)$. A component of a graph is largest if it has maximum number of vertices. By $|G|$ we denote $|V(G)| + |E(G)|$. The disjoint union of two graphs G and G' is denoted by $G \cup G'$ and the disjoint union of t copies of G is denoted by tG. A simple path on t vertices is denoted by P_t. The graph t-diamond is $K_2 + tK_1$, the join of K_2 and tK_1. Hence, 2-diamond is the diamond graph. The minimum degree of a graph G is denoted by $\delta(G)$ and the maximum degree is denoted by $\Delta(G)$. Degree of a vertex v in a graph G is denoted by $\deg_G(v)$. We remove the subscript when there is no ambiguity. We denote the complement of a graph G by $\overline{G}$. For a graph H and a vertex set $V' \subseteq V(H)$, $H[V']$ is the graph induced by V' in H. A null graph is a graph without any edge.

For integers ℓ and h such that $h > \ell$, (ℓ, h)-degree graph is a graph in which every vertex has degree either ℓ or h. The set of vertices with degree ℓ is denoted by V_ℓ and the set of vertices with degree h is denoted by V_h. An (ℓ, h)-degree graph is called *sparse* if V_l induces a graph with at most one edge and V_h induces a graph with at most one edge.

The context determines whether H-FREE EDGE DELETION denotes the classical problem or the parameterized problem. This applies to COMPLETION and EDITING problems. For the parameterized problems, we use k (the size of the solution being sought) as the parameter. In this paper, edge modification implies either deletion, completion or editing.

Technique for Proving Parameterized Lower Bounds: Exponential Time Hypothesis (ETH) is a widely believed complexity theoretic assumption that 3-SAT cannot be solved in time $2^{o(n)}$, where n is the number of variables in the 3-SAT instance. A linear parameterized reduction is a polynomial time reduction from a parameterized problem A to a parameterized problem B such that for every instance (G, k) of A, the reduction gives an instance (G', k') such that $k' = O(k)$. The following result helps us to obtain parameterized lower bound under ETH.

Proposition 2.1 ([7]). *If there is a linear parameterized reduction from a parameterized problem A to a parameterized problem B and if A does not admit a parameterized subexponential time algorithm, then B does not admit a parameterized subexponential time algorithm.*

Two parameterized problems A and B are linear parameter equivalent if there is a linear parameterized reduction from A to B and there is a linear parameterized reduction from B to A. We refer the book [7] for various aspects of parameterized algorithms and complexity. The following are some folklore observations.

Proposition 2.2. *H-free Edge Deletion and $\overline{H}$-free Edge Completion are linear parameter equivalent. Similarly, H-free Edge Editing and $\overline{H}$-free Edge Editing are linear parameter equivalent.*

Proposition 2.3. *(i) H-free Edge Deletion is NP-complete if and only if $\overline{H}$-free Edge Completion is NP-complete. Furthermore, H-free Edge Deletion cannot be solved in parameterized subexponential time if and only if $\overline{H}$-free Edge Completion cannot be solved in parameterized subexponential time.*

(ii) H-free Edge Editing is NP-complete if and only if $\overline{H}$-free Edge Editing is NP-complete. Furthermore, H-free Edge Editing cannot be solved in parameterized subexponential time if and only if $\overline{H}$-free Edge Editing cannot be solved in parameterized subexponential time.

Proposition 2.4. *(i) H-free Edge Deletion is polynomial time solvable if H is a graph with at most one edge.*

(ii) H-free Edge Completion is polynomial time solvable if H is a graph with at most one non-edge.

(iii) H-free Edge Editing is polynomial time solvable if H is a graph with at most two vertices.

In this paper, we prove that these are the only polynomial time solvable H-free edge modification problems. For any fixed graph H, the H-free edge modification problems trivially belong to NP. Hence, we may state that these problems are NP-complete by proving their NP-hardness.

2.1 Basic Tools

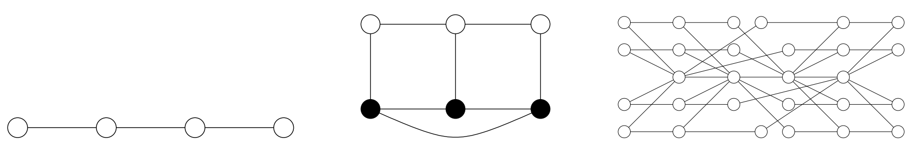
The following construction is a slightly modified version of the main construction used in [1]. The modification is done to make it work for reductions of Completion and Editing problems. The input of the construction is a tuple (G', k, H, V'), where G' and H are graphs, k is a positive integer and $V' \subseteq V(H)$. In the old construction (Construction 1 in [1]), for every copy C of $H[V']$ in G', we introduced $k+1$ copies of H such that the intersection of every pair of them is C. In the modified construction given below, we do the same for every copy C of $H[V']$ on a complete graph on $V(G')$.

