# A Polynomial Kernel for Line Graph Deletion

**Eduard Eiben**
Department of Computer Science, Royal Holloway, University of London, Egham, United Kingdom
eduard.eiben@rhul.ac.uk

**William Lochet**
Department of Informatics, University of Bergen, Bergen, Norway
william.lochet@uib.no

## Abstract

The line graph of a graph $G$ is the graph $L(G)$ whose vertex set is the edge set of $G$ and there is an edge between $e, f \in V(L(G)) = E(G)$ if $e$ and $f$ share an endpoint in $G$. A graph is called line graph if it is a line graph of some graph. We study the LINE-GRAPH-EDGE DELETION problem, which asks whether we can delete at most $k$ edges from the input graph $G$ such that the resulting graph is a line graph. More precisely, we give a polynomial kernel for LINE-GRAPH-EDGE DELETION with $\mathcal{O}\left(k^5\right)$ vertices. This answers open question posed by Falk Hüffner at Workshop on Kernels (WorKer) in 2013.

**2012 ACM Subject Classification** Theory of computation → Design and analysis of algorithms; Theory of computation → Graph algorithms analysis; Theory of computation → Parameterized complexity and exact algorithms

**Keywords and phrases** Kernelization, line graphs, $H$-free editing, graph modification problem

# 1 Introduction

For a family of graph $\mathcal{G}$, the general $\mathcal{G}$-Graph Modification problem asks whether we can modify a graph $G$ into a graph in $\mathcal{G}$ by performing at most $k$ simple operations. Typical examples of simple operations well-studied in the literature include vertex deletion, edge deletion, edge addition, or combination of edge deletion and addition. We call these problems $\mathcal{G}$-Vertex Deletion, $\mathcal{G}$-Edge Deletion, $\mathcal{G}$-Edge Addition, and $\mathcal{G}$-Edge Editing, respectively. By a classical result by Lewis and Yannakakis [20], $\mathcal{G}$-Vertex Deletion is NP-complete for all non-trivial hereditary graph classes. The situation is quite different for the edge modification problems. Earlier efforts for edge deletion problems [13, 23], though having produced fruitful concrete results, shed little light on a systematic answer, and it was noted that such a generalization is difficult to obtain.

$\mathcal{G}$-Graph Modification problems have been extensively investigated for graph classes $\mathcal{G}$ that can be characterized by a finite set of forbidden induced subgraphs. We say that a graph is $\mathcal{H}$-free, if it does not contain any graph in $\mathcal{H}$ as induced subgraph. For this special case, the $\mathcal{H}$-free Vertex Deletion is well understood. If $\mathcal{H}$ contains a graph on at least two vertices, then all of these problems are NP-complete, but admit $c^k n^{\mathcal{O}(1)}$ algorithm [4], where $c$ is the size of the largest graph in $\mathcal{H}$ (the algorithms with running time $f(k)n^{\mathcal{O}(1)}$ are called fixed-parameter tractable (FPT) algorithms [7, 11]). On the other hand, the NP-hardness proof of Lewis and Yannakakis [20] excludes algorithms with running time $2^{o(k)}n^{\mathcal{O}(1)}$ under Exponential Time Hypothesis (ETH) [18]. Finally, as observed by Flum and Grohe [15] a simple application of sunflower lemma [14] gives a *kernel* with $\mathcal{O}(k^c)$ vertices, where $c$ is again the size of the largest graph in $\mathcal{H}$. A kernel is a polynomial time preprocessing algorithm which outputs an equivalent instance of the same problem such that the size of the reduced instance is bounded by some function $f(k)$ that depends only on $k$. We call the function $f(k)$ the size of the kernel. It is well-known that any problem that admits an FPT algorithm admits a kernel. Therefore, for problems with FPT algorithms one is interested in polynomial kernels, i.e., kernels where size is a polynomial function.

For the edge modification problems, the situation is more complicated. While all of these problems also admit $c^k n^{\mathcal{O}(1)}$ time algorithm, where $c$ is the maximum number of edges in a graph in $\mathcal{H}$ [4], the P vs NP dichotomy is still not known. Only recently Aravind et al. [1] gave the dichotomy for the special case when $\mathcal{H}$ contains precisely one graph $H$ [1]. From the kernelization point of view, the situation is also more difficult. The reason is that deleting or adding an edge to a graph can introduce a new copy of $H$ and this might further propagate. Hence, we cannot use the sunflower lemma to reduce the size of the instance. Cai asked the question whether $H$-free Edge Deletion admits a polynomial kernel for all graphs $H$ [3]. Kratsch and Wahlström [19] showed that this is probably not the case and gave a graph $H$ on 7 vertices such that $H$-free Edge Deletion and $H$-free Edge Editing does not admit a polynomial kernel unless coNP $\subseteq$ NP/poly. Consequently, it was shown that this is not an exception, but rather a rule [5, 16]. Indeed the result by Cai and Cai [5] shows that $H$-free Edge Deletion, $H$-free Edge Addition, and $H$-free-Edge Editing do not admit a polynomial kernel whenever $H$ or its complement is a path or a cycle with at least 4 edges or a 3-connected graph with at least 2 edges missing. This suggests that actually the $H$-free modification problems with a polynomial kernels are rather rare and only for small graphs $H$. Very recently, Eiben, Lochet, and Saurabh [12] announced a polynomial kernel for the case when $H$ is a paw, which leaves only one last graph on 4 vertices for which the kernelization of $H$-free edge modification problems remains open, namely $K_{1,3}$ known also as the claw.

The class of claw-free graphs is a very well studied class of graphs with some interesting algorithmic properties. The most notorious example is probably the algorithm of Sbihi [21] for computing the maximal independent set in polynomial time. It also has been extensively studied from a structural point of view, and Chudnosky and Seymour proposed, after a series of papers, a complete characterization of claw-free graphs [6]. Because of such a characterization, it seems reasonable to believe that a polynomial kernel for Claw-free Edge Deletion exists. However, the characterization of Chudnosky and Seymour is quite complex, which makes it hard to use. For this reason, as noted by Cygan et al. [8], trying to show the existence of a polynomial kernel in the cases of sub-classes of claw-free graphs seems like a good first step to try to understand this problem. In this paper, we prove the result for the most famous such class, line graphs.

▶ **Theorem 1.** Line-Graph Edge Deletion *admits a kernel with* $\mathcal{O}\left(k^5\right)$ *vertices.*

## Overview of the Algorithm

As the first step of the kernelization algorithm, we use the characterization of line graphs by forbidden induced subgraphs to find a set $S$ of at most $6k$ vertices such that for every vertex $v \in S$, $G - (S \setminus \{v\})$ is a line graph. This is simply done by a greedy edge-disjoint packing of forbidden induced subgraphs. Having the set $S$, we use the algorithm by Degiorgi and Simon [9] to find a partition of edges of $G - S$ into cliques such that each vertex is in precisely 2 cliques. Let $\mathcal{C} = \{C_1, \ldots, C_q\}$ be the cliques in the partition. Since $G - (S \setminus \{v\})$ is also a line graph, it is rather simple consequence of Whitney's isomorphism theorem that the neighborhood of $v$ can be covered by constantly many cliques of $\mathcal{C}$. Furthermore, we will show that if a clique $C$ in $\mathcal{C}$ has more than $k+7$ vertices then the optimal solution does not contain an edge in $C$. Hence, we can partition the cliques in $\mathcal{C}$ into two groups "large" and "small". Note that if the optimal solution contains an edge in some small clique $C$, then for this change to be necessary, it has to be propagated from $S$ by modifying small cliques on some clique-path from $S$ to $C$ using only small cliques. We will therefore define the distance of a clique to $S$, without going into too many details in here, to be basically the length of a shortest clique-path from the clique to $S$ using only small cliques. Since there are only $\mathcal{O}(|S|)$ cliques in immediate neighborhood of $S$ and the number of cliques in the neighborhood of a small clique is bounded by its size, we obtain that there are at most $\mathcal{O}\left(k^d\right)$ cliques at distance at most $d$. Our main contribution and most technical part of our proof is to show that we can remove the edges covered by cliques at distance at least 5 from $G$. This is covered by Section 4. Afterwards we end up with an instance with all cliques in $\mathcal{C}$ at distance at least 5 from $S$ being singletons. As discussed above there are only $\mathcal{O}\left(k^4\right)$ cliques at distance at most 4 and because large cliques stay intact in any optimal solution, it suffices to keep $k+7$ vertices in each large clique, which leads to the desired kernel of size $\mathcal{O}\left(k^5\right)$.

