Polynomial Kernels for Paw-Free Edge Modification Problems

Yixin Cao[1(✉)] , Yuping Ke[1], and Hanchun Yuan[2]

[1] Department of Computing, Hong Kong Polytechnic University,
Hong Kong, China
`yixin.cao@polyu.edu.hk`
[2] School of Computer Science and Engineering, Central South University,
Changsha, China

Abstract. Let H be a fixed graph. Given a graph G and an integer k, the H-free edge modification problem asks whether it is possible to modify at most k edges in G to make it H-free. Sandeep and Sivadasan (IPEC 2015) asks whether the paw-free completion problem and the paw-free edge deletion problem admit polynomial kernels. We answer both questions affirmatively by presenting, respectively, $O(k)$-vertex and $O(k^4)$-vertex kernels for them. This is part of an ongoing program that aims at understanding compressibility of H-free edge modification problems.

Keywords: Kernelization · Paw-free graph · Graph modification

1 Introduction

A graph modification problem asks whether one can apply at most k modifications to a graph to make it satisfy certain properties. By modifications we usually mean additions and/or deletions, and they can be applied to vertices or edges. Although other modifications are also considered, most results in literature are on vertex deletion and the following three edge modifications: edge deletion, edge addition, and edge editing (addition/deletion).

Compared to the general dichotomy results on vertex deletion problems [1,5], the picture for edge modification problems is far murkier. Embarrassingly, this remains true for the simplest case, namely, H-free graphs for fixed graphs H. This paper is a sequel to [2], and we are aiming at understanding for which H, the H-free edge modification problems admitting polynomial kernels. Our current focus is on the four-vertex graphs; see Fig. 1 (some four-vertex graphs are omitted because they are complement of ones presented here) and Table 1.[1] We refer the reader to [2] for background of this research and related work.

---

[1] Disclaimer: Independent of our work, Eiben et al. [3] obtain similar results for edge modification problems to paw-free graphs. They are also able to develop a polynomial kernel for the editing problem.

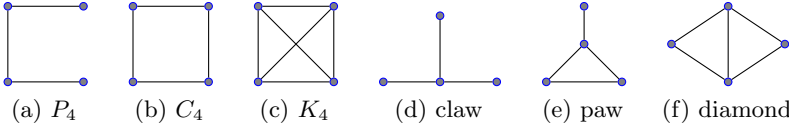(a) P_4 (b) C_4 (c) K_4 (d) claw   (e) paw   (f) diamond

Fig. 1. Graphs on four vertices (their complements are omitted).

Table 1. The compressibility results of H-free edge modification problems for H being four-vertex graphs. Note that every result holds for the complement of H; e.g., the answers are also no when H is $2K_2$ (the complement of C_4).

H	Completion	Deletion	Editing
K_4	Trivial	$O(k^3)$	$O(k^3)$ [8]
P_4	$O(k^3)$	$O(k^3)$	$O(k^3)$ [4]
Diamond	Trivial	$O(k^3)$	$O(k^8)$ [2]
Paw	$O(k)$ [this paper]	$O(k^4)$ [this paper]	$O(k^6)$ [3]
Claw	Unknown	Unknown	Unknown
C_4	No	No	No [4]

In this paper, we show polynomial kernels for both the completion and edge deletion problems when H is the paw (Fig. 1(e)). They answer open problems posed by Sandeep and Sivadasan [7].

Theorem 1. *The paw-free completion problem has a $38k$-vertex kernel.*

Theorem 2. *The paw-free edge deletion problem has an $O(k^4)$-vertex kernel.*

It is easy to see that each component of a paw-free graph is either triangle-free or a complete multipartite graph with at least three parts [6]. This simple observation motivates us to take the modulator approach. Here by modulator we mean a set of vertices that intersect every paw of the input graph by at least two vertices. Note that the deletion of all the vertices in the modulator leaves the graph paw-free. We then study the interaction between the modulator M and the components of $G - M$, which are triangle-free or complete multipartite. We use slightly different modulators for the two problems under study.

2   Paw-Free Graphs

All graphs discussed in this paper are undirected and simple. A graph G is given by its vertex set $V(G)$ and edge set $E(G)$. For a set $U \subseteq V(G)$ of vertices, we denote by $G[U]$ the subgraph induced by U, whose vertex set is U and whose edge set comprises all edges of G with both ends in U. We use $G - X$, where $X \subseteq V(G)$, as a shorthand for $G[V(G) \setminus X]$, which is further shortened as $G - v$ when $X = \{v\}$. For a set E_+ of edges, we denote by $G + E_+$ the graph obtained

by adding edges in E_+ to G,—its vertex set is still $V(G)$ and its edge set becomes $E(G) \cup E_+$. The graph $G - E_-$ is defined analogously. A paw is shown in Fig. 1(e).