Construction 1 *Let (G', k, H, V') be an input to the construction, where G' and H are graphs, k is a positive integer and V' is a subset of vertices of H. Label the vertices of H such that every vertex gets a unique label. Let the labelling be ℓ_H. Consider a complete graph K' on $V(G')$. For every subgraph (not necessarily induced) C with a vertex set $V(C)$ and an edge set $E(C)$ in K' such that C is isomorphic to $H[V']$, do the following:*

– *Give a labelling ℓ_C for the vertices in C such that there is an isomorphism f between C and $H[V']$ which maps every vertex v in C to a vertex v' in $H[V']$ such that $\ell_C(v) = \ell_H(v')$, i.e., $f(v) = v'$ if and only if $\ell_C(v) = \ell_H(v')$.*
– *Introduce $k+1$ sets of vertices $V_1, V_2, \ldots, V_{k+1}$, each of size $|V(H) \setminus V'|$.*
– *For each set V_i, introduce an edge set E_i of size $|E(H) \setminus E(H[V'])|$ among $V_i \cup V(C)$ such that there is an isomorphism h between H and $(V(C) \cup V_i, E(C) \cup E_i)$ which preserves f, i.e., for every vertex $v \in V(C)$, $h(v) = f(v)$.*

This completes the construction. Let the constructed graph be G.

We remark that the complete graph K' on $V(G')$ is not part of the constructed graph. The complete graph is only used to find where we need to introduce new vertices and edges. An example of the construction is shown in Figure 1. We use the terminology used in [1]. We repeat it here for convenience. Let C be a copy of $H[V']$ in K'. Then, C is called a *base*. Let $\{V_i\}$ be the $k+1$ sets of vertices introduced in the construction for the base C. Then, each V_i is called a *branch* of C and the vertices in V_i are called the *branch vertices* of C. If V_j is a branch of C, then the vertex set of C is denoted by B_j. The vertex set of G' in G is denoted by $V_{G'}$. The copy of H formed by V_j, E_j and C is denoted by H_j. Since H is a fixed graph, the construction runs in polynomial time. The following two Lemmas are the generalized version of Lemma 2.3 and 3.5 of [1].



(a) G'

(b) H. The vertices in V' are blackened.

(c) Output of Construction 1 with an input $(G', k = 1, H, V')$.

Fig. 1: An example of Construction 1

Lemma 2.5. *Let G be obtained by Construction 1 on the input (G', k, H, V'), where G' and H are graphs, k is a positive integer and $V' \subseteq V(H)$. Then, if (G, k) is a yes-instance of H-FREE EDGE EDITING (DELETION/COMPLETION), then (G', k) is a yes-instance of H'-FREE EDGE EDITING (DELETION/COMPLETION), where H' is $H[V']$.*

Proof. Let F be a solution of size at most k of (G, k). For a contradiction, assume that $G' \triangle F$ has an induced H' with a vertex set U. Hence there is a base C in G' isomorphic to H' with the vertex set $V(C) = U$. Since there are $k+1$ copies of H in G, where each pair of copies of H has the intersection C, and $|F| \leq k$, operating with F cannot kill all the copies of H associated with C. Therefore, since U induces an H' in $G' \triangle F$, there exists a branch V_i of C such that $U \cup V_i$ induces H in $G \triangle F$, which is a contradiction. $\square$

Lemma 2.6. *Let H be any graph and d be any integer. Let V' be the set of vertices in H with degree more than d. Let H' be $H[V']$. Then, there is a linear parameterized reduction from H'-*FREE EDGE EDITING *(*DELETION/COMPLETION*) to H-*FREE EDGE EDITING *(*DELETION/COMPLETION*).*

Proof. Let (G', k) be an instance of H'-FREE EDGE EDITING (DELETION/COMPLETION). Apply Construction 1 on (G', k, H, V') to obtain G. We claim that (G', k) is a yes-instance of H'-FREE EDGE EDITING (DELETION/COMPLETION) if and only if (G, k) is a yes-instance of H-FREE EDGE EDITING (DELETION/COMPLETION).

Let F' be a solution of size at most k of (G', k). For a contradiction, assume that $G \triangle F'$ has an induced H with a vertex set U. Since a branch vertex has degree at most d, every vertex in U with degree more than d in $(G \triangle F')[U]$ must be from $V_{G'}$. Hence there is an induced H' in $G' \triangle F'$, which is a contradiction. Lemma 2.5 proves the converse. □

3 *H*-FREE EDGE EDITING

In this section, we prove that H-FREE EDGE EDITING is NP-complete if and only if H is a graph with at least three vertices. We also prove that these problems cannot be solved in parameterized subexponential time unless ETH fails. We use the following known results.