# 2 Preliminaries

We assume familiarity with the basic notations and terminologies in graph theory. We refer the reader to the standard book by Diestel [10] for more information. Given a graph $G$ and a set of edges $F \subseteq E(G)$, we denote by $G - F$ the graph whose set of vertices is $V(G)$ and set of edges is the set $E(G) \setminus F$. Given two vertices $u, v \in V(G)$, we let the *distance* between $u$ and $v$ in $G$, denoted $\text{dist}_G(u,v)$), be the number of edges on a shortest path from $u$ to $v$. Furthermore, for $S \subseteq V(G)$ and $u \in V(G)$ we let $\text{dist}_G(u,S) = \min_{v \in V(G)} \text{dist}_G(u,v)$). We omit the subscript $G$, if the graph is clear from the context.

**Parameterized Algorithms and Kernelization.** For a detailed illustration of the following facts the reader is referred to [7, 11]. A *parameterized problem* is a language $\Pi \subseteq \Sigma^* \times \mathbb{N}$, where $\Sigma$ is a finite alphabet; the second component $k$ of instances $(I,k) \in \Sigma^* \times \mathbb{N}$ is called the *parameter*. A parameterized problem Π is *fixed-parameter tractable* if it admits a *fixed-parameter algorithm*, which decides instances $(I, k)$ of Π in time $f(k) \cdot |I|^{\mathcal{O}(1)}$ for some computable function $f$.

A *kernelization* for a parameterized problem Π is a polynomial-time algorithm that given any instance $(I, k)$ returns an instance $(I', k')$ such that $(I, k) \in \Pi$ if and only if $(I', k') \in \Pi$ and such that $|I'| + k' \leq f(k)$ for some computable function $f$. The function $f$ is called the *size* of the kernelization, and we have a polynomial kernelization if $f(k)$ is polynomially bounded in $k$. It is known that a parameterized problem is fixed-parameter tractable if and only if it is decidable and has a kernelization. However, the kernels implied by this fact are usually of superpolynomial size.

A *reduction rule* is an algorithm that takes as input an instance $(I, k)$ of a parameterized problem Π and outputs an instance $(I', k')$ of the same problem. We say that the reduction rule is *safe* if $(I, k)$ is a *yes*-instance if and only if $(I', k')$ is a *yes*-instance. In order to describe our kernelization algorithm, we present a series of reduction rules.

**Line graphs.** Given a graph $G$, its *line graph* $L(G)$ is a graph such that each vertex of $L(G)$ represents an edge of $G$ and two vertices of $L(G)$ are adjacent if and only if their corresponding edges share a common endpoint (are incident) in $G$. It is well known that if the line graphs of two connected graphs $G_1$ and $G_2$ are isomorphic then either $G_1$ and $G_2$ are $K_3$ and $K_{1,3}$, respectively, or $G_1$ and $G_2$ are isomorphic as well (Whitney's isomorphism theorem [22], see also Theorem 8.3 in [17]). Formally, we then study the following parameterized problem:

LINE-GRAPH-EDGE DELETION

| | |
|---|---|
| Input: | A graph $G = (V, E)$ and $k \in \mathbb{N}$. |
| Parameter: | $k$. |
| Question: | Is there a set of edges $F \subseteq E(G)$ such that $G - F$ is a line graph and $\|F\| \leq k$. |

We call a set of edges $F \subseteq V(G)$ such that $G - F$ is a line graph a *solution* for $G$. A solution $F$ is *optimal*, if there does not exists a solution $F'$ such that $|F'| < |F|$. To obtain our kernel, we will make use of several equivalent characterizations of line graphs.

▶ **Theorem 2** (see, *e.g.*, Theorem 8.4 in [17]). *The following statements are equivalent:*
1. *$G$ is a line graph.*
2. *The edges of $G$ can be partitioned into complete subgraphs is such a way that no point lies in more than two of the subgraphs.*
3. *$G$ does not have $K_{1,3}$ as an induced subgraph, and if two odd triangles (triangles with the property that there exists another vertex adjacent to an odd number of triangle vertices) share a common edge, then the subgraph induced by their points is $K_4$.*
4. *None of nine graphs of Figure 1 is an induced subgraph of $G$.*

## 3 Structure of Line Graphs

To obtain our kernel, we heavily rely on different characterizations of line graphs given by Theorem 2. The two main characterizations used throughout the paper are given in points 2. and 4. To ease the presentation of our techniques, we will define a notion of a *clique partition witness* for $G$, whose existence is implied by the point 2. of Theorem 2. Let $G$ be a line graph, a *clique partition witness* for $G$ is a set $\mathcal{C} = \{C_1, \ldots, C_q\}$ be such that:

- $C_i \subseteq V(G)$ for all $i \in [q]$,
- $G[C_i]$ is a complete graph for all $i \in [q]$, that is every $C_i$ is a clique in $G$,
- $|C_i \cap C_j| \leq 1$ for all $i \neq j \in [q]$,
- every $v \in V(G)$ is in exactly two sets in $\mathcal{C}$, and

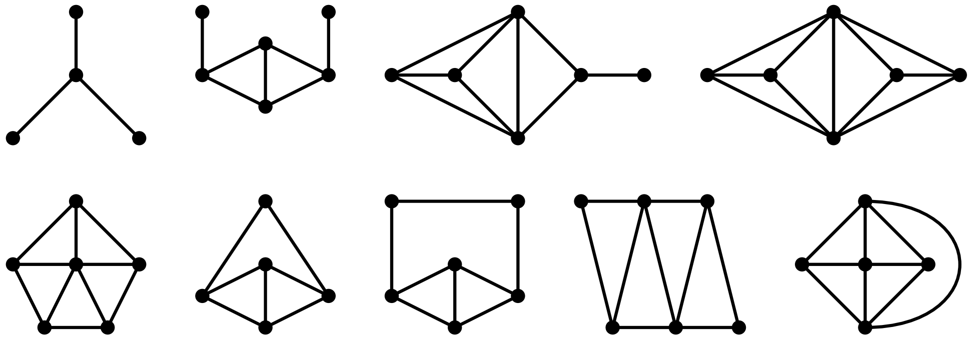


Figure 1 The nine minimal non-line graphs, from characterization of line graphs by forbidden induced subgraphs of Beineke [2]

- for every edge $uv \in E(G)$ there exists exactly one set $C_i \in \mathcal{C}$ such that $\{u, v\} \subseteq C_i$.

Note that by Theorem 2, $G$ is a line graph if and only if there exists a clique partition witness for $G$. The following three observations follow directly from the definition of clique partition witness and will be useful throughout the paper.

▶ **Observation 1.** *If $\mathcal{C}$ is clique partition witness for $G$ then every clique in $\mathcal{C}$ is either a singleton, $K_2$, or a maximal clique in $G$.*

▶ **Observation 2.** *If $\mathcal{C}$ is clique partition witness for $G$, then every maximal clique in $G$ of size at least 4 is in $\mathcal{C}$.*

▶ **Observation 3.** *If $\mathcal{C}$ is clique partition witness for $G$, then any clique of $G$ which is not a sub-clique of some element of $\mathcal{C}$ is a triangle.*

We would like to point out that given a line graph $G$ one can find a clique partition witness for $G$ for example by using an algorithm of Degiorgi and Simon [9] for recognition of line graphs in polynomial time by. We sketch the main procedure of their algorithm in the appendix together with necessary modifications to actually output a clique partition witness instead of the underlying graph $H$ such that $G = L(H)$, for completeness.

▶ **Lemma 3.** *Given a graph $G$, there is an algorithm that in time $\mathcal{O}(|E(G)| + |V(G)|)$ decides whether $G$ is a line graph and if so, constructs a clique partition witness for $G$.*

## 3.1 Level Structure of Instances

For the rest of the paper, let $G$ be the input graph and let $S$ be a set of at most $6k$ vertices such that for every $v \in S$ the graph $G - (S \setminus \{v\})$ is a line graph. We let $\mathcal{C} = \{C_1, \ldots, C_q\}$ be a clique partition witness for $G - S$. The goal of this subsection is to split the cliques in $\mathcal{C}$ to levels such that 1.) each level contains only bounded number of cliques (that are not singletons) and 2.) if we do not remove any edge at level $i$, then we do not need to remove any edge at level $j > i$. We will later show that we do not need to remove any edges in cliques in level 5. The following lemma is useful to define/bound the number of cliques at the first level, i.e., cliques that interact with $S$.

