Difference Determines the Degree: Structural Kernelizations of Component Order Connectivity

Sriram Bhyravarapu
The Institute of Mathematical Sciences, HBNI, Chennai, India

Satyabrata Jana
The Institute of Mathematical Sciences, HBNI, Chennai, India

Saket Saurabh
The Institute of Mathematical Sciences, HBNI, Chennai, India
University of Bergen, Norway

Roohani Sharma
Max Planck Institute for Informatics, Saarland Informatics Campus, Saarbrücken, Germany

Abstract

We consider the question of polynomial kernelization of a generalization of the classical Vertex Cover problem parameterized by a parameter that is provably smaller than the solution size. In particular, we focus on the c-Component Order Connectivity problem (c-COC) where given an undirected graph G and a non-negative integer t, the objective is to test whether there exists a set S of size at most t such that every component of $G - S$ contains at most c vertices. Such a set S is called a c-coc set. It is known that c-COC admits a kernel with $\mathcal{O}(ct)$ vertices. Observe that for $c = 1$, this corresponds to the Vertex Cover problem.

We study the c-Component Order Connectivity problem parameterized by the size of a d-coc set (c-COC/d-COC), where $c, d \in \mathbb{N}$ with $c \leq d$. In particular, the input is an undirected graph G, a positive integer t and a set M of at most k vertices of G, such that the size of each connected component in $G - M$ is at most d. The question is to find a set S of vertices of size at most t, such that the size of each connected component in $G - S$ is at most c. In this paper, we give a kernel for c-COC/d-COC with $\mathcal{O}(k^{d-c+1})$ vertices and $\mathcal{O}(k^{d-c+2})$ edges. Our result exhibits that the difference in d and c, and not their absolute values, determines the exact degree of the polynomial in the kernel size.

When $c = d = 1$, the c-COC/d-COC problem is exactly the Vertex Cover problem parameterized by the solution size, which has a kernel with $\mathcal{O}(k)$ vertices and $\mathcal{O}(k^2)$ edges, and this is asymptotically tight [Dell & Melkebeek, JACM 2014]. We also show that the dependence of $d - c$ in the exponent of the kernel size cannot be avoided under reasonable complexity assumptions.

2012 ACM Subject Classification Theory of computation → Fixed parameter tractability

Keywords and phrases Kernelization, Component Order Connectivity, Vertex Cover, Structural Parameterizations

Digital Object Identifier 10.4230/LIPIcs.IPEC.2023.5

Funding *Sriram Bhyravarapu*: Supported by the SERB-DST via grant PDF/2021/003452.
Saket Saurabh: Supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 819416), and Swarnajayanti Fellowship (No. DST/SJF/MSA01/2017-18).

1 Introduction

The design of parameterized algorithms and kernelization has traditionally relied on the size of the solution as a crucial parameter. Nonetheless, when a problem is established as fixed-parameter tractable based on the solution size, it becomes natural to explore the problem using a parameter that is provably smaller than the solution size.

Over the past decade, several interesting papers have explored these inquiries, particularly in the realm of kernelization [2, 7, 9–12, 15]. Notable contributions in this area include polynomial kernels for the Vertex Cover problem parameterized by the *feedback vertex set* [10] and the *odd cycle transversal* [12]. Hols, Kratsch, and Pieterse [9] provide a comprehensive perspective on most of the aforementioned structural kernelization of Vertex Cover. Additionally, kernelization of Vertex Cover with respect to above-guarantee parameters has also been studied [11]. More recently, in another direction of work, Bougeret, Jansen & Sau, gave a characterization for which structural parameters, that serve as modulators of minor-closed graph classes, Vertex Cover admits polynomial kernels [1].