Proposition 3.1. *The following problems are* NP-complete*. Furthermore, they cannot be solved in time $2^{o(k)} \cdot |G|^{O(1)}$, unless ETH fails.*

(i) P_3-FREE EDGE EDITING *[12].*
(ii) P_4-FREE EDGE EDITING *[Follows from the proof of the lower bound of* $\{C_4, P_4\}$-FREE EDGE EDITING *in [8][3]].*
(iii) C_ℓ-FREE EDGE EDITING*, for any fixed $l \geq 3$ [Follows from the proof for the corresponding* DELETION *problems in [16]].*
(iv) $2K_2$-FREE EDGE EDITING *[(iii) and Proposition 2.3(ii)].*
(v) DIAMOND-FREE EDGE EDITING *[3].*

In our previous work [1], we proved that R-FREE EDGE DELETION is NP-complete if R is a regular graph with at least two edges. We also proved that these NP-complete problems cannot be solved in parameterized subexponential time, unless ETH fails. We observe that the results for R-FREE EDGE DELETION follows for R-FREE EDGE EDITING as well. The proofs are very similar except that we use Construction 1 instead of its ancestor in [1] and we reduce from EDITING problems instead of DELETION problems. We can use P_3-FREE EDGE EDITING, C_ℓ-FREE EDGE EDITING and $2K_2$-FREE EDGE EDITING as the base cases instead of their DELETION counterparts. We skip the proof as it will be a repetition of that in [1].

[3] We thank Pål Grønås Drange for pointing out this and sharing a complete proof of the same.

Lemma 3.2. *Let R be a regular graph with at least two edges. Then R-*FREE EDGE EDITING *is* NP-complete. *Furthermore, the problem cannot be solved in time $2^{o(k)} \cdot |G|^{O(1)}$, unless ETH fails.*

Now, we strengthen the above lemma by proving the same results for all regular graphs with at least three vertices.

Lemma 3.3. *Let R be a regular graph with at least three vertices. Then R-*FREE EDGE EDITING *is* NP-complete. *Furthermore, the problem cannot be solved in time $2^{o(k)} \cdot |G|^{O(1)}$, unless ETH fails.*

Proof. If R has at least two edges then the statements follows from Lemma 3.2. Assume that R has at most one edge and at least three vertices. It is straight-forward to see that R must be the null graph. Then the complement of R is a complete graph with at least two edges. Now, the statements follows from Proposition 2.3(ii) and Lemma 3.2. □

Having these results in hand, we use Lemma 2.6 to prove the dichotomy result and the parameterized lower bound of H-FREE EDGE EDITING. Given a graph H with at least three vertices, we introduce a method Editing-Churn(H) to obtain a graph H' such that there is a linear parameterized reduction from H'-FREE EDGE EDITING to H-FREE EDGE EDITING and H' is a graph with at least three vertices and is a regular graph or a P_3 or a P_4 or a diamond.

Editing-Churn(H)
H is a graph with at least three vertices.

Step 1: If H is a regular graph, a P_3, a P_4 or a diamond, then return H.
Step 2: If H is a graph in which the number of vertices with degree more than $\delta(H)$ is at most two, then let $H = \overline{H}$ and goto Step 1.
Step 3: Delete all vertices with degree $\delta(H)$ in H and go to Step 1.

Observation 3.4 *Let H be a graph with at least three vertices. Then Editing-Churn(H) returns a graph H' which has at least three vertices and is a regular graph or a P_3 or a P_4 or a diamond. Furthermore, there is a linear parameterized reduction from H'-*FREE EDGE EDITING *to H-*FREE EDGE EDITING.

Proof. At any stage of the method, we make sure that the graph has at least three vertices. Let H' be an intermediate graph obtained in the method such that it is neither a regular graph nor a P_3 nor a P_4 nor a diamond. If Step 2 is applicable to both H' and $\overline{H'}$, then H hat at most four vertices. Hence H has either three or four vertices. It is straight-forward to verify that a graph (with three or four vertices) or its complement, satisfying the condition in Step 2, is either a regular graph or a P_3 or a P_4 or a diamond, which is a contradiction. The linear parameterized reduction from H'-FREE EDGE EDITING to H-FREE EDGE EDITING follows from Proposition 2.3(ii) and Lemma 2.6. □

Theorem 3.5. *H-free Edge Editing is* NP-complete *if and only if H is a graph with at least three vertices. Furthermore, these* NP-complete *problems cannot be solved in time $2^{o(k)} \cdot |G|^{O(1)}$, unless ETH fails.*

Proof. If H is a graph with at most two vertices, the statements follows from Proposition 2.4(iii). Let H be a graph with at least three vertices. Let H' be the graph returned by Editing-Churn(H). By Observation 3.4, H' is either a regular graph or a P_3 or a P_4 or a diamond and there is a linear parameterized reduction from H'-free Edge Editing to H-free Edge Editing. Now, the statements follows from the lower bound results for these graphs (3.1(i), (ii), (v) and Lemma 3.3). □

4 $\boldsymbol{H}$-free Edge Deletion

In this section, we prove that H-free Edge Deletion is NP-complete if and only if H is a graph with at least two edges. We also prove that these NP-complete problems cannot be solved in parameterized subexponential time, unless ETH fails. Then, from Proposition 2.3(i), we obtain a dichotomy result for H-free Edge Completion. We apply a technique similar to that we applied for Editing in the last section.