For the paw-free completion (resp., edge deletion) problem, a *solution* of an instance (G, k) consists of a set E_+ (resp., E_-) of at most k edges such that $G + E_+$ (resp., $G - E_-$) is paw-free. For a positive integer k, a *k-partite* graph is a graph whose vertices can be partitioned into k different independent sets, called parts, and a k-partite graph is *complete* if all the possible edges are present, i.e., there is an edge between every pair of vertices from different parts. A *complete multipartite graph* is a graph that is complete k-partite for some $k \geq 3$. Note that here we exclude complete bipartite graphs for convenience.

Proposition 1 ([6]). *A graph G is paw-free if and only if every component of G is triangle-free or complete multipartite.*

In other words, if a connected paw-free graph contains a triangle, then it is necessarily a complete multipartite graph. Another simple fact is on the adjacency between a vertex and a (maximal) clique in a paw-free graph.

Proposition 2 ($\star$[2]). *Let K be a clique in a paw-free graph. If a vertex v is adjacent to K, then $|K \setminus N[v]| \leq 1$.*

A set $M \subseteq V(G)$ of vertices is a *modulator* of a graph G if every paw in G intersects M by at least two vertices. Note that $G - M$ is paw-free. The following three propositions characterize the interaction between the modulator M and the components of $G - M$.

Proposition 3 ($\star$). *Let M be a modulator of G. If $v \in M$ forms a triangle with some component C of $G - M$, then all the neighbors of v are in M and C.*

In other words, if a vertex v in M forms a triangle with a component of $G - M$, then v is a "private" neighbor of this component. As we will see, these components (forming triangles with a single vertex from M) are the focus of our algorithms.

Proposition 4 ($\star$). *Let G be a graph and M a modulator of G. If a vertex $v \in M$ forms a triangle with an edge in a triangle-free component C of $G - M$, then (i) v is adjacent to all the vertices of C; and (ii) C is complete bipartite.*

We say that a triangle-free component of $G - M$ is of type I if it forms a triangle with some vertex in M, or type II otherwise. By Proposition 4, for each type-I triangle-free component, all its vertices have a common neighbor in M. A component is trivial if it consists of a single vertex. Note that all trivial components of $G - M$ are type-II triangle-free components.

Proposition 5. *Let G be a graph and M a modulator of G. For any complete multipartite component C of $G - M$ and vertex $v \in M$ adjacent to C, the set of vertices in C that are nonadjacent to v is either empty or precisely one part of C.*

---

[2] The proof of a proposition marked with a $\star$ is deferred to the full version.

Proof. Suppose that the parts of C are $U_1, \ldots, U_p$. We have nothing to prove if all the vertices in C are adjacent to v. In the following we assume that, without loss of generality, v is adjacent to $u \in U_1$ and nonadjacent to $w \in U_p$. We need to argue that v is adjacent to all vertices in the first $p-1$ parts but none in the last part. Any vertex $x \in U_i$ with $1 < i < p$ makes a clique with u and w. It is adjacent to v by the definition of the modulator ($\{u, v, w, x\}$ cannot induce a paw) and Proposition 2. Now that v is adjacent to some vertex from another part ($p \geq 3$), the same argument implies $U_1 \subseteq N(v)$. To see $U_p \cap N(v) = \emptyset$, note that a vertex $w' \in U_p \cap N(v)$ would form a paw together with u, v, w, contradicting the definition of the modulator.   □

A *false twin class* of a graph G is a vertex set in which every vertex has the same open neighborhood. It is necessarily independent. The following is immediate from Proposition 5.

Corollary 1. *Let M be a modulator of G, and C a complete multipartite component of $G - M$. Each part of C is a false twin class of G.*

The preservation of false twins by all minimum paw-free completions may be of independent interest.

Lemma 1 (⋆). *Let G be a graph and E_+ a minimum set of edges such that $G + E_+$ is paw-free. A false twin class of G remains a false twin class of $G + E_+$.*

3   Paw-Free Completion

The safeness of our first rule is straightforward.