▶ **Lemma 4.** *For every vertex $v \in S$ there are at most two cliques $C_1, C_2 \in \mathcal{C}$ such that $v$ is adjacent to all vertices in $C_1 \cup C_2$ and to at most 6 vertices in $V(G) \setminus (S \cup C_1 \cup C_2)$.*

**Proof.** Let $\mathcal{C}'$ be clique partition witness for $G - (S \cup \{v\})$. By definition, there are at most two cliques $C'_1$ and $C'_2$ in $\mathcal{C}'$ that contains $v$. If $|C'_i| \geq 5$, $i \in \{1, 2\}$, then by Observation 2, $C'_i \setminus \{v\}$ is a clique in $\mathcal{C}$. Else $|C'_i \setminus v| \leq 3$ and $C'_i$ contributes to at most 3 neighbors of $v$ in $G - S$. ◀

The following lemma shows that cliques of size at least $k+7$ can serve as kind of separators that will never be changed by a solution of size at most $k$. Hence, we can remove all cliques separated from $S$ by large cliques. Moreover, it allows us to define the $(i+1)$-st level by only considering the cliques of size at most $k+6$ at level $i$.

▶ **Lemma 5.** *Let $C \in \mathcal{C}$ such that $|C| \geq k+7$ and let $A \subset E(G)$ be an optimal solution for $G$. Then $A \cap E(G[C]) = \emptyset$. Moreover, the clique partition witness $\mathcal{C}'$ for $G - A$ contains a clique $C'$ such that $C' \setminus S' = C$, where $S' \subseteq S$ is the set of vertices in $S$ that are adjacent to all vertices in $C$.*

**Proof.** Let $\{u, v\} \in A$ such that $\{u, v\} \subseteq C$. Clearly there are at most $k-1$ vertices $w$ in $C$ such that either $\{u, w\} \in A$ or $\{w, v\} \in A$. Let's $x \in C$ be such that $xv, xu$ are edges in $G - A$. Similarly, there are at most $k-1$ non-edges to $u, v, x$ in $G - A$, so let $y \in C$ be a vertex such that $yu, yv, yx$ are edges in $G - A$. Repeating the same argument once again, there is $z \in C$ such that $zu, zv, zx, zy$ are edges in $G - A$. However, the subgraph of $G - A$ induced on $u, v, x, y, z$ is $K_5$ minus an edge, which is one of the forbidden induced subgraphs in the characterization of line graphs.

The moreover part follows from the following argument. Since no two cliques in $\mathcal{C}'$ share more than 1 vertex and every vertex is in at most 2 cliques, the only way to cover all the edges of $C$, for $|C| \geq 4$ in $\mathcal{C}'$ is by a single clique $C'$. It remains to show that no vertex in $V(G) \setminus (S \cup C)$ is in $C'$. Every vertex in $V(G) \setminus S$ is in two cliques $C_1$, $C_2$ in $\mathcal{C}$ that cover all its incident edges in $G - S$. If none of these two cliques is $C$, then $C$ intersect each of these two cliques in at most 1 vertex. It follows that, because $|C| \geq 3$, there is not vertex in $V(G) \setminus (S \cup C)$ adjacent to all vertices of $C$. ◀

Let us now partition the cliques in $\mathcal{C}$ into two parts $\mathcal{C}_{<k+7}$ and $\mathcal{C}_{\geq k+7}$ such that $\mathcal{C}_{<k+7}$ contains precisely all the cliques in $\mathcal{C}$ with less than $k+7$ vertices and $\mathcal{C}_{\geq k+7}$ contains the remaining cliques. Intuitively, the deletion of an edge cannot propagate through edges covered by a clique in $\mathcal{C}_{\geq k+7}$.

We are now ready to define the level structure on the cliques in $\mathcal{C}$. We divide the cliques in $\mathcal{C}$ into levels $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_p$, for some $p \in \mathbb{N}$, that intuitively reflects on how far from $S$ the clique $C \in \mathcal{C}$ is if we consider a shortest path using only cliques in $\mathcal{C}_{<k+7}$. We will define the levels recursively as follows. By Lemma 4 for every vertex $v \in S$ there exists at most two cliques $C_1, C_2 \in \mathcal{C}$ such that $v$ is adjacent to all vertices in $C_1 \cup C_2$ and to at most 6 vertices in $V(G) \setminus (S \cup C_1 \cup C_2)$. Now let $\mathcal{N}^v$ denote the set of cliques that contains $C_1, C_2$ and all the cliques that contain at least one of the neighbors of $v$ in $V(G) \setminus (S \cup C_1 \cup C_2)$. Note that $|\mathcal{N}^v| \leq 14$. We let $\mathcal{L}_1$ be precisely the set $\bigcup_{v \in S} \mathcal{N}^v$. For $i > 1$, we then let $\mathcal{L}_i$ be the set of cliques $C$ in $\mathcal{C} \setminus (\bigcup_{j \in \{1..i-1\}} \mathcal{L}_j)$ such that there is a clique $C'$ in $\mathcal{L}_{i-1} \cap \mathcal{C}_{<k+7}$ such that $C \cap C'$ is not empty.

▶ **Observation 4.** *Let $C \in \mathcal{C}$ and $w$ a vertex in $C$. If $w$ has a neighbor in $S$, then either $C \in \mathcal{L}_1 \cup \mathcal{L}_2$ or $w$ is in a clique in $\mathcal{C}_{\geq k+7}$.*

**Proof.** Let $v \in S$ be a neighbor of $w$. Then $\mathcal{N}^v \subseteq \mathcal{L}_1$ contains a clique $C'$ with $w \in C'$. Clearly $C'$ intersects $C$ in $w$. Hence either $C' \in \mathcal{C}_{\geq k+7}$ or by the definition of $\mathcal{L}_2$ the clique $C$ is in $\mathcal{L}_1 \cup \mathcal{L}_2$. ◀

Let $p \in \mathbb{N}$ be such that $\mathcal{L}_p \neq \emptyset$ and $\mathcal{L}_{p+1} = \emptyset$. While the following Reduction Rule is not completely necessary and would be subsumed by Reduction Rule 2, we include it to showcase some of the ideas needed for the proof in a simplified setting.

▶ **Reduction Rule 1.** *Remove all vertices in $V(G)$ that are not in a clique in $\bigcup_{i \in [p]} \mathcal{L}_i$.*

**Proof of safeness.** Let $H$ be the resulting graph and let $\mathcal{C}_H$ be a set of cliques of $H$ obtained from $\mathcal{C}$, by taking all cliques in $\bigcup_{i\in[p]} \mathcal{L}_i$ and for every clique in $C \in (\mathcal{C}\setminus\bigcup_{i\in[q]} \mathcal{L}_i)$, $\mathcal{C}_H$ contains $C \cap V(H)$, if it is nonempty. Since $H$ is an induced subgraph of $G$ and line graphs can be characterized by a set for forbidden induced subgraphs, it follows that for every $A \in E(G)$, if $G - A$ is a line graphs, then $H - A$ is a line graph. It remains to show that if there is a set of edges $A \in E(H)$ such that $|A| \leq k$ and $H - A$ is a line graph, then $G - A$ is also a line graph. Let $A$ be such a set of edges of minimum size and let $\mathcal{C}_A$ be a clique partition witness for $H - A$. It suffices to show that for every clique in $C \in (\mathcal{C}_H \setminus \bigcup_{i\in[p]} \mathcal{L}_i)$, it holds that $C \in \mathcal{C}_A$. If this is the case, we get a clique partition witness for $G - A$ by replacing the cliques of $\mathcal{C}_H \setminus \bigcup_{i\in[p]} \mathcal{L}_i$ in $\mathcal{C}_A$ by $\mathcal{C} \setminus \bigcup_{i\in[p]} \mathcal{L}_i$.