Proposition 4.1. *The following problems are* NP-complete*. Furthermore, they cannot be solved in time $2^{o(k)} \cdot |G|^{O(1)}$, unless ETH fails.*

(i) P_3-free Edge Deletion *[12].*
(ii) Diamond-free Edge Deletion *[10, 15].*
(iii) H-free Edge Deletion, *if H is a graph with at least two edges and has a largest component which is a regular graph or a tree [1].*

The following Lemma is a consequence of Lemma 2.6 and Proposition 2.3(i).

Lemma 4.2. *Let H be any graph. Then the following hold true:*

*(i) Let H' be the subgraph of H obtained by removing all vertices with degree $\delta(H)$. Then there is a linear parameterized reduction from H'-*free Edge Deletion *to H-*free Edge Deletion.
*(ii) Let H' be the subgraph of H obtained by removing all vertices with degree $\Delta(H)$. Then there is a linear parameterized reduction from H'-*free Edge Deletion *to H-*free Edge Deletion.

Proof. The first part directly follows from Lemma 2.6 by setting $d = \delta(H)$. To prove the second part, consider the problem $\overline{H}$-free Edge Completion. Let H'' be the graph obtained by removing all vertices with degree $\delta(\overline{H})$ from $\overline{H}$. Now, by Lemma 2.6, there is a linear parameterized reduction from H''-free Edge Completion to $\overline{H}$-free Edge Completion. We observe that H'' is $\overline{H'}$. Hence, by Proposition 2.3(i), there is a linear parameterized reduction from H'-free Edge Deletion to H-free Edge Deletion. □

Given a graph H, we keep on deleting either the minimum degree vertices or the maximum degree vertices by making sure that the resultant graph has at least two edges. We do this process until we obtain a graph in which vertices with degree more than $\delta(H)$ induces a graph with at most one edge and vertices with degree less than $\Delta(H)$ induces a graph with at most one edge. We call this method Deletion-Churn.

Deletion-Churn(H)
H is a graph with at least two edges.

Step 1: If H is a graph in which the vertices with degree more than $\delta(H)$ induces a subgraph with at most one edge and the vertices with degree less than $\Delta(H)$ induces a subgraph with at most one edge, then return H.
Step 2: If H is a graph in which the vertices with degree more than $\delta(H)$ induces a subgraph with at least two edges, then delete all vertices with degree $\delta(H)$ from H and goto Step 1.
Step 3: If H is a graph in which the vertices with degree less than $\Delta(H)$ induces a subgraph with at least two edges, then delete all vertices with degree $\Delta(H)$ from H. Goto Step 1.

Observation 4.3 *Let H be a graph with at least two edges. If the vertices with degree more than $\delta(H)$ induces a graph with at most one edge and the vertices with degree less than $\Delta(H)$ induces a graph with at most one edge, then H is either regular graph or a forest or a sparse (ℓ, h)-degree graph.*

Proof. Assume that H is not a regular graph. Since H has at least two edges and it satisfies the premises, $\delta(H) \geq 1$. If $\delta(H) = 1$, the premises imply that H is a forest. Assume that $\delta(H) \geq 2$. Then we prove that H is a sparse (ℓ, h)-degree graph. For a contradiction, assume that there exists a vertex $v \in V(H)$ such that $\delta(H) < deg(v) < \Delta(H)$. The premises imply that v has degree at most two, which is a contradiction. □

Lemma 4.4. *Let H be a graph with at least two edges. Then Deletion-Churn(H) returns a graph H' such that:*

(i) There is a linear parameterized reduction from H'-free Edge Deletion to H-free Edge Deletion.
(ii) H' has at least two edges and is either a regular graph or a forest or a sparse (ℓ, h)-degree graph.

Proof. In every step, we make sure that there are at least two edges in the resultant graph. Now, the first part follows from Lemma 4.2 and the second part follows from Observation 4.3. □

If the output of Deletion-Churn(H), H' is a regular graph or a forest, we obtain from Proposition 4.1(iii) that H-free Edge Deletion is NP-complete and cannot be solved in parameterized subexponential time, unless ETH fails. Therefore, the only graphs to be handled now is the sparse (ℓ, h)-degree graphs with at least two edges. We do that in the next two subsections.

4.1 t-diamond-free Edge Deletion

We recall that t-diamond is the graph $K_2 + tK_1$ and that 2-diamond is the diamond graph. Clearly, t-diamond is a sparse (ℓ, h)-degree graph. In this subsection, we prove that t-diamond-free Edge Deletion is NP-complete. Further, we prove that the problem cannot be solved in parameterized subexponential time, unless ETH fails. We use an inductive proof where the base case is Diamond-free Edge Deletion. For the proof, we introduce a simple construction, which is given below.