Rule 1. *If a component of G contains no paw, delete it.*

Behind our kernelization algorithm for the paw-free completion problem is the following simple and crucial observation. After Rule 1 is applied, each remaining component of G contains a paw, hence a triangle, and by Proposition 1, we need to make it complete multipartite. We say that a vertex v and an edge *xy dominate* each other if at least one of x and y is adjacent to v. Note that an edge dominates, and is dominated by, both endpoints of this edge. Every edge in a complete multipartite graph dominates all its vertices, and hence in a yes-instance, every edge "almost" dominates vertices in the component.

Lemma 2 (⋆). *Let G be a connected graph containing a paw and uv an edge in G. We need to add at least $|V(G) \setminus N[\{u, v\}]|$ edges incident to u or v to G to make it paw-free.*

For the paw-free completion problem, we build the modulator using the procedure in Fig. 2, whose correctness is proved in Lemma 3. Starting from an empty set of paws, we greedily add paws: If a paw does not intersect any previously chosen paw with two or more vertices, then add it. All the vertices of the selected paws already satisfy the definition of the modulator. After that, we have two

more steps, taking all the degree-one vertices of all paws in G, and deleting a vertex from $M \cap G'$ for certain component G' of G. Their purposes are to simplify the disposal of triangle-free components of $G - M$: In particular, (iii) and (iv) of Lemma 3 are instrumental for dealing with, respectively, type-I and type-II triangle-free components of $G - M$.

---

    0. $\mathcal{F} \leftarrow \emptyset$; $M \leftarrow \emptyset$;
    1.   **for each** paw F of G **do**
    1.1.   **if** $|F \cap F'| \leq 1$ for each paw F' in $\mathcal{F}$ **then**
             $\mathcal{F} \leftarrow \mathcal{F} \cup \{F\}$;
             add the vertices of F to M;
    1.2.   **else** add the degree-one vertex of F to M;
    2.   **for each** component G' of G **do**
    2.1.   **if** an isolated vertex v of $G' - M$ dominates all the edges in G' **then**
             find an edge uw in $G[N(v)]$;
             remove u from M;
    3.   **return** M.

Fig. 2. The construction of the modulator for G.

Lemma 3. *Let (G, k) be an instance of the paw-free completion problem. The vertex set M constructed in Fig. 2 has the following properties.*

- *(i)* *The construction is correct and its result is a modulator of G.*
- *(ii)* *For each component G' of G, we need to add at least $|M \cap G'|/4$ edges to G' to make it paw-free.*
- *(iii)* *Let C be a triangle-free component of $G - M$. If C is nontrivial and any vertex in C is contained in a triangle, then C is of type I.*
- *(iv)* *For each isolated vertex v in $G - M$, there is an edge in $G_v - N[v]$, where G_v is the component of G containing v.*

Proof. We may assume without loss of generality that G is connected and contains a paw; otherwise it suffices to work on its components that contain paws one by one, because both the construction and all the statements are component-wise.

    We denote by M' the set of vertices added to M in step 1. Note that it is a modulator of G because vertices added in step 1.1 already satisfy the definition. Let X be the set of isolated vertices in $G - M'$ each of which dominates all the edges in G. If X is empty, then step 2 is not run, $M = M'$ and we are done. In the rest, $X \neq \emptyset$. We argue first that X is a false twin class. Vertices in X are pairwise nonadjacent by definition. Suppose for contradiction that $N(x_1) \neq N(x_2)$ for $x_1, x_2 \in X$, then there is a vertex v in $N(x_1) \setminus N(x_2)$ or in $N(x_2) \setminus N(x_1)$. But then x_2 does not dominate edge vx_1, or x_1 does not dominate edge vx_2,

contradicting the definition of X. We then argue that any vertex $x \in X$ is in a triangle. By assumption, G contains a triangle uvw. If $x \in \{u, v, w\}$, then we are done. Otherwise, x must be adjacent to at least two of $\{u, v, w\}$ to dominate all the three edges in this triangle. Note that $N(x) \in M'$ because x is isolated in $G - M'$. This justifies step 2.1 of the construction of M. Note that it removes only one vertex from M'.

Now we prove by contradiction that M is a modulator of G. Suppose that there is a paw F with $|F \cap M| \leq 1$. By construction, $|F \cap M'| \geq 2$, which means $|F \cap M| = 1$ and the only vertex in $M' \setminus M$ is in F. Let $\{v\} = M' \setminus M$ and $\{u\} = F \cap M$; note that the other two vertices of F are in $V(G) \setminus M'$. Since any vertex in X is isolated in $G - M'$ and dominates all the edges of G, every component of $G - M'$ is trivial, which means that the two vertices in $F \setminus \{u, v\}$ are not adjacent. Therefore, one of u and v must be the degree-three vertex of F, and the other is a degree-two vertex of F. But the degree-one vertex of F has been added to M' in step 1.1 or 1.2, a contradiction. This justifies (i).