Now, $C \in (\mathcal{C}_H \setminus \bigcup_{i\in[p]} \mathcal{L}_i)$ means that all cliques intersecting $C$ are in $\mathcal{C}_{\geq k+7}$. Moreover, because all vertices in $H$ are in some clique on some level, by Lemma 5, for each clique $C_1 \in \mathcal{C}_H$ that intersect $C$ there is a clique in $C_1' \in \mathcal{C}_A$ that is the union of $C_1$ and some vertices in $S$. Hence, all vertices in $C$ are already in at least one clique in $\mathcal{C}_A \setminus C$ and all the edges incident to exactly one vertex in $C$ are already covered by these cliques. And hence every clique that contains a vertex in $C$ and intersects every other clique in $\mathcal{C}_A$ in at most one vertex has to be a subset of $C$. Moreover, the cliques in $\mathcal{C}_A$ that are subsets of $C$ have to be vertex disjoint, since every vertex is in at most 2 cliques in $\mathcal{C}_A$. Hence, if $C$ is not in $\mathcal{C}_A$, then some of the edges in $C$ have to be in $A$, but replacing all the subsets of $C$ in $\mathcal{C}_A$ by $C$ gives a clique partition witness for $H - A'$ for some $A' \subsetneq A$ which contradicts the fact that $A$ is of minimum size. ◀

We will also say that $C \in \mathcal{C}$ is at $\mathcal{L}$-distance $d$ from $S$, denoted by $\text{dist}^{\mathcal{L}}(C)$, if $C$ is in $\mathcal{L}_d$. We note that $\mathcal{C}$ still contains some cliques that are not in any of $\mathcal{L}_i$'s. We will let $\text{dist}^{\mathcal{L}}(C) = \infty$ for such a clique $C$. We can now upper bound the number of cliques at $\mathcal{L}$-distance $d$ from $S$.

▶ **Lemma 6.** *There are at most $14|S|(k+6)^{d-1}$ cliques in $\mathcal{C}$ at level $d$,* i.e.*, in $\mathcal{L}_d$.*

**Proof.** By the definition of $\mathcal{L}_1$, we have that $\mathcal{L}_1$ contains at most $14|S|$ cliques. Now by the definition of $\mathcal{L}_d$ we know that for any $d \geq 2$ a clique is in $\mathcal{L}_d$ if and only if it shares a vertex with a clique in $\mathcal{C}_{<k+7}$ in $\mathcal{L}_{d-1}$. Since no three cliques in $\mathcal{C}$ can share a vertex the number of cliques in $\mathcal{L}_d$ is precisely the number of vertices in the cliques from $\mathcal{C}_{<k+7}$ in $\mathcal{L}_{d-1}$ and the lemma follows by a simple induction on $d$. ◀

The remainder of the algorithm consists of two steps. First, in Section 4, we show that we can remove all edges from cliques that are at $\mathcal{L}$-distance at least 5 from $S$. Afterwards, due to Lemma 6, we are left with only $\mathcal{O}\left(k^4\right)$ non-singleton cliques in $\mathcal{C}$. To finish the algorithm in Section 5, for each clique $C \in \mathcal{C}$ that is not a singleton, we mark arbitrary $k+7$ vertices in $C$ and remove all unmarked vertices from $G$. It is then rather straightforward consequence of Lemma 5 that this rule is safe and we get an equivalent instance with $\mathcal{O}\left(k^5\right)$ vertices.

## 4 Bounding the Distance from $S$

The purpose of this section is to show that it is only necessary to keep the cliques in $\mathcal{C}$ that are at $\mathcal{L}$-distance at most 4 from $S$ (and adding a singleton for vertices covered by exactly one clique at $\mathcal{L}$-distance at most 4). To do so, we need to show that there is always a solution that does not change the cliques at $\mathcal{L}$-distance 5 at all. To do so, we first need to understand the interaction of cliques at $\mathcal{L}$-distance 4 from $S$ with the solution. The first step will be to show that there is an optimal solution $A$ with clique partition witness $\mathcal{C}_A$ such that all cliques in $\mathcal{C}_A$ that share an edge with a clique in $\mathcal{C}$ at $\mathcal{L}$-distance at least 4 from $S$ are actually subcliques of a clique in $\mathcal{C}$ (when restricted to $G - S$). It is a simple consequence of Lemma 5 that this is true for any clique that intersect a clique in $\mathcal{C}_{\geq k+7}$ in an edge. Hence, we can only care about cliques in $\mathcal{C}_A$ that intersect a clique $C$ in $\mathcal{C}_{<k+7}$ in an edge. By Observation 4, no vertex in $C$

has a neighbor in $S$. It then follows by Observation 3 that any clique in $\mathcal{C}_A$ that intersects $C$ in an edge and is not a subclique of a clique in $\mathcal{C}$ is indeed a triangle. This leads us to the following definition.

▶ **Definition 7** (bad triangle). *Let $A \subseteq E(G)$ be such that $G - A$ is a line graph and let $\mathcal{C}_A$ be a clique partition witness of $G - A$. A triangle $xyz \in \mathcal{C}_A$ is said to be* bad *if it is not a sub-clique of a clique in $\mathcal{C}$, and one of the edges of the triangle, say $xy$, is an edge contained in a clique of $\mathcal{L}$-distance at least 4 from $S$.*

▶ **Lemma 8.** *There exists an optimal solution without any bad triangle.*

**Proof.** Let $A$ be an optimal solution and $\mathcal{C}_A$ the clique partition witness of $G - A$. Suppose $xyz$ is a bad triangle and let $C_1, C_2$ and $C_3$ be the elements of $\mathcal{C}$ containing the edges $xy, yz$ and $zx$ respectively. See also Figure 2 for an illustration. Since $xyz$ is a bad triangle, no clique in $\mathcal{C}_A$ is a superset of $C_i$, $i \in \{1, 2, 3\}$ and it is a simple consequence of Lemma 5 that $C_i \in \mathcal{C}_{<k+7}$. By definition of bad triangle, at least one of $C_1$, $C_2$, and $C_3$ is at $\mathcal{L}$-distance 4 from $S$ and hence all of these cliques are at $\mathcal{L}$-distance at least 3 from $S$. Let $X$ (resp. $Y$, $Z$) denote the other clique of $\mathcal{C}_A$ containing $x$ (reps. $y$, $z$). Let us define $X_1 = X \cap C_1, X_3 = X \cap C_3, Y_1 = Y \cap C_1, Y_2 = Y \cap C_2, Z_3 = Z \cap C_3$ and $Z_2 = Z \cap C_2$.

Let $C'_1 = X_1 \cup Y_1$, $C'_2 = Y_2 \cup Z_2$ and $C'_3 = Z_3 \cup X_3$. Note that $C'_i$ is a sub-clique of $C_i$ for $i \in [3]$. Now for every $i \in [3]$ we will update $C'_i$ as follows. As long as there exists an edge $e$ in $C'_i$ such that $e$ belongs to $K_i \in \mathcal{C}_A$, $K_i$ is a sub-clique of $C_i$ and $K_i \not\subseteq C'_i$, we set $C'_i := C'_i \cup K_i$ (see also Figure 2b). When this process stops, $C'_i$ corresponds to the union of a set of elements of $\mathcal{C}_A$ : $K^i_1, \dots, K^i_{l_i}$ which are sub-clique of $C_i$, and $C'_i$. Moreover, for any edge $e$ of $C'_i$ which is strictly contained in another clique of $\mathcal{C}_A$ (meaning this clique is not $e$), then this clique has to be a triangle by Observation 3, as the clique of $\mathcal{C}$ containing $e$ is $C_i$. Let $e^i_1, \dots, e^i_{s_i}$ denote the set of such edges and let $C^i_1, \dots, C^i_{s_i}$ be the triangles of $\mathcal{C}_A$ containing these edges. Note that $|A \cap C'_1| \geq s_1$, as for any edge $e^1_j$, either $x$ or $y$ has to be non adjacent to each extremity in $G - A$ or the edge would be in two cliques of $\mathcal{C}_A$ (the same statement is also correct for $|A \cap C'_2|$ and $|A \cap C'_3|$ ). Let $A'$ be the set obtained from $A$ by

- Removing all the edges of $A \cap C'_1$, $A \cap C'_2$ and $A \cap C'_3$.
- Adding one of the two edges of $C^i_j$ different from $e^i_j$ for every $i \in [3]$ and $j \in [s_i]$ (see Figure 2c illustrating the replacement of $C^j_i$ in $\mathcal{C}_A$ by its proper subclique in $\mathcal{C}_{A'}$ implied by this addition of an edge in $A'$.).