In this paper, we consider a generalized version of the Vertex Cover problem known as the c-Component Order Connectivity (c-COC) problem. In the c-COC problem, we are given a graph G and an integer t, and the objective is to identify a set of at most t vertices, say S, such that the size of each connected component of $G - S$ is at most c. Such a set S is referred to as a c-coc set. It is worth noting that when c equals 1, the c-COC problem is equivalent to the Vertex Cover problem. The current best-known kernel for the Vertex Cover problem parametrized by solution size (t) consists of $2t - c\log t$ vertices [14] for all $c > 0$. Although previously there was a kernel for Vertex Cover with $\mathcal{O}(t)$ vertices and $\mathcal{O}(t^2)$ edges [4] which is asymptotically best. For c-COC we can obtain a simple kernel with $\mathcal{O}((t + c)t)$ vertices by deleting vertices of degree at least $t + c$, iteratively. Kumar and Lokshtanov [13] designed a kernel with $2ct$ vertices running in time $n^{\mathcal{O}(c)}$. Finally, Xiao [18] obtained a kernel with $9ct$ vertices running in time $n^{\mathcal{O}(1)}$. Here, the polynomial in the running time does not depend on c.

Observe that when $d \geq c$, the size of a d-coc set is at most the size of a c-coc set. This observation leads us to a natural hierarchy of parameterized problems known as c-COC parameterized by a d-coc set, where c and d are positive integers satisfying $c \leq d$. We refer to these parameterized problems as c-COC/d-COC.

> c-COC/d-COC **Parameter:** k
> **Input:** An undirected graph G, an integer t, a set $M \subseteq V(G)$ such that $|M| \leq k$ and for each connected component C of $G - M$, $|C| \leq d$
> **Question:** Does there exist a set $S \subseteq V(G)$ such that $|S| \leq t$ and for each connected component C' of $G - S$, $|C'| \leq c$?

It is natural to ask how do we get the modulator. Either, we can assume that it is given as part of the input or we can obtain a set M of size at most $(d + 1)\mathsf{opt}$, where opt is the size of a smallest d-coc set. Indeed, start with an empty M, and while M is not a d-coc set, greedily select a arbitrary connected subgraph with $d + 1$ vertices and include each of these $d + 1$ vertices into M and remove them from the graph. Thus, from now onwards we assume that M is given as part of the input.

Our main result shows that c-COC/d-COC admits a polynomial kernel with $\mathcal{O}(k^{d-c+1})$ vertices and $\mathcal{O}(k^{d-c+2})$ edges, where k is the size of M, a d-coc set. Notably, our result establishes that the degree of the polynomial in the kernel size is solely determined by the difference between d and c, rather than the specific values of d and c. To illustrate, both 5-COC/7-COC and 23-COC/25-COC exhibit kernels of size $\mathcal{O}(k^3)$. The formal statement of our main result is presented in Theorem 1.

▶ **Theorem 1.** c-COC/d-COC *admits a kernel with* $\mathcal{O}(k^{d-c+1})$ *vertices and* $\mathcal{O}(k^{d-c+2})$ *edges.*

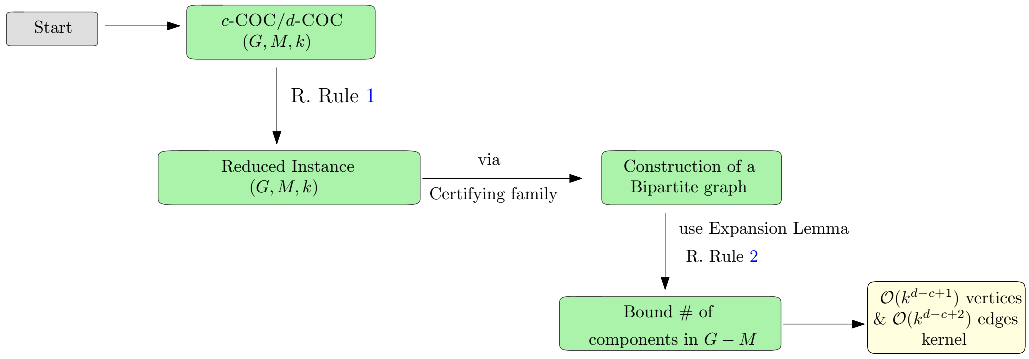

Figure 1 A summary of the main steps of our kernelization.