Let U_1 and U_2 be the sets of vertices added to M' in steps 1.1 and 1.2 respectively; $U_1 \cup U_2 = M'$. For each paw F added in step 1.1, at least one of its missing edges needs to be added to G to make it paw-free. This edge is not in any previous selected paw F', because we add F only when $|F \cap F'| \leq 1$. Therefore, we need to add at least $|U_1|/4$ edges to $G[U_1]$ to make it paw-free. On the other hand, each vertex v in U_2 is the degree-one vertex of some paw F, (it is possible that all other three vertices of F are in U_1,) we need to add at least one edge incident to v. Therefore, we need to add at least $|U_2|/2$ edges incident to vertices in U_2 to G to make it paw-free. Note that these two sets of edges we need to add are disjoint. The total number of edges we need to add to G to make it paw-free is at least $|U_1|/4 + |U_2|/2 \geq |U_1 \cup U_2|/4 = |M'|/4 \geq |M|/4$. This concludes assertion (ii).

Assertion (iii) follows from Proposition 4 if the triangle has two vertices from C: Note that the other vertex must be from M because C itself is triangle-free. Let the vertices in the triangle be $u, v \in M$ and $w \in C$. If C contains the vertex in $M' \setminus M$, then $X \subseteq C$ because it is a false twin class, and there is a vertex in M making a triangle with C, and it follows from Proposition 4. Now that C is a nontrivial component of $G - M'$, we can find a neighbor x of w in C. Note that it is adjacent to at least one of u and v; otherwise, x is the degree-one vertex of the paw induced by $\{x, u, v, w\}$ and should be in M'. As a result, x is adjacent to at least one of u and v, and then we can use Proposition 4.

Assertion (iv) follows from the construction of M and the fact that X is a false twin class we proved above.   □

Corollary 2. *If (G, k) is a yes-instance, then M contains at most $4k$ vertices.*

We proceed only when $|M| \leq 4k$. A consequence of this modulator is a simple upper bound on the number of vertices in all the type-II triangle-free components of $G - M$. Note that all trivial components of $G - M$ are considered here.

Lemma 4 (⋆). *Let (G, k) be a yes-instance to the paw-free completion problem on which Rule 1 is not applicable, and M the modulator of G. The total number of vertices in all the type-II triangle-free components of $G - M$ is at most $2k$.*

Hereafter we consider the components G' of G one by one; let $M' = M \cap V(G')$. If all components of $G' - M'$ are type-II triangle-free components, then a bound of the size of $V(G') \setminus M'$ is given in Lemma 4. In the rest, at least one component of $G' - M'$ is a type-I triangle-free component or a complete multipartite component. The way we bound $|V(G') \setminus M'|$ for such a component is to show, after applying some reductions, the minimum number of edges we need to add to G' to make it paw-free is linear on $|V(G') \setminus M'|$. The first one is very straightforward.

Lemma 5 ($\star$). *If two components in $G' - M'$ are not type-II triangle-free components, then we need to add at least $|V(G') \setminus M'|/2$ edges to G' to make it paw-free.*

Henceforth, $G' - M'$ has precisely one type-I triangle-free component or one complete multipartite component, but not both. Each part of such a component is an independent set (recall that a type-I triangle-free component is complete bipartite by Proposition 4). The next two propositions are on independent sets I of G. The first is about the cost of separating vertices in I into more than one part; it also means that a sufficiently large independent set cannot be separated. The second states that if each of the vertices in I is adjacent to all the other vertices, then we can remove all but one vertex in I from the graph.

Proposition 6 ($\star$). *Let G' be a connected graph containing a paw, and I an independent set of G'. If we do not add all the missing edges between I and $N(I)$, then we need to add at least $|I| - 1$ edges among I to G' to make it paw-free.*

Proposition 7 ($\star$). *Let I be an independent set in a component G' of a graph G. If every vertex in I is adjacent to every vertex in $V(G') \setminus I$, then (G, k) is a yes-instance if and only if $(G - (I \setminus \{v\}), k)$ is a yes-instance for any $v \in I$. Moreover, if $G - I$ is connected, then (G, k) is a yes-instance if and only if $(G - I, k)$ is a yes-instance.*

We are now ready to consider type-I triangle-free components.