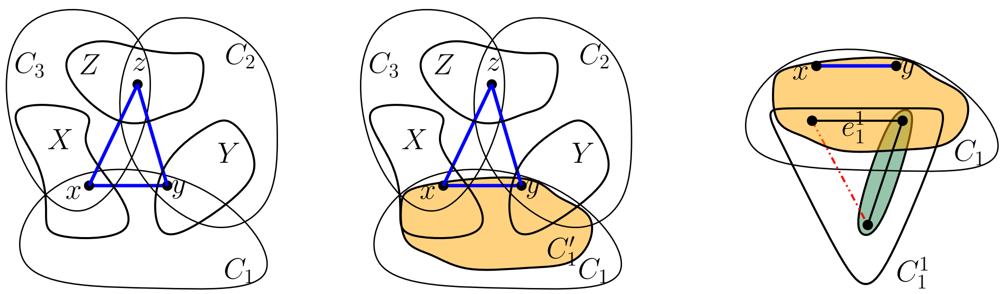
▷ Claim 9. $A'$ is a set of edges not larger than $A$ and such that $G - A'$ is a line graph with fewer bad triangles than $G - A$.

**Proof.** The fact that $|A'| \leq |A|$ follows from the fact that $|A \cap C'_i| \geq s_i$ for all $i \in [3]$. To see that $G - A'$ is a line graph, let us show that $\mathcal{C}_{A'}$ defined as follows is a clique partition witness for $G - A'$. Let $\mathcal{C}_{A'}$ be the set defined from $\mathcal{C}_A$ by

- Removing $C_A$, $X$, $Y$, $Z$, every $C^i_j$ for $i \in [3]$, $j \in [s_i]$, every $K^i_j$ for every $i \in [3]$ and $j \in [l_i]$ and every edge which are contained in one of the $C'_i$.
- Adding $C'_i$ for $i \in [3]$ and for every $i \in [3]$ and $j \in [s_i]$ the edge of $C^i_j$ which has not been removed from $A$, as well as singletons for vertices belonging to only one clique.

First it is clear that any set added to $\mathcal{C}'_A$ is a clique as $A'$ does not contain any edge in $A \cap C'_1$, $A \cap C'_2$ and $A \cap C'_3$ and these sets are cliques of $G$.

Now take $B$ and $C$ two cliques of $\mathcal{C}'_A$. If $B$ and $C$ belong to $\mathcal{C}_A$, then clearly their intersection has size at most 1. If one belongs to $\mathcal{C}_A$ and the other is the remaining edge of $C^i_j$ for $i \in [3]$ and $j \in [s_i]$, then it is also clear as it is true for $C^i_j$. For $i, j \in [3]^2$, $C'_i$ and $C'_j$ also intersect on one vertex, because $C_i$ and $C_j$ do and moreover, the cliques of $\mathcal{C}_A$ intersecting $C'_i$




**(a)** A bad triangle $xyz$ in $\mathcal{C}_A$. $C_1$, $C_2$, $C_3$ are three cliques in $\mathcal{C}$ containing $xy$, $yz$, and $xz$ respectively. $X, Y, Z \in \mathcal{C}_A$ are the cliques containing $x, y, z$ other than $xyz$.

**(b)** $C_1'$ is the inclusion minimal clique such that $(X \cup Y) \cap C_1 \subseteq C_1' \subseteq C_1$ and for all $K \in \mathcal{C}_A$ if $K \subseteq C_1$ and $|K \cap C_1'| \geq 2$, then $K \subseteq C_1'$. $C_2'$ and $C_3'$ are defined analogously.

**(c)** $C_1^1$ intersects $C_1'$ in an edge $e_1^1$. $C_1^1$ is replaced by an edge other than $e_1^1$. This forces to include one edge in $C_1^1$ to a solution $A'$. However, this can be seen as replacing an edge between $\{x, y\}$ and endpoints of $e_1^1$ that is in $A$.

**Figure 2** The treatment of bad triangles. Let $A \subseteq E(G)$ be an optimal solution, $\mathcal{C}_A$ a clique partition witness for $A$. A bad triangle $xyz$ together with cliques $X$, $Y$, $Z$, as defined in Subfigure 2a are replaced by cliques $C_1'$, $C_2'$, and $C_3'$ defined in Subfigure 2b. Subfigure 2c show the treatment of cliques in $\mathcal{C}_A$ that intersect $C_i'$ in an edge. By definition of $C_i'$, such clique is not a subclique of $C_i$ and hence a triangle.

on two vertices are exactly the $C_j^i$, so if $B = C_i'$ and $A \in \mathcal{C}_A$, the intersection has also size at most 1, and we covered all the cases for $|C \cap B|$.

Now for every vertex $x \in V(G)$, if $x$ does not belong to $C_1', C_2'$ and $C_3'$, then it belongs to the same cliques as in $\mathcal{C}_A$ (where the $C_j^i$ have been reduced to an edge and a singleton). For the vertices of $C_1', C_2'$ and $C_3'$ different from $x, y, z$, we replaced one sub-clique of $C_i$ by another. Finally $x$ belongs to $C_1'$ and $C_3'$, $y$ to $C_1'$ and $C_2'$ and $z$ to $C_2'$ and $C_3'$.

Suppose $uv$ is an edge of $E(G - A')$. If $uv$ belongs to one of the $C_i'$, then by definition of the $C_j^i$ and because we removed all these triangles, $uv$ only belongs to one clique. For the other edges of $E(G - A')$, the fact that $uv$ belongs to exactly one clique of $\mathcal{C}_A'$ follows from the fact that $A'$ differs on those edges from $A$ only because we added some edges of the $C_j^i$, and $\mathcal{C}_A$ differs on these vertices only because we changed $C_j^i$ into the remaining edge outside $C_i'$.

Overall $\mathcal{C}_{A'}$ is indeed a clique partition for $G - A'$. Moreover, to obtain it, we removed at least one bad triangle from $\mathcal{C}_A$ $(C_A)$ without adding one. This ends the proof of the claim. ◀

Finally, we can repeat the process until $\mathcal{C}_{A'}$ is without any bad triangles, which ends the proof of the lemma. ◀

Before we show that indeed all cliques at $\mathcal{L}$-distance at least 5 from $S$ are intact in some optimal solution, we show another auxiliary lemma that is rather simple consequence of Lemma 8, namely that there is a clique partition witness for some optimal solution $A$ such that no two cliques $\mathcal{C}_A$ that intersect the same clique $C \in \mathcal{C}$ at $\mathcal{L}$-distance at least 4 from $S$ in an edge can intersect. This is important later to show that indeed no vertex in a clique $C \in \mathcal{C}$ at $\mathcal{L}$-distance 5 from $S$ will be in two cliques in $\mathcal{C}_A$ that are not subsets of $C$.

▶ **Lemma 10.** *There exists an optimal solution $A \subseteq E(G)$ without any bad triangles and clique partition witness $\mathcal{C}_A$ for $G - A$ such that for every $C \in \mathcal{C}$ of $\mathcal{L}$-distance at least 4 and every $w \in C$, if $C_1^w$ and $C_2^w$ are the two cliques in $\mathcal{C}_A$ containing $w$, then either $C_1^w \cap C = \{w\}$ or $C_1^w \cap C = \{w\}$.*

**Proof.** Let $A \subseteq E(G)$ be an optimal solution for $G$ without any bad triangles and clique partition witness $\mathcal{C}_A$ for $G - A$ minimizing the number of pairs $(C, w)$ for which $C$ is at

$\mathcal{L}$-distance at least 4, $w \in C$ and the two cliques, denoted $C_1^w$ and $C_2^w$, in $\mathcal{C}_A$ containing $w$ intersect $C$ in two vertices. Furthermore, it follows from Lemma 5 that $C \in \mathcal{C}_{<k+7}$, as the clique containing $C$ as a subclique in $\mathcal{C}_A$ would intersect $C_1^w$ in two vertices. Since there are no bad triangles and $C$ is at $\mathcal{L}$-distance at least 4, it follows that $C_1^w \subseteq C$ and $C_2^w \subseteq C$ and in particular $C_1^w \cup C_2^w$ is a clique in $G$. Indeed, our goal is to replace $C_1^w$ and $C_2^w$ by a clique $D$ such that $C_1^w \cup C_2^w \subseteq D \subseteq C$. We start by setting $D = C_1^w \cup C_2^w$. We will also keep a track of cliques we will remove from $\mathcal{C}_A$. This set will be $\mathcal{D}$ and initialize it as $\mathcal{D} = \{C_1, C_2\}$.