Note that when $c = d = 1$, the c-COC/d-COC problem corresponds to the Vertex Cover problem parameterized by the size of the solution. In this scenario, our result is asymptotically consistent with the best-known bounds for Vertex Cover.

In the light of Theorem 1, the subsequent question arises as to whether the exponent of k in the kernel size can be made a constant value and be independent of $d - c$. Specifically, does c-COC/d-COC admit a uniform kernel of size $f(d, c)k^{\mathcal{O}(1)}$, where f is a function that only depends on d and c? However, we demonstrate that this is not possible. In particular, we establish that Vertex Cover/d-COC does not admit a kernel of size $\mathcal{O}(k^{d-\epsilon})$ for any $\epsilon > 0$ and positive integer d. This result is precisely formalized in Theorem 2. Vertex Cover therefore does not admit a kernel that is uniformly polynomial in the value of d. The phenomenon that the degree of the kernel size for Vertex Cover has to increase when using smaller and smaller structural parameterization is well-known [8].

▶ **Theorem 2.** *For every $\epsilon > 0$ and every positive integer d,* Vertex Cover/d-COC *has no compression of vertex size $\mathcal{O}(k^{d-\epsilon})$ unless* co-NP $\subseteq$ NP/poly*.*

Our methods

In order to construct a kernel for c-COC/d-COC, our algorithm employs the Expansion Lemma, a combinatorial tool that played a crucial role in developing a quadratic kernel for the Feedback Vertex Set problem. Given an input instance (G, M, k, t) of c-COC/d-COC, we generate sets of “certifying families” for every subset $T \subseteq M$ that correspond to certain components in $G - M$. In particular, the idea is to understand the following. Suppose we do not include any vertex from T in our solution. Then for which components C of $G - M$ do we need to select a strictly larger number of vertices than what is required to *locally* solve the problem in C. These components (in fact, a subset of these) are part of a certifying family corresponding to T. By utilizing these certifying families, we construct a bipartite graph and apply the Expansion Lemma to identify an irrelevant component in $G - M$. Through repeated applications of the Expansion Lemma, we can upper bound the number of components in $G - M$ by $\mathcal{O}(k^{d-c+1})$. Since the size of M is at most k, and each component in $G - M$ contains at most d vertices, we can bound the number of vertices in the kernel to $\mathcal{O}(d \cdot k^{d-c+1} + k)$ and the number of edges to $\mathcal{O}(d^2 \cdot k^{d-c+2} + k^2)$. A summary of the key steps in our kernelization algorithm is provided in Figure 1. Moreover, our lower bound results are established through a parameter-preserving reduction from the d-SAT problem.

2 Preliminaries

In this paper, we consider finite, undirected graphs. For a graph G, we use $V(G)$ and $E(G)$ to refer to its vertex and edge sets, respectively. By $|G|$ we denote the number of vertices in G. We use $\texttt{comp}(G)$ to denote the size of the largest component in G, defined as $\texttt{comp}(G) = \max\{|V(C)| : C \text{ is a component of } G\}$. Thus, an n-vertex graph G is connected if and only if $\texttt{comp}(G) = n$. Given two vertex-disjoint sets X and Y from $V(G)$, the set $N_X(Y) = N(Y) \cap X$ represents the subset of vertices in X that has at least one neighbour in Y. For any positive integer ℓ and a subgraph $H \subseteq G$, the *ℓ-component order connectivity* of H, denoted as $\texttt{coc}_\ell(H)$, is defined as the size of the minimum set $X \subseteq V(H)$ such that $\texttt{comp}(H - X) \leq \ell$. In other words, we have $\texttt{coc}_\ell(H) = \min\{|X| : \texttt{comp}(H - X) \leq \ell, X \subseteq V(H)\}$. We use $[q]$ to denote the set $\{1, \ldots, q\}$. For details on parameterized complexity, kernelization, and compression we refer to the textbooks [3] and [6].