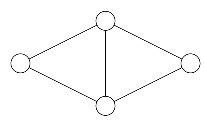


Fig. 2: A 2-diamond is isomorphic to a diamond graph.

Construction 2 *Let (G', k) be an input to the construction. For every edge $\{u, v\}$ in G', introduced a clique $C_{\{u,v\}}$ of $k+1$ vertices such that every vertex in $C_{\{u,v\}}$ is adjacent to both u and v. This completes the construction. Let G be the resultant graph.*

Lemma 4.5. *For any $t \geq 2$, t-diamond-free Edge Deletion is* NP-complete. *Furthermore, the problem cannot be solved in time $2^{o(k)} \cdot |G|^{O(1)}$, unless ETH fails.*

Proof. The proof is by induction on t. If $t = 2$, the problem is Diamond-free Edge Deletion and the theorem follows from Proposition 4.1(ii). Assume that $t \geq 3$ and that the statements hold true for $t - 1$. We give a reduction from $(t-1)$-diamond-free Edge Deletion to t-diamond-free Edge Deletion.

Let (G', k) be an instance of $(t-1)$-diamond-free Edge Deletion. Apply Construction 2 on (G', k) to obtain G. We claim that (G', k) is a yes-instance of $(t-1)$-diamond-free Edge Deletion if and only if (G, k) is a yes-instance of t-diamond-free Edge Deletion.

Let (G', k) be a yes-instance of $(t-1)$-diamond-free Edge Deletion. Let F' be a solution of size at most k of (G', k). We claim that F' is a solution of (G, k). For a contradiction, assume that $G - F'$ has an induced t-diamond on a vertex set $U \subseteq V(G)$. Let x and y be the $(t+1)$-degree vertices in the t-diamond induced by U in $G - F'$. Now there are three cases to be considered.

Case 1: Both x and y are from a clique $C_{\{u,v\}}$ introduced in the construction.

We note that x and y are adjacent to u and v and all other vertices in $C_{\{u,v\}}$. Hence the common neighborhood of x and y does not have an independent set of size at least 3, which is a contradiction.

Case 2: Let x is from a clique $C_{\{u,v\}}$ introduced in the construction and y be u.

The common neighborhood of x and y does not have an independent set of size at least 3, which is a contradiction.

Case 3: Both x and y are from G'. The common neighborhood of x and y in $G-F'$ is constituted by $C_{\{x,y\}}$ and the common neighbors of x a and y in $G'-F'$. Since $C_{\{x,y\}}$ is a clique, it can contribute at most one to the independent set of the common neighborhood of x and y. Hence, there should be an independent set of size at least $t-1$ in the common neighborhood of x and y in $G'-F'$. Since $G'-F'$ is $(t-1)$-diamond-free, this is a contradiction.

Conversely, let F be a solution of size at most k of (G,k). We prove that $G'-F$ is $(t-1)$-diamond-free. For a contradiction, assume that $G'-F$ has an induced $(t-1)$-diamond on a vertex set $U \subseteq V(G')$. Let x and y be the t-degree vertices of the $(t-1)$-diamond induced by U in $G'-F$. Since there are $k+1$ common neighbors of x and y ($C_{\{x,y\}}$) introduced by the construction, there exists a common neighbor $z \in C_{\{x,y\}}$ such that $U \cup \{z\}$ induces a t-diamond in $G-F'$, which is a contradiction. □

4.2 Handling sparse (ℓ, h)-degree graphs

We recall that for $h > \ell$, every vertex of a sparse (ℓ,h)-degree graph H is either of degree ℓ or of degree h and that V_ℓ induces a graph with at most one edge and V_h induces a graph with at most one edge. We have already handled t-diamond graphs. We handle the rest of the sparse (ℓ,h)-degree graphs in this subsection. Let H be any sparse (ℓ,h)-graph. There are four cases to be handled:

Case 1: V_h is an independent set; V_ℓ is an independent set
Case 2: V_h induces a graph with one edge; V_ℓ is an independent set
Case 3: V_h is an independent set; V_ℓ induces a graph with one edge
Case 4: V_h induces a graph with one edge; V_ℓ induces a graph with one edge

Observation 4.6 *Let H be a sparse (ℓ,h)-graph with at least two edges. Then the following hold true:*

(i) If $\ell = 1$, then H is a forest.
(ii) If $\ell \geq 2$, then $|V_\ell| \geq 2$ and the equality holds only when H is a diamond.

Proof. To prove the first part, we observe that $H \setminus V_\ell$ has at most one edge. To prove the second part, we observe that if $|V_\ell| \leq 2$ and if H is not a diamond, then $h \leq \ell$, which is a contradiction. □

Since the case of forest is already handled in Proposition 4.1(iii), we can safely assume that $\ell \geq 2$ and hence $h \geq 3$. We start with handling Case 1. We use a slightly modified version of Construction 1. We recall that, in Construction 1, with an input (G', k, H, V'), For every copy C of $H[V']$ in K' (a complete graph on $V(G')$), we introduced $k+1$ branches such that each branch along with C form a copy of H. In the modified construction, in addition to this, we make every pair of vertices from different branches mutually adjacent.