Similarly to the proof of Lemma 8, the only reason why we cannot replace $C_1$ and $C_2$ by $D$ and obtain a solution that removes a subset of edges of $A$ is because there exist two vertices $v_1, v_2 \in D$ and a clique $C_{12} \in \mathcal{C}_A$ with $\{v_1, v_2\} \subseteq C_{12}$. Observe, that by our assumption there is no bad triangle and $C_{12} \subseteq C$. We let $D = D \cup C_{12}$ and $\mathcal{D} = \mathcal{D} \cup C_{12}$ and repeat until there is no such pair of vertices. Note that every vertex in $G$ is in at most two cliques of $\mathcal{C}_A$. Therefore, this process has to stop after at most $2|C|$ steps.

When there are no two vertices in $D$ that appear together in a different clique, we remove $\mathcal{D}$ from $\mathcal{C}_A$ and replace it by $D$ and $\{v\}$. For every vertex that appear in $D$, we removed one clique that it appeared in. Hence, every vertex appear in at most 2 cliques and we can always add a singleton to clique partition witness for vertices that are only in one clique. Moreover, no two cliques intersect in two vertices, since $D$ is the only clique we added, and we removed/changed all the cliques that intersected $D$ in at least two vertices. Finally, all edges in $G - A$ remain covered, we only potentially covered some additional edges in $D$.

Note that this procedure does not introduce any bad triangles or new pair $(C', w')$ for which $C'$ is at $\mathcal{L}$-distance at least 4, $w' \in C'$ and the two cliques in $\mathcal{C}_A$ containing $w'$ intersect $C'$ in two vertices. As it also removes one such pair, we obtain a contradiction with the choice of $A$. We can therefore deduce that $A$ does not contain such pair $(C, w)$ and the lemma follows. ◀

Finally, we can state the main lemma of this section.

▶ **Lemma 11.** *There exists an optimal solution $A$ for $G$ and a clique partition witness $\mathcal{C}_A$ for $G - A$ such that for every clique $C \in \mathcal{C}$ at $\mathcal{L}$-distance at least 5 it holds that $C \in \mathcal{C}_A$.*

**Proof.** Let $A$ be an optimal solution without any bad triangles and clique partition witness $\mathcal{C}_A$ for $G - A$ such that for every $C \in \mathcal{C}$ of $\mathcal{L}$-distance at least 4 and every $w \in C$, if $C_1^w$ and $C_2^w$ are the two cliques in $\mathcal{C}_A$ containing $w$, then either $C_1^w \cap C = \{w\}$ or $C_1^w \cap C = \{w\}$. Note that existence of such a solution is guaranteed by Lemma 10. Moreover let $(A, \mathcal{C}_A)$ be such an optimal solution satisfying properties in Lemma 10 that minimizes the number of cliques $C \in \mathcal{C}$ of $\mathcal{L}$-distance at least 5 such that $C \notin \mathcal{C}_A$. We claim that $A$ satisfies the properties of the lemma.

For a contradiction let $C \in \mathcal{C}$ be a clique at $\mathcal{L}$-distance at least 5 and let $C_1, \ldots, C_p$ be the cliques in $\mathcal{C}_A$ that intersects $C$ in at least 2 vertices. Since there is no bad triangle, it follows that $C_i \subseteq C$ for all $i \in [p]$ and by optimality of $A$, $p = 1$ (else $\bigcup_{i \in [p]} C_i$ is missing at least one edge). We claim that $C = C_1$. Else let $v \in C \setminus C_1$. Note that $C \in \mathcal{C}_{<k+7}$ and hence by Observation 4 $v$ does not have a neighbor in $S$. In particular all neighbors of $v$ are covered by two cliques in $\mathcal{C}$, one of those cliques is $C$ and let the other clique be $C^v$. Moreover, Let $C_1^v$ and $C_2^v$ be the two cliques in $\mathcal{C}_A$ containing $v$. Since $v \in C \setminus C_1$ both $C_1^v$ and $C_2^v$ are subsets of $C^v$. However, $C^v$ is either in $\mathcal{C}_{\geq k+7}$ and $\mathcal{C}_A$ contains $C^v$ and the cliques $C_1^v$ and $C_2^v$ are $C^v$ and $\{v\}$ respectively, or $C^v \in \mathcal{C}_{<k+7}$, in which case $C^v$ is at $\mathcal{L}$-distance at least 4 from $S$, because it shares a vertex with the clique $C$ at $\mathcal{L}$-distance at least 5 from $S$. It follows by the choice of $A$ that either $C^v \cap C_1^v = \{v\}$ or $C^v \cap C_2^v = \{v\}$, but then again either $C_1^v$ or $C_2^v$ is the singleton $\{v\}$. However then the clique partition witness $(\mathcal{C}_A \setminus \{C_1, \{v\}\}) \cup \{C_1 \cup \{v\}\}$ defines a better solution. It follows that indeed $C \in \mathcal{C}_A$ for all cliques in $\mathcal{C}$ at $\mathcal{L}$-distance at least 5 in $G$. ◀

We are now ready to present our main reduction rule. Note that it would seem that we could remove just the vertices that do no appear in a clique at distance at most 4. However, because of the cliques in $\mathcal{C}_{\geq k+7}$ at the first four levels, we would be potentially left with many cliques at $\mathcal{L}$-distance infinity that we cannot remove because all of their vertices are in a large clique at $\mathcal{L}$-distance at most 4 from $S$. While this case could have been dealt with separately, we can actually show a stronger claim, *i.e.,* that we can remove all edges from $G$ that are covered by a clique at $\mathcal{L}$-distance at least 5 from $S$. Note that in this case we cannot easily claim that if $(G, k)$ is YES-instance then so is the reduced instance and we crucially need the fact that cliques at $\mathcal{L}$-distance at least 5 are kept in clique partition witness of some optimal solution.

▶ **Reduction Rule 2.** *Remove all edges $uv \in E(G)$ such that $\{u, v\} \subseteq C$ for some clique $C$ with $\mathrm{dist}^{\mathcal{L}}(C) \geq 5$. Afterwards remove all isolated vertices from $G$.*

Let $\mathcal{D}$ be the set of cliques at $\mathcal{L}$-distance at least 5 from $S$, $V_5$ the set of vertices that appear in a clique in $\mathcal{D}$ and in a clique in $\mathcal{C} \setminus \mathcal{D}$ and $G'$ be the graph obtained after applying the reduction rule and let $\mathcal{C}' = \mathcal{C} \setminus \mathcal{D} \cup \bigcup_{v \in V_5} \{v\}$. Note that $\mathcal{C}'$ is a clique partition witness for $G' - S$ and that $\{v\}$, for $v \in V_5$, is a clique at $\mathcal{L}$-distance at least 5.

**Proof of safeness.** Let $\mathcal{D}$, $V_5$, $G'$, $\mathcal{C}'$ be as described above and let $A$ be an optimal solution for $G'$, that is $G' - A$ is a line graph, and let $\mathcal{C}_A$ be clique partition witness for $G' - A$. By Lemma 11, we can assume that $\bigcup_{v \in V_5} \{v\} \subseteq \mathcal{C}_A$. We will show that $(\mathcal{C}_A \setminus \bigcup_{v \in V_5} \{v\}) \cup \mathcal{D}$ is a clique partition witness for $G - A$. Clearly each edge in $G - A$ is either covered by $(\mathcal{C}_A \setminus \bigcup_{v \in V_5} \{v\})$ or by $\mathcal{D}$. It is also easy to see that every vertex is in precisely two cliques. Moreover, two cliques in $\mathcal{D}$ intersect in at most 1 vertex, because $\mathcal{D} \subseteq \mathcal{C}$ and similarly two cliques in $\mathcal{C}_A$ intersect in at most one vertex. Finally, let $D \in \mathcal{D}$ and $C \in (\mathcal{C}_A \setminus \bigcup_{v \in V_5} \{v\})$. Clearly, $D \cap C \subseteq V_5$. Moreover, for $\{u, v\} \subseteq D$, the edge $uv$ is not in $G'$ and hence $\{u, v\} \nsubseteq C$. Hence, $|D \cap C| \leq 1$.