Lemma 6. *Let C be a type-I triangle-free component of $G' - M'$ and let $L \uplus R$ be the bipartition of C with $|L| \geq |R|$. If any of the following conditions is satisfied, then we need to add at least $|C|/32$ edges to G' to make it paw-free.*

(i) $|L| \leq 4|M'|$;
(ii) there is an edge in $G' - N[L]$;
(iii) $V(G') \neq N[C]$ and $|L| \leq 2|R|$;
(iv) there are $|L|/2$ or more missing edges between L and $N(L)$;
(v) $|L| \leq |R| + |M'|$ and $G - N[R]$ has an edge; or
(vi) $|L| \leq |R| + |M'|$ and there are $|R|/2$ or more missing edges between R and $N(R)$.

Proof. (i) If $|L| \leq 4|M'|$, then $|C| = |L| + |R| \leq 2|L| \leq 8|M'|$, and it follows from Lemma 3(ii). (ii) By Lemma 2, we need to add at least $|L| \geq |C|/2$ edges.

(iii) Since C is complete bipartite and $|L| \geq |R|$, we can find a matching of size $|R|$ between L and R. By Lemma 2, for each vertex $v \in V(G') \setminus N[C]$, the number of edges between v and C we need to add is at least $|R| = (2|R|+|R|)/3 \geq (|L|+|R|)/3 = |C|/3$. (iv) By Proposition 6, we need to add at least $|L|/2 \geq |C|/4$ edges.

In the rest, (v) and (vi), $|L| \leq |R|+|M'|$. We may assume none of the previous conditions is satisfied. Therefore, $|L| > 4|M'|$, which means $|L| \leq 2|R|$. Also note that the proofs for these two conditions are almost the same as conditions (ii) and (iv) respectively. (v) By Lemma 2, we need to add at least $|R| \geq |C|/3$ edges. (vi) By Proposition 6, we need to add at least $|R|/2 \geq |C|/6$ edges. □

We say that a type-I triangle-free component C of $G' - M'$ is *reducible* if none of the conditions in Lemma 6 holds true.

Rule 2 ($\star$). *Let C be a type-I triangle-free component of $G' - M'$ and let $L \uplus R$ be the bipartition of C with $|L| \geq |R|$. If C is reducible, then add all the missing edges between L and $N(L)$ and all the missing edges between $V(G') \setminus N[L]$ and $N(L)$; decrease k accordingly; and remove all but one vertex from $(V(G') \setminus N[L]) \cup L$.*

In the last we consider the complete multipartite components of $G' - M'$.

Lemma 7 ($\star$). *Let C be a complete multipartite component of $G' - M'$, and let P^* be a largest part of C. If any of the following conditions is satisfied, then we need to add at least $|C|/12$ edges to G' to make it paw-free.*

 (i) $|C| \leq 3|M'|$;
 (ii) there is an edge in $G' - N[C]$;
(iii) $|P^*| > 2|C|/3$ and $G' - N[P^*]$ has an edge;
(iv) $|P^*| \leq 2|C|/3$ and $V(G') \neq N[C]$; or
 (v) $|P^*| \leq 2|C|/3$ and $V(G') = N[C]$, and for every part P of C,
   – $G' - N[P]$ contains an edge, or
   – there are at least $|P|$ missing edges between $V(G') \setminus N[P]$ and $N(P)$.

We say that a complete multipartite component C of $G' - M'$ is *reducible* if none of the conditions in Lemma 7 holds true.

Rule 3 ($\star$). *Let C be a reducible complete multipartite component of $G' - M'$ and P^* a largest part of C.*

(1) If $|P^*| > 2|C|/3$, then add all the missing edges between $V(G') \setminus N[P^*]$ and $N(P^*)$; decrease k accordingly; and remove $(V(G') \setminus N[P^*]) \cup P^*$ from G.
(2) Otherwise, find a part P such that $V(G') \setminus N[P]$ is an independent set and there are less than $|P|$ missing edges between $V(G') \setminus N[P]$ and $N(P)$. Add all the missing edges between $V(G') \setminus N[P]$ and $N(P)$; decrease k accordingly; and remove $P \cup (V(G') \setminus N[P])$ from G.

We summarize our kernelization algorithm for the paw-free completion problem in Fig. 3 and use it to prove our main result of this section.