Construction 3 *Let (G', k, H, V') be an input to the construction, where G' and H are graphs, k is a positive integer and V' is a subset of vertices of H. Apply Construction 1 on (G', k, H, V') to obtain G''. For every pair of vertices $\{v_i, v_j\}$ such that $v_i \in V_i$ and $v_j \in V_j$, where $i \neq j$, make v_i and v_j adjacent. This completes the construction. Let the constructed graph be G.*

Now, we have a lemma similar to Lemma 2.5. We skip the proof as it is quite similar to that of Lemma 2.5.

Lemma 4.7. *Let G be obtained by Construction 3 on the input (G', k, H, V'), where G' and H are graphs, k is a positive integer and $V' \subseteq V(H)$. Then, if (G, k) is a yes-instance of H-free Edge Deletion, then (G', k) is a yes-instance of H'-free Edge Deletion, where H' is $H[V']$.*

Lemma 4.8. *Let H be a sparse (ℓ, h)-graph, where $h > \ell \geq 2$ such that both V_ℓ and V_h are independent sets. Then H-free Edge Deletion is* NP-complete. *Furthermore, the problem cannot be solved in time $2^{o(k)} \cdot |G|^{O(k)}$, unless ETH fails.*

Proof. We reduce from P_3-free Edge Deletion. Let $V' = \{u, v, w\} \subseteq V(H)$ be such that $v \in V_h$, $u, w \in V_\ell$ and V' induces a P_3 in H. Since $h \geq 3$, such a subset of vertices does exist in H. Let (G', k) be an instance of P_3-free Edge Deletion. Apply Construction 3 on (G', k, H, V') to obtain G. Let H' be $H[V']$. We claim that (G', k) is a yes-instance of P_3-free Edge Deletion if and only if (G, k) is a yes-instance of H-free Edge Deletion.

Let (G', k) be a yes-instance of P_3-free Edge Deletion. Let F' be a solution of size at most k of (G', k). For a contradiction, assume that $G - F'$ has an induced H on a vertex set U. Let V_ℓ^U and V_h^U be the V_ℓ and V_h respectively of the H induced by U in $G - F'$.

Claim 1: V_h^U is a subset of a single branch, say V_1.

Since V_h^U is an independent set in $(G - F')[U]$, V_h^U cannot span over multiple branches. Hence $V_h^U \subseteq V_1 \cup V_{G'}$. Let $x \in V_h^U \cap V_{G'}$. Consider the neighborhood of x, $N(x)$ in $(G - F')[U]$. Since the neighborhood of every vertex in H is triangle-free, $N(x)$ cannot contain vertices from multiple branches. Further, since $G' - F'$ is P_3-free, $N(x)$ can have at most one vertex from $V_{G'}$. Let x is adjacent to vertices in V_1. We note that, by construction, x has at most $h-2$ neighbors in V_1. Therefore $|N(x)| < h$, which is a contradiction. Thus we obtained that $V_h^U \subseteq V_1$.

Claim 2: $|V_\ell^U \cap V_{G'}| \leq 1$

Assume that $x \in U \cap V_{G'}$. Since degree of x in $(G - F')[U]$ is ℓ, x must have ℓ edges to V_h^U. Therefore, x must be the middle vertex of the P_3 formed by B_1 in G'.

Claim 1 and 2 imply that $|U \cap (V_1 \cup V_{G'})| \leq |U| - 2$. Hence, there exists a branch, other than V_1, say V_2, such that $V_\ell^U \cap V_2 \neq \emptyset$. Since V_ℓ is an independent set, no other branches can have vertices in V_ℓ^U. Therefore, $V_\ell^U \subseteq V_2 \cup \{x\}$. Let y be a vertex in $V_\ell^U \cap V_2$. Since y is adjacent to all vertices in V_h^U, $\ell = |V_h^U|$. Hence H is a complete bipartite graph. Further, $|V_\ell^U \cap V_2| \geq |V_\ell| - 1$. It is straight-forward to verify that V_2 does not have an independent set of size $|V_\ell| - 1$, which is a contradiction. Lemma 4.7 proves the converse. □

Now we handle the cases in which V_ℓ induces a graph with one edge.

Lemma 4.9. *Let H be a sparse (ℓ, h)-graph with at least two edges such that V_l induces a graph with one edge. Let v_{ℓ_1} and v_{ℓ_2} be the two adjacent vertices in V_ℓ. Let H' be the graph induced by $V(H) \setminus \{v_{\ell_1}, v_{\ell_2}\}$. Then, there is a linear parameterized reduction from H'-free Edge Deletion to H-free Edge Deletion.*

Proof. Let (G', k) be an instance of H'-free Edge Deletion. Apply Construction 1 on (G', k, H, V'), where V' is $V(H) \setminus \{v_{\ell_1}, v_{\ell_2}\}$. Let G be the graph obtained from the construction. We claim that (G', k) is a yes-instance of H'-free Edge Deletion if and only if (G, k) is a yes-instance of H-free Edge Deletion.