On the other hand, let $A$ be an optimal solution for $G$ and a clique partition witness $\mathcal{C}_A$ for $G - A$ such that for every clique $C \in \mathcal{C}$ at $\mathcal{L}$-distance at least 5 it holds that $C \in \mathcal{C}_A$. Note that the existence of $(A, \mathcal{C}_A)$ is guaranteed by Lemma 11. We claim that $G' - A$ is a line graph. By the choice of $(A, \mathcal{C}_A)$, it follows that $\mathcal{D} \subseteq \mathcal{C}_A$. Moreover, for every edge $e$ that is covered by a clique in $\mathcal{D}$ it holds that $e \notin E(G')$. It follows rather straightforwardly that $\mathcal{C}_A \setminus \mathcal{D} \cup \bigcup_{v \in V_5} \{v\}$ is indeed a clique partition witness for $G' - A$. ◀

## 5 Finishing the Proof

Suppose now that $G$, $S$, and $\mathcal{C}$ correspond to the instance after applying Reduction Rules 1 and 2. Clearly all cliques in $\mathcal{C}$ are either at $\mathcal{L}$-distance at most 4 from $S$ or there are singletons at distance 5 or infinity, depending on whether the singleton intersects a clique in $\mathcal{C}_{<k+7}$ or a clique in $\mathcal{C}_{\geq k+7}$, respectively. It follows from Lemma 6 that there are at most $\mathcal{O}\left(k^4\right)$ cliques at distance at most 4. We let $M$ be any minimal w.r.t. inclusion set of vertices such that for every clique $C$ in $\mathcal{C}$ at $\mathcal{L}$-distance at most 4 it holds that $|M \cap C| \geq \min\{|C|, k + 7\}$. Such a set $M$ can be easily obtained by including arbitrary $\min\{|C|, k + 7\}$ vertices from every clique $C$ at distance at most 4 and then removing the vertices $v$ such that $|(M \setminus \{v\}) \cap C| \geq \min\{|C|, k + 7\}$ for all $C \in \mathcal{C}$ at $\mathcal{L}$-distance at most 4. From this construction it is easy to see that $|M| = \mathcal{O}\left(k^5\right)$.

▶ **Reduction Rule 3.** *Remove all vertices in $V(G) \setminus (S \cup M)$ from $G$.*

**Proof of safeness.** Let the clique partition witness $\mathcal{C}'$ for $G - (S \cup M)$ be $\{C \cap M \mid C \in \mathcal{C}, C \cap M \neq \emptyset\}$. Since line graphs are characterized by a finite set of forbidden induced subgraphs, it is easy to see that if $G - A$ is a line graph, for some $A \subseteq E(G)$, then $G[S \cup M] - A = (G - A)[S \cup M]$

is also a line graph. For the other direction, let $A \subseteq E(G)$ be such that $G[S \cup M] - A$ is line graph. We will show that $G - A$ is a line graph. Let $\mathcal{C}_A$ be a clique partition witness for $G[S \cup M] - A$. Now let $\mathcal{C}'_A$ be the set we obtain from $\mathcal{C}_A$ by adding to it all the singleton cliques in $\mathcal{C}$ that do not contain a marked vertex and for every clique $C \in \mathcal{C}_A$ for which there exists $C' \in \mathcal{C}$ with $C \setminus S \subseteq C'$, we replace $C$ by $C' \cup (C \cap S)$.

First let us verify that every vertex in $V(G)$ is in precisely two cliques in $\mathcal{C}'_A$. It is easy to see that this holds for $v \in S \cup M$, because $\mathcal{C}_A$ is a clique partition witness for $G[S \cup M] - A$ and we only added new cliques containing vertices in $V(G) \setminus (M \cup S)$ or extended existing cliques in $\mathcal{C}_A$ by vertices in $V(G) \setminus (M \cup S)$. Now let $v \in V(G) \setminus M$ and let $C_1, C_2 \in \mathcal{C}$ be two cliques that contain $v$. Because all cliques in $\mathcal{C}$ at $\mathcal{L}$-distance at least 5 are singletons and we keep all vertices of the cliques at $\mathcal{L}$-distance at most 4 of size less than $k + 7$, it follows that $C_1$ and $C_2$ either both contain at least $k + 7$ vertices or one of them, say $C_2$, is a singleton and the other, $C_1$, contains at least $k + 7$ vertices. If $C_2$ is a singleton, then $C_2 \in \mathcal{C}'_A$. Else for $C_i$, $i \in \{1, 2\}$, with $|C_i| \geq k + 7$ there is $C'_i \in \mathcal{C}'$ with $|C'_i| \geq k + 7$ and $C'_i \subseteq C_i$. By Lemma 5, $\mathcal{C}_A$ contains a clique $C_i^A$ such that $C_i^A \setminus S = C'_i \setminus C_i$. By the construction of $\mathcal{C}'_A$ it now follows that $\mathcal{C}'_A$ contains $C_i^A \cup C_i$. From Lemma 4 it follows that if $u \in S$ is adjacent to at least 7 vertices in a clique in $\mathcal{C}$, then it is adjacent to the whole clique. Hence $C_i^A \cup C_i$ indeed induces a complete subgraph of $G - A$. It follows that $v$ is indeed in precisely two cliques in $\mathcal{C}'_A$. Note that above also shows that the sets in $\mathcal{C}'_A$ induce cliques in $G - A$. Furthermore every edge in $G - A$ either has both endpoints in $S \cup M$ and are covered by a clique $C$ in $\mathcal{C}_A$ such that $\mathcal{C}'_A$ contains a superset of $C$, or they are in the same clique of size at least $k + 7$ in $\mathcal{C}$ that is a subset of a clique in $\mathcal{C}'_A$ as well.

It remains to show that $|C_1 \cap C_2| \leq 1$ for all cliques in $\mathcal{C}'_A$. If $|C_1 \cap C_2| \geq 2$, then at least one of the vertices in $C_1 \cap C_2$ has to be outside $S \cup M$. But then from the above discussion follows that $C_1 \setminus S$ and $C_2 \setminus S$ are in $\mathcal{C}$, $|C_1 \setminus S| \geq k + 7$, $|C_2 \setminus S| \geq k + 7$ and at least $k + 7$ vertices from each of $C_1 \setminus S$ and $C_2 \setminus S$ are in $G[S \cup M]$. Clearly, $C_1 \setminus S$ and $C_2 \setminus S$ intersect in at most one vertex, let us denote it $u$, and the other vertices in the intersection of $C_1$ and $C_2$ are in $S$. Let $v$ be arbitrary vertex in $C_1 \cap C_2 \cap S$. Note that $v$ is adjacent to at least 7 vertices in both $C_1 \setminus S$ and $C_2 \setminus S$ and by Lemma 4 it is adjacent to all vertices in $(C_1 \cup C_2) \setminus S$. Since $G - (S \setminus \{v\})$ is a line graph, it follows that $G[(C_1 \cup C_2) \setminus (S \setminus \{v\})]$ is a line graph. Every vertex in $C_1 \setminus (S \cup \{u\})$ is in exactly one other clique in $\mathcal{C}$. This clique intersects $C_2 \setminus (S \cup \{u\})$ in at most one vertex. Therefore, there is a pair of vertices $w_1 \in C_1 \setminus (S \cup \{u\})$, $w_2 \in C_2 \setminus (S \cup \{u\})$ such that $w_1 w_2 \notin E(G)$. Now $uvw_1$ and $uvw_2$ are two odd triangles (any vertex in $C_i \setminus (S \cup \{u, w_i\})$ is adjacent to three vertices of the triangle $uvw_i$) that share a common edge, however $uvw_1w_2$ is not a $K_4$. Hence, $G[(C_1 \cup C_2) \setminus (S \setminus \{v\})]$ is not a line graph, a contradiction. It follows that if two cliques in $\mathcal{C}$ of size at least $k + 7$ intersect in a vertex in $G - S$, then no vertex in $S$ is adjacent to both cliques and consequently no two cliques in $\mathcal{C}'_A$ intersect in at least two vertices.