3 Kernelization

We represent an instance of c-COC/d-COC as (G, M, k, t), where G is a graph, $M \subseteq V(G)$ is a subset of vertices with size at most k, and $\texttt{comp}(G - M) \leq d$. Recall that the problem seeks to determine whether there exists a set $S \subseteq V(G)$ of size at most t such that $\texttt{comp}(G - S) \leq c$. We use $\mathcal{C}$ to denote the set of all components in $G - M$. For any component C in $\mathcal{C}$, since $|C| \leq d$, we have $\texttt{coc}_c(C) \leq d - c$, where $\texttt{coc}_c(C)$ denotes the size of the smallest vertex set $X \subseteq V(C)$ such that $\texttt{comp}(C - X) \leq c$. Let $\mathcal{C}_\ell = \{C \mid C \in \mathcal{C} \text{ and } \texttt{coc}_c(C) = \ell\}$ denote the set of components C in $G - M$ for which the size of a smallest c-coc is ℓ. It is possible for the set $\mathcal{C}_\ell$ to be empty. Note that for each $C \in \mathcal{C}$, $\texttt{coc}_c(C) \leq d - c$. Consequently, for each $\ell > d - c$, we have $\mathcal{C}_\ell = \emptyset$. In the subsequent section, we show that the number of components in $\mathcal{C}_\ell$, for any $\ell \in \{0, 1, \ldots, d - c\}$, can be upper bounded by $\mathcal{O}(k^{d-c+1})$ (after the application of certain reduction rules). Once this is accomplished the bounds on the number of vertices and edges in the kernel follow immediately, as each component has at most d vertices. Hence, in the remaining we focus on bounding the size of each set $\mathcal{C}_\ell$.

Note that the family $\mathcal{C}_\ell$ can be constructed in polynomial time. Indeed, for each component C of $G - M$, the value of $\texttt{coc}_c(C)$ can be computed in $2^{|C|} \cdot |C|^{\mathcal{O}(1)}$ time by considering all possible subsets of C as c-coc sets. Given that $|C| \leq d$ for each connected component C, the computation of $\texttt{coc}_c(C)$ can be done in time that only depends on d (which is a constant). Recall that (G, M, k, t) represents an instance of c-COC/d-COC. Consider a vertex set $T \subseteq M$ and a component $C \in \mathcal{C}_\ell$. We use $\texttt{local}(T, C)$ to denote the size of the smallest set $X \subseteq V(C)$ such that $\texttt{comp}(C - X) \leq c$ and $N_C(T) \subseteq X$, where $N_C(T) = N(T) \cap V(C)$. Informally, $\texttt{local}(T, C)$ represents the size of the smallest solution corresponding to $\texttt{coc}_c(C)$ in $G[C]$ that must include all the neighbors of T in C. Notably, for any pair T and C, the value of $\texttt{local}(T, C)$ can be computed in $2^{|C|} \cdot |C|^{\mathcal{O}(1)}$ time by examining all subsets of C that are supersets of the neighborhood of T in C, considering them as solution sets of $G[C]$, and determining the minimum possible set among them. Since $\texttt{coc}_c(C) = \ell$ for each component $C \in \mathcal{C}_\ell$, we make the following observation.

▶ **Observation 3.** *For each pair (T, C) where $T \subseteq M$ and $C \in \mathcal{C}_\ell$, we have $\texttt{local}(T, C) \geq \ell$.*

Certifying family for a fixed ℓ. Let $\mathcal{T} = \{T \mid T \subseteq M \text{ and } |T| \leq \ell + 1\}$ denote the set of all subsets of M of size at most $(\ell + 1)$. We refer to each $T \in \mathcal{T}$ as an *unordered tuple* of size $|T|$. For every $T \in \mathcal{T}$, we define a set of components $\mathcal{F}_T$ associated with T as

$\mathcal{F}_T = \{C : C \in \mathcal{C}_\ell,\ \texttt{local}(T, C) > \ell\}$. We refer to such a family $\mathcal{F}_T$ as a *certifying family* for T. Essentially, if $C \in \mathcal{F}_T$, then there exists *no solution* of $G[C]$ corresponding to $\texttt{coc}_c(C)$ of size ℓ that includes all the neighbors of T in C.