---

procedure `reduce`(G, k)

0. **if** $k < 0$ **then return** a trivial no-instance;
1. remove all paw-free components from G;
2. construct modulator M;
3. **if** $|M| > 4k$ **then return** a trivial no-instance;
4. **if** $> 2k$ vertices in type-II triangle-free components of $G - M$ **then**
4.1.   **return** a trivial no-instance;
5. **for each** component G' of G **do**
5.1. $M' \leftarrow V(G') \cap M$;
5.2.   **if** 2 components in $G' - M'$ are not type-II triangle-free components **then**
            goto 5;
5.3.   **if** $G' - M'$ has a type-I triangle-free component C **then**
            if C is reducible **then** apply Rule 2 and **return** `reduce`(G, k);
5.4.   **if** $G' - M'$ has a complete multipartite component C **then**
            if C is reducible **then** apply Rule 3 and **return** `reduce`(G, k);
6. **if** $|V(G)| \le 38k$ **then return** (G, k);
7. **else return** a trivial no-instance.

Fig. 3. The kernelization algorithm for the paw-free completion problem.

Proof (of Theorem 1). We use the algorithm described in Fig. 3. The correctness of steps 0 and 1 follows from the definition of the problem and Rule 1 respectively. Steps 2 and 3 are justified by Lemma 3 and Corollary 2. Step 4 is correct because of Lemma 4, and after that we only need to consider the components of $G - M$ that are not type-II triangle-free components, which are dealt with in step 5. The cost of a component of G is the minimum number of edges we need to add to it to make it paw-free.

If two components of $G' - M'$ are not type-II triangle-free components, then by Lemma 5, the cost of G' is at least $|V(G') \backslash M|/2$. Therefore, there is nothing to do for step 5.2. Henceforth, $G' - M'$ has precisely one type-I triangle-free component or one complete multipartite component, but not both. The algorithm enters step 5.3 if there is a type-I triangle-free component C in $G' - M'$. If C is reducible, we rely on the correctness of Rule 2; otherwise, the cost of G' is at least $|C|/32$ by Lemma 6. The algorithm enters step 5.4 if there is a complete multipartite component C in $G' - M'$. If C is reducible, we rely on the correctness of Rule 3; otherwise, the cost of G' is at least $|C|/12$ by Lemma 7.

When the algorithm reaches step 6, neither of Rules 2 and 3 is applicable. There are at most $4k$ vertices in M, at most $2k$ vertices in all the type-II triangle-free components of $G - M$. On the other hand, for each other vertex, there is an amortized cost of at least $1/32$. Therefore, if (G, k) is a yes-instance, then the number of vertices is at most $38k$, and this justifies steps 6 and 7.

We now analyze the running time of this algorithm. When each time the algorithm calls itself in step 5.3 or 5.4, it removes at least one vertex from the graph. Therefore, the recursive calls can be made at most n times. On the other

hand, each step clearly takes polynomial time. Therefore, the algorithm returns in polynomial time. □

4   Paw-Free Edge Deletion

For this problem, we construct the modulator in the standard way. We greedily find a maximal packing of edge-disjoint paws. We can terminate by returning "no-instance" if there are more than k of them. Let M denote the set of vertices in all the paws found; we have $|M| \leq 4k$. It is a modulator because every paw not included shares at least an edge with some chosen one, hence at least 2 vertices.

The safeness of the following rule is straightforward: If we do not delete this edge, we have to delete a distinct one from each of the paws, hence $k+1$.

Rule 4. *Let uv be an edge of G. If there exist $k+1$ paws such that for any pair of them, the only common edge is uv, then delete uv from G and decrease k by 1.*

We first deal with complete multipartite components of $G - M$.

Rule 5 (⋆). *Let C be a complete multipartite component of $G - M$. From each part of C, delete all but $k+1$ vertices.*

Rule 6 (⋆). *Let C be a complete multipartite component of $G - M$. Delete all but $k+4$ parts of C that are adjacent to all vertices in $N(C)$.*

Lemma 8 (⋆). *After Rules 5 and 6 are applied, there are at most $O(k^3)$ vertices in the complete multipartite components of $G - M$.*

In the following, we assume that Rule 4 is not applicable. We mark some vertices from each of the triangle-free components that should be preserved, and then remove all the unmarked vertices. Recall that a triangle-free component of $G - M$ is of type I or type II depending on whether it forms a triangle with some vertex in M.

The following simple observation is a consequence of Proposition 3 and the definition of type-I triangle-free components.