Let (G', k) be a yes-instance of H'-free Edge Deletion and let F' be a solution of size at most k of (G', k). For a contradiction, assume that $G - F'$ has an induced H with a vertex set U. It is straight-forward to verify that If a branch vertex $v_1 \in V_1$ is in U, then its neighbor in the same branch $u_1 \in V_1$ must be in U and both acts as v_{ℓ_1} and v_{ℓ_2} in the H induced by U in $G - F'$. Hence $' - F'$ has an induced H', which is a contradiction. Lemma 2.5 proves the converse. □

Observation 4.10 *Let H be a sparse (ℓ, h)-graph with at least two edges where $h > \ell \geq 2$ such that V_l induces a graph with one edge. Let v_{ℓ_1} and v_{ℓ_2} be the two adjacent vertices in V_ℓ. Let H' be the graph induced by $V(H) \setminus \{v_{\ell_1}, v_{\ell_2}\}$. Then H' has at least two edges.*

Proof. By Observation 4.6(ii), since H is not a diamond, $|V_\ell| \geq 3$. This implies that $V \setminus \{v_{\ell_1}, v_{\ell_2}\}$ is nonempty. Now the observation follows from the fact that $\ell \geq 2$. □

Now we handle Case 2, i.e., V_h induces a graph with one edge and V_ℓ is an independent set.

Lemma 4.11. *Let H be a sparse (ℓ, h) graph where $h > \ell \geq 2$, V_h induces a graph with one edge and V_ℓ is an independent set. Let H be not a t-diamond. Let v_{h_1} and v_{h_2} be the two adjacent vertices in $H[V_h]$. Let V' be $V_\ell \cup \{v_{h_1}, v_{h_2}\}$. Let H' be $H[V']$. Then, there is a linear parameterized reduction from H'-free Edge Deletion to H-free Edge Deletion.*

Proof. For convenience, we give a reduction from $\overline{H'}$-free Edge Completion to $\overline{H}$-free Edge Completion. Then the statements follow from Proposition 2.3(i).

Let (G', k) be an instance of $\overline{H'}$-free Edge Completion. Apply Construction 1 on (G', k, H, V'), where V' is $V_\ell \cup \{v_{h_1}, v_{h_2}\}$. Let G be the graph obtained from the construction. We claim that (G', k) is a yes-instance of $\overline{H'}$-free Edge Completion if and only if (G, k) is a yes-instance of $\overline{H}$-free Edge Completion.

Let (G', k) be a yes-instance of $\overline{H'}$-free Edge Completion and let F' be a solution of size at most k of (G', k). For a contradiction, assume that $G + F'$ has an induced H with a vertex set U. It is straight-forward to verify that If a branch vertex $v_1 \in V_1$ is in U, then all its neighbors in the same branch are in U and V_1 acts as $V_h \setminus \{v_{h_1}, v_{h_2}\}$ of H in $\overline{H}$ induced by U in $G + F'$. Hence $G' + F'$ has an induced $\overline{H'}$, which is a contradiction. Lemma 2.5 proves the converse. □

Observation 4.12 *Let H be a sparse (ℓ, h) graph where $h > \ell \geq 2$, V_h induces a graph with one edge and V_ℓ is an independent set. Let H be not a t-diamond, for $t \geq 2$. Let v_{h_1} and v_{h_2} be the two adjacent vertices in $H[V_h]$. Let V' be $V_\ell \cup \{v_{h_1}, v_{h_2}\}$. Let H' be $H[V']$. Then H' has at least two edges and $|V(H')| < |V(H)|$.*

Proof. Follows from the facts that $h \geq 3$ and H is not a t-diamond. □

Lemma 4.13. *Let H be a sparse (ℓ, h)-degree graph with at least two edges. Then H-free Edge Deletion is* NP-complete*. Furthermore, the problem cannot be solved in time $2^{o(k)} \cdot |G|^{O(1)}$, unless ETH fails.*

Proof. If V_ℓ induces a graph with an edge, then we apply the technique used in Lemma 4.9 and obtain a graph H' with at least two edges. Similarly, if H is not a t-diamond and V_h induces a graph with an edge, then we apply the technique used in Lemma 4.11 to obtain a graph H' with at least two edges. If the obtained graph H' is not a sparse (ℓ, h)-degree graph, then we apply Deletion-Churn(H') to obtain H''. We repeat this process until no more repetition is possible. Then, it is straight-forward to verify that we obtain a graph which is either a t-diamond, or a graph handled in Lemma 4.8 or a regular graph or a forest with at least two edges. □

4.3 Dichotomy Results

We are ready to state the dichotomy results and the parameterized lower bounds for H-free Edge Deletion and H-free Edge Completion.