Given two disjoint vertex sets V_1 and V_2, the boundary of V_1 with respect to V_2, denoted by $\texttt{bdry}_{V_2}(V_1)$, is defined as the set $N(V_2) \cap V_1$. We present two key lemmas that are crucial for our analysis.

▶ **Lemma 4.** *Let $C \in \mathcal{C}_\ell$ such that $|\texttt{bdry}_M(C)| \geq \ell + 1$. Then there exists $T \in \mathcal{T}$ satisfying $C \in \mathcal{F}_T$.*

Proof. Since $C \in C_\ell$, an optimal c-coc set in $G[C]$ has size ℓ but each c-coc set containing $N_G(T) \cap C$ has size more than ℓ because $|N_C(T)| > \ell$. Now consider an arbitrary set $C^* \subseteq \texttt{bdry}_M(C)$ of size $(\ell+1)$. Let $U \subseteq N_M(C)$ be a set containing a neighbor of each vertex in C^* (arbitrarily select a neighbor of each vertex in C^*). Clearly, the size of $|U| \leq \ell + 1$ and $C^* \subseteq N_C(U)$. According to the definition of a certifying family, the component C is associated with U.

This concludes the proof. ◀

▶ **Lemma 5.** *Let $C \in \mathcal{C}_\ell$ be a component with the property that $|\texttt{bdry}_M(C)| \leq \ell$. Then either there exists a tuple $T \in \mathcal{T}$ such that $C \in \mathcal{F}_T$, or there exists a vertex set $U \subseteq V(C)$ with $|U| = \ell$, satisfying $\texttt{bdry}_M(C) \subseteq U$ and $\texttt{comp}(C - U) \leq c$.*

Proof. Consider the vertex set $N_M(C) \subseteq M$. Let $X \subseteq N_M(C)$ be a set containing a neighbor of each vertex in $\texttt{bdry}_M(C)$ (arbitrarily select a neighbor of each vertex in $\texttt{bdry}_M(C)$). Clearly, the size of $|X| \leq |\texttt{bdry}_M(C| \leq \ell$. Furthermore, we have $N_C(X) = \texttt{bdry}_M(C)$.

If C belongs to the certifying family $\mathcal{F}_X$, then our assertion is proven. So we assume that $C \notin \mathcal{F}_X$. According to the definition of $\mathcal{F}_X$, this implies $\texttt{local}(X, C) = \ell$. Therefore, there exists a solution U associated with $\texttt{local}(X, C)$ such that $U \subseteq V(C)$, $|U| = \ell$, $\texttt{bdry}_M(C) \subseteq U$, and $\texttt{comp}(C - U) \leq c$. ◀

Lemmas 4 and 5, essentially, say that a component C of $G - M$ is not in any certifying family if there exists a minimum size local solution for the component C that contains all the boundary vertices ($\texttt{bdry}_M(C)$). This observation leads to the following reduction rule.

▶ **Reduction Rule 1.** *Consider a component $C \in \mathcal{C}_\ell$ for which there is no tuple $T \in \mathcal{T}$ that satisfies that $C \in \mathcal{F}_T$. Then we remove C from the graph G and reduce the value of t by ℓ. The resulting instance is $(G - C, M, k, t - \ell)$.*

To apply Reduction Rule 1 finding such a component takes $k^{\ell+1} \cdot 2^d \cdot n^{\mathcal{O}(1)}$ time. The correctness of the Reduction Rule 1 follows from the following Lemma 6.