Corollary 3. *If a vertex in M is adjacent to the triangle-free components of $G - M$, then either it is adjacent to precisely one type-I triangle-free component, or it is adjacent to only type-II triangle-free components.*

By Proposition 4, a type-I triangle-free component C of $G - M$ is complete bipartite.

Rule 7. *Let $\mathcal{C}$ be all the type-I triangle-free components of $G - M$, and let $U = \bigcup_{C \in \mathcal{C}} V(C)$.*

(i) *For each $S \subseteq M$ with $|S| = 3$ and each $S' \subseteq S$, mark $k+1$ vertices from $\{x \in U \mid N(x) \cap S = S'\}$.*

(ii) For each $C \in \mathcal{C}$ with bipartition $L \uplus R$ do the following. For each $S \subseteq M$ with $|S| = 2$ and each $S' \subseteq S$, mark $k + 3$ vertices from $\{x \in L \mid N(x) \cap S = S'\}$ and $k + 3$ vertices from $\{x \in R \mid N(x) \cap S = S'\}$.

Delete all the unmarked vertices from U.

Lemma 9. *Rule 7 is safe.*

Proof. Let G' be the graph obtained after applying Rule 7. If (G, k) is a yes-instance, then (G', k) is a yes-instance. For the other direction, suppose that (G', k) is a yes-instance, with a solution E_-. We prove by contradiction that $G - E_-$ is paw-free as well. A paw F in $G - E_-$ contains at least one deleted vertex, because $G' - E_-$ is paw-free, and at most two deleted vertices, because otherwise F is a paw of G and should be in the modulator.

Consider first that F contains only one deleted vertex x. Let C be the triangle-free component of $G - M$ containing it. If all the other three vertices in F are from M, then in step (i) we have marked $k + 1$ vertices in C that have the same adjacency to $F \setminus \{x\}$ as x in G. Since $|E_-| \leq k$, the adjacency between $F \setminus \{x\}$ and at least one of these marked vertex is unchanged. This vertex forms a paw with $F \setminus \{x\}$ in $G' - E_-$, a contradiction. Now at most two vertices of F are from M. We may assume without loss of generality that $x \in L$, where $L \uplus R$ is the bipartition of C. In step (ii) we have marked $k + 3$ vertices in L that have the same adjacency to $F \cap M$ as x; let them be Q. By Proposition 4, every vertex in $Q \cup \{x\}$ is adjacent to all vertices in R; on the other hand, no vertex in $Q \cup \{x\}$ is adjacent to any vertex in another component of $G - M$ different from C. Therefore, all vertices in $Q \cup \{x\}$ have the same adjacency to $F \setminus L$ in G. Since $|E_-| \leq k$, the adjacency between $F \setminus \{x\}$ and at least one vertex in Q is unchanged (noting that $|Q \cap F| \leq 2$). This vertex forms a paw with $F \setminus \{x\}$ in $G' - E_-$, a contradiction.

In the rest, F contains two deleted vertices x and y. If x and y are adjacent, then they are from the different parts of some component $C = L \uplus R$. Without loss of generality, we assume that $x \in L$ and $y \in R$. Since $|F \cap M| \leq 2$, by step (ii), we can find two set $Q_1 \subseteq L$ and $Q_2 \subseteq R$ that have the same adjacency to $F \cap M$ as x and y respectively. Note that $|Q_1| \geq k + 3$ and $|Q_2| \geq k + 3$. Each vertex in Q_1 has the same adjacency to $F \setminus \{x\}$. The situation is similar for Q_2 and $F \setminus \{y\}$. For $i = 1, 2$, since $|E_-| \leq k$ and $|Q_i \cap F| \leq 2$, the adjacency between $F \setminus \{x\}$ and at least one vertex in Q_i is unchanged. These two vertices form a paw with $F \setminus \{x, y\}$ in $G' - E_-$, a contradiction (because $Q_1 \uplus Q_2$ is complete bipartite). Now that x and y are not adjacent, then they are in the same part or in different components. Then one of x and y is the degree-one vertex of F and the other is a degree-two vertex of F, and we can get that the adjacency of x and y to $F \cap M$ are different. By Proposition 4, the component(s) containing x and y is complete bipartite, then x and y are adjacent to all vertices in the part that does not contain them in corresponding component. By step (ii), we can find two set Q_1 in the part containing x and Q_2 in the part containing y that have the same adjacency to $F \cap M$ as x and y respectively. Then $Q_1 \neq Q_2$. Since $|E_-| \leq k$, $|Q_1 \cap F| \leq 2$ and $|Q_2 \cap F| \leq 2$, at least one vertex in Q_1 and

at least one vertex in Q_2 are unchanged. These two vertices form a paw with $F \backslash \{x, y\}$ in $G' - E_-$, a contradiction.   □

Lemma 10 (⋆). *After Rule 7 is applied, there are at most $O(k^4)$ vertices in all the type-I triangle-free components of $G - M$.*

Finally, we deal with type-II triangle-free components of $G - M$.