Theorem 4.14. *H-free Edge Deletion is* NP-complete *if and only if H is a graph with at least two edges. Furthermore, the problem cannot be solved in time $2^{o(k)} \cdot |G|^{O(k)}$. H-free Edge Completion is* NP-complete *if and only if H is a graph with at least two non-edges. Furthermore, the problem cannot be solved in time $2^{o(k)} \cdot |G|^{O(k)}$.*

Proof. Consider H-free Edge Deletion. The statements follow from Proposition 2.4(i), Lemma 4.4, Proposition 4.1(iii) and Lemma 4.13. Now the results for H-free Edge Completion follows from Proposition 2.3(i). □

5 Concluding Remarks

Our results have wide implications on the incompressibility of H-free edge modification problems. Polynomial parameter transformation (PPT) is a widely used technique to prove the incompressibility of problems. To prove the incompressibility of a problem it is enough to to give a PPT from a problem which is already known to be incompressible, under some complexity theoretic assumption. All our reductions are linear parameterized reductions and hence are polynomial parameter transformations. The following lemma is a direct consequence of Lemma 2.6.

Lemma 5.1. *Let H be a graph and d be any integer. Let H' be obtained from H by deleting vertices with degree d or less. Then, if H'-*FREE EDGE EDITING *(*DELETION/COMPLETION*) is incompressible, then H-*FREE EDGE EDITING *(*DELETION/COMPLETION*) is incompressible.*

We give a simple example to show an implication of this lemma. Consider an n-sunlet graph which is a graph in which a vertex with degree one is attached to each vertex of a cycle of n vertices. From the incompressibility of C_n-FREE EDGE EDITING, DELETION and COMPLETION, for any $n \geq 4$, it follows that n-SUNLET-FREE EDGE EDITING, DELETION and COMPLETION are incompressible for any $n \geq 4$.

We believe that our result is a step towards a dichotomy result on the incompressibility of H-free edge modification problems. Another direction is to get a dichotomy result on the complexities of $\mathcal{H}$-free edge modification problems where $\mathcal{H}$ is a finite set of graphs.

References

[1] N. R. Aravind, R. B. Sandeep, and Naveen Sivadasan. Parameterized lower bound and NP-completeness of some H-free edge deletion problems. In *COCOA*, 2015.

[2] Nikhil Bansal, Avrim Blum, and Shuchi Chawla. Correlation clustering. *Machine Learning*, 56(1-3):89–113, 2004.

[3] Jean-Pierre Barthélemy and François Brucker. Np-hard approximation problems in overlapping clustering. *J. Classification*, 18(2):159–183, 2001.

[4] Leizhen Cai. Fixed-parameter tractability of graph modification problems for hereditary properties. *Inf. Process. Lett.*, 58(4):171–176, 1996.

[5] Leizhen Cai and Yufei Cai. Incompressibility of h-free edge modification problems. *Algorithmica*, 71(3):731–757, 2015.

[6] Yufei Cai. Polynomial kernelisation of H-free edge modification problems. Mphil thesis, Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong SAR, China, 2012.

[7] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015.

[8] P. G. Drange and M. Pilipczuk. A polynomial kernel for trivially perfect editing. In *ESA*, 2015.

[9] Ehab S El-Mallah and Charles J Colbourn. The complexity of some edge deletion problems. *Circuits and Systems, IEEE Transactions on*, 35(3):354–362, 1988.

[10] Michael R. Fellows, Jiong Guo, Christian Komusiewicz, Rolf Niedermeier, and Johannes Uhlmann. Graph-based data clustering with overlaps. *Discrete Optimization*, 8(1):2–17, 2011.

[11] Paul W Goldberg, Martin C Golumbic, Haim Kaplan, and Ron Shamir. Four strikes against physical mapping of DNA. *Journal of Computational Biology*, 2(1):139–152, 1995.

[12] Christian Komusiewicz and Johannes Uhlmann. Cluster editing with locally bounded modifications. *Discrete Applied Mathematics*, 160(15):2259–2270, 2012.

[13] Stefan Kratsch and Magnus Wahlström. Two edge modification problems without polynomial kernels. In *Parameterized and Exact Computation*, pages 264–275. Springer, 2009.

[14] Donald J Rose. A graph-theoretic study of the numerical solution of sparse positive definite systems of linear equations. *Graph theory and computing*, 183:217, 1972.

[15] R. B. Sandeep and Naveen Sivadasan. Parameterized lower bound and improved kernel for Diamond-free Edge Deletion. In *Parameterized and Exact Computation - 10th International Symposium, IPEC 2015.*

[16] Mihalis Yannakakis. Edge-deletion problems. *SIAM J. Comput.*, 10(2):297–309, 1981.