It follows that $\mathcal{C}'_A$ is indeed a clique partition witness for $G - A$ and by point 2. in Theorem 2, $G - A$ is indeed a line graph. ◀

We are now ready to proof Theorem 1.

▶ **Theorem 1.** Line-Graph Edge Deletion *admits a kernel with* $\mathcal{O}\left(k^5\right)$ *vertices.*

**Proof.** We start the algorithm by finding the set $S$ of at most $6k$ vertices such that for every $v \in S$ the graph $G - (S \setminus \{v\})$ is a line graph. This is simply done by greedily finding maximal set of pairwise edge-disjoint forbidden induced subgraphs. Afterwards, we construct a clique partition witness $\mathcal{C}$ for $G - S$ by using the algorithm of Lemma 3. Finally, we apply Reduction Rules 1, 2, and 3 in this order. By the discussion above Reduction Rule 3, after applying all the reduction rules, the resulting instance has $\mathcal{O}\left(k^5\right)$ vertices. The correctness of the kernelization algorithm follows from the safeness proofs of the reduction rules. ◀

## References


1 N. R. Aravind, R. B. Sandeep, and Naveen Sivadasan. Dichotomy results on the hardness of h-free edge modification problems. *SIAM J. Discrete Math.*, 31(1):542–561, 2017. doi:10.1137/16M1055797.

2 Lowell W. Beineke. Characterizations of derived graphs. *Journal of Combinatorial Theory*, 9(2):129 – 135, 1970. URL: http://www.sciencedirect.com/science/article/pii/S0021980070800199, doi:10.1016/S0021-9800(70)80019-9.

3 Hans L Bodlaender, Leizhen Cai, Jianer Chen, Michael R Fellows, Jan Arne Telle, and Dániel Marx. Open problems in parameterized and exact computation-iwpec 2006. *UU-CS*, 2006, 2006.

4 Leizhen Cai. Fixed-parameter tractability of graph modification problems for hereditary properties. *Inf. Process. Lett.*, 58(4):171–176, 1996. doi:10.1016/0020-0190(96)00050-6.

5 Leizhen Cai and Yufei Cai. Incompressibility of h-free edge modification problems. *Algorithmica*, 71(3):731–757, 2015. doi:10.1007/s00453-014-9937-x.

6 Maria Chudnovsky and Paul Seymour. Claw-free graphs. iv. decomposition theorem. *Journal of Combinatorial Theory, Series B*, 98(5):839 – 938, 2008. URL: http://www.sciencedirect.com/science/article/pii/S0095895607000792, doi:https://doi.org/10.1016/j.jctb.2007.06.007.

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

8 Marek Cygan, Marcin Pilipczuk, Michal Pilipczuk, Erik Jan van Leeuwen, and Marcin Wrochna. Polynomial kernelization for removing induced claws and diamonds. *Theory Comput. Syst.*, 60(4):615–636, 2017. doi:10.1007/s00224-016-9689-x.

9 Daniele Giorgio Degiorgi and Klaus Simon. A dynamic algorithm for line graph recognition. In Manfred Nagl, editor, *Graph-Theoretic Concepts in Computer Science, 21st International Workshop, WG '95, Aachen, Germany, June 20-22, 1995, Proceedings*, volume 1017 of *Lecture Notes in Computer Science*, pages 37–48. Springer, 1995. doi:10.1007/3-540-60618-1_64.

10 R. Diestel. *Graph Theory, 4th Edition*. Springer, 2012.

11 Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

12 Eduard Eiben, William Lochet, and Saket Saurabh. A polynomial kernel for paw-free editing. *CoRR*, abs/1911.03683, 2019. URL: http://arxiv.org/abs/1911.03683, arXiv:1911.03683.

13 Ehab S. El-Mallah and Charles J. Colbourn. The complexity of some edge deletion problems. *IEEE Transactions on Circuits and Systems*, 35(3):354–362, 1988. doi:10.1109/31.1748.

14 Paul Erdös and Richard Rado. Intersection theorems for systems of sets. *Journal of the London Mathematical Society*, 1(1):85–90, 1960.

15 Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2006. doi:10.1007/3-540-29953-X.

16 Sylvain Guillemot, Frédéric Havet, Christophe Paul, and Anthony Perez. On the (non-)existence of polynomial kernels for $P_l$-free edge modification problems. *Algorithmica*, 65(4):900–926, 2013. doi:10.1007/s00453-012-9619-5.

17 Frank Harary. *Graph theory*. Addison-Wesley series in mathematics. Addison-Wesley Pub. Co., 1969.

18 Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-sat. *J. Comput. Syst. Sci.*, 62(2):367–375, 2001. doi:10.1006/jcss.2000.1727.

19 Stefan Kratsch and Magnus Wahlström. Two edge modification problems without polynomial kernels. *Discrete Optimization*, 10(3):193–199, 2013. doi:10.1016/j.disopt.2013.02.001.

20 John M. Lewis and Mihalis Yannakakis. The node-deletion problem for hereditary properties is np-complete. *J. Comput. Syst. Sci.*, 20(2):219–230, 1980. doi:10.1016/0022-0000(80)90060-4.

21 Najiba Sbihi. Algorithme de recherche d'un stable de cardinalite maximum dans un graphe sans etoile. *Discrete Mathematics*, 29(1):53 – 76, 1980. URL: http://www.sciencedirect.com/science/article/pii/0012365X9090287R, doi:https://doi.org/10.1016/0012-365X(90)90287-R.

22 Hassler Whitney. Congruent graphs and the connectivity of graphs. *American Journal of Mathematics*, 54(1):150–168, 1932. URL: http://www.jstor.org/stable/2371086.

23 Mihalis Yannakakis. Edge-deletion problems. *SIAM J. Comput.*, 10(2):297–309, 1981. doi:10.1137/0210021.

# A Proof of Lemma 3

▶ **Lemma 3.** *Given a graph $G$, there is an algorithm that in time $\mathcal{O}(|E(G)| + |V(G)|)$ decides whether $G$ is a line graph and if so, constructs a clique partition witness for $G$.*

**Proof.** The algorithm by Degiorgi and Simon construct the input graph $G$ by adding vertices one at a time, at each step it chooses a vertex to add that is already adjacent to at least one previously-added vertex. That is it construct graphs $G_1, G_2, \ldots, G_n = G$ such that $G_i$ is a connected subgraph of $G$ on $i$ vertices. At each step it maintains a graph $H_i$ such that $G_i$ is a line graph of $H_i$. In here, we can actually keep a clique partition witness $\mathcal{C}_i$ for $G_i$ such that there is a bijection $\varphi_i$ between vertices of $H_i$ and clique in $\mathcal{C}_i$ such that $uv \in E(H_i)$ if and only if $|\varphi_i(u) \cap \varphi_i(v)| = 1$.

The algorithm heavily relies on the Whitney's isomorphism theorem that implies that if the underlying graph of $G_i$ has at least 4 vertices, then the underlying graph $H_i$ is unique up to isomorphism. When adding a vertex $v$ to a graph $G_i$ for $i \leq 4$, the algorithm simply brute-forces the possibilities for $H_i$ and $\mathcal{C}_i$.

When adding a vertex $v$ to $G_i$ when $i > 4$, let $S$ be the subgraph of $H_i$ formed by the edges that correspond to the neighbors of $v$ in $G_i$. Check that $S$ has a vertex cover consisting of one vertex or two non-adjacent vertices, *i.e.*, there are cliques $C_1$ and $C_2$ in $\mathcal{C}_i$ with $C_i \cap C_2 = \emptyset$ and $S \subseteq C_1 \cap C_2$. If there are two vertices in the cover, add an edge (corresponding to $v$) that connects these two vertices in $H_i$ and add $v$ to both $C_1$ and $C_2$. If there is only one vertex $u$ in the cover, then add a new vertex to $H_i$, adjacent to this vertex, add $v$ to the clique $\varphi_i(u)$ in $\mathcal{C}_i$ and add a new clique $\{v\}$ to $\mathcal{C}_i$ to create $\mathcal{C}_{i+1}$. ◀