Rule 8. *Let $\mathcal{C}$ be all the type-II triangle-free components of $G - M$, and let $U = \bigcup_{C \in \mathcal{C}} V(C)$.*

(i) *For each $S \subseteq M$ with $|S| = 3$ and each $S' \subseteq S$, mark $k + 1$ vertices from $\{x \in U \mid N(x) \cap S = S'\}$.*
(ii) *Mark all the vertices in non-trivial components that form a triangle with M, and for each of them, mark $k + 1$ of its neighbors in C.*

Delete all the unmarked vertices from C.

Lemma 11. *Rule 8 is safe.*

Proof. Let G' be the graph obtained after applying Rule 8. If (G, k) is a yes-instance, then (G', k) is a yes-instance. For the other direction, suppose that (G', k) is a yes-instance, with a solution E_-. We prove by contradiction that $G - E_-$ is paw-free as well. A paw F in $G - E_-$ contains at least one deleted vertex since $G' - E_-$ is paw-free.

By the definition of type-II triangle-free components, no triangle contains an edge in $\mathcal{C}$, implying that the triangle t in F contains no edge in $\mathcal{C}$. Note that if F contains three vertices in $\mathcal{C}$, then t must contain an edge in $\mathcal{C}$, a contradiction. If F contains precisely one vertex v in $\mathcal{C}$, then by step (i), we can find a vertex v' in $G' - E_-$ such that v' has the same adjacency to $F \cap M$ as v, implying that $F \setminus \{v\} \cup \{v'\}$ in $G' - E_-$ forms a paw. If F contains two vertices x and y in $\mathcal{C}$, then either x or y is in a triangle t of F. Without loss of generality, we assume that x is in t, implying that x is marked in step (ii). If y is adjacent to x, then by step (ii), there are $k + 1$ marked vertices adjacent to x; let them be Q. The vertices in Q are not adjacent to any vertex in $F \cap M$ since no triangle in G contains an edge in $\mathcal{C}$. Then, each vertex in Q forms a paw with $F \setminus \{y\}$ in G'. Since $|E_-| \le k$, there is a vertex v' in Q forms a paw with $F \setminus \{y\}$ in $G' - E_-$, a contradiction. If x is not adjacent to y, by step (i), there are $k + 1$ marked vertices Q' having the same adjacency to $F \cap M$ as y such that each vertex in Q' is not adjacent to x since no triangle in G contains an edge in $\mathcal{C}$. Then, each vertex in Q' forms a paw with $F \setminus \{y\}$ in G'. Since $|E_-| \le k$, there is a vertex v' in Q' forms a paw with $F \setminus \{y\}$ in $G' - E_-$, a contradiction.   □

Lemma 12 (⋆). *After Rule 8 is applied, there are at most $O(k^4)$ vertices in all the type-II triangle-free components of $G - M$.*

References

1. Cai, L.: Fixed-parameter tractability of graph modification problems for hereditary properties. Inf. Process. Lett. **58**(4), 171–176 (1996)
2. Cao, Y., Rai, A., Sandeep, R.B., Ye, J.: A polynomial kernel for diamond-free editing. In: ESA 2018, pp. 10:1–10:13 (2018)
3. Eiben, E., Lochet, W., Saurabh, S.: A polynomial kernel for paw-free editing (2019). arXiv:1911.03683
4. Guillemot, S., Havet, F., Paul, C., Perez, A.: On the (non-)existence of polynomial kernels for P_l-free edge modification problems. Algorithmica **65**(4), 900–926 (2013)
5. Lewis, J.M., Yannakakis, M.: The node-deletion problem for hereditary properties is NP-complete. J. Comput. Syst. Sci. **20**(2), 219–230 (1980)
6. Olariu, S.: Paw-free graphs. Inf. Process. Lett. **28**(1), 53–54 (1988)
7. Sandeep, R.B., Sivadasan, N.: Parameterized lower bound and improved kernel for diamond-free edge deletion. In: IPEC 365–376 (2015)
8. Tsur, D.: Kernel for K_t-free edge deletion (2019). arXiv:1908.03600