▶ **Lemma 6.** *Reduction Rule 1 is safe.*

Proof. The forward direction is straightforward. Let S be a solution to the instance (G, M, k, t). Since $G - C$ is a subgraph of G and $\texttt{coc}_c(C) = \ell$, there are at least ℓ vertices of C in S. Therefore, $S \setminus C$ is a solution for $(G - C, M, k, t - \ell)$.

In the backward direction, let S_1 be a solution to $(G - C, M, k, t - \ell)$. We aim to show that there exists a vertex set $Z \subseteq V(C)$ such that $|Z| = \ell$ and $S \cup Z$ is a solution to (G, M, k, t). We consider two cases based on the size of $\texttt{bdry}_{M \setminus S_1}(C)$.

Case 1. $\texttt{bdry}_{M \setminus S_1}(C) \geq \ell + 1$. Based on Lemma 4, there exists $T \in \mathcal{T}$ such that $T \subseteq M$, $|T| \leq (\ell + 1)$, and $C \in \mathcal{F}_T$. This contradicts our assumption that there is no $T \in \mathcal{T}$ satisfying $C \in \mathcal{F}_T$.

Case 2. $\texttt{bdry}_{M \setminus S_1}(C) \leq \ell$. Based on Lemma 5, we have two possibilities: either there exists $T \in \mathcal{T}$ such that $|T| \subseteq M \setminus S_1$, $|T| \leq (\ell+1)$, and $C \in \mathcal{F}_T$, or there is a vertex set $U \subseteq V(C)$ satisfying $|U| = \ell$, $\texttt{bdry}_{M \setminus S_1}(C) \subseteq U$, and $\texttt{comp}(C - U) \leq c$. However, based on our assumption, the former condition cannot occur. Therefore, there must exist a vertex set $U \subseteq V(C)$ satisfying $|U| = \ell$, $\texttt{bdry}_{M \setminus S_1}(C) \subseteq U$, and $\texttt{comp}(C - U) \leq c$. In this case, we set $Z = U$.

This completes the proof. ◀

Expansion Lemma. From now onwards, we assume that we have an instance (G, M, k, t) of c-COC/d-COC, on which we have applied Reduction Rule 1 exhaustively. Now, we bound the number of components in $\mathcal{C}_\ell$ using the *expansion lemma* in strengthened form of [16]. Let us first recall the definition of expansion and the expansion lemma.

▶ **Definition 7** (q-expansion [3]). Let H be a bipartite graph with vertex bipartition (X, Y) and q be a positive integer. A set of edges $E^* \subseteq E(H)$ is called a *q-expansion of X into Y* if (i) each vertex of X is incident with exactly q edges of E^*, and (ii) E^* saturates exactly $q|X|$ vertices in Y.

▶ **Lemma 8** (Expansion Lemma [[3], Lemma 2.18]). *Let $q \in \mathbb{N}$ and G be a bipartite graph with vertex bipartition (P, Q) such that $|Q| > q \cdot |P|$ and there are no isolated vertices in Q. Then there exist nonempty vertex sets $X \subseteq P$ and $Y \subseteq Q$ such that (i) X has a q-expansion E^* into Y, (ii) no vertex in Y has a neighbor outside X. Furthermore, two such sets X and Y and such vertex w can be found in the time that is polynomial in the size of G.*

Next we mention q–Expansion Lemma given by Fomin et al. [5] which is a generalization of a result due to Thomass´e [[17], Theorem 2.3].

▶ **Lemma 9** (The q-Expansion Lemma [[5], Lemma 5.1]). *Let $q \in \mathbb{N}$ and G be a bipartite graph with vertex bipartition (P, Q) such that $|Q| > q \cdot t$, where t is the size of a maximum matching in G, and there are no isolated vertices in Q. Then there exist nonempty vertex sets $X \subseteq P$ and $Y \subseteq Q$ such that (i) X has a q-expansion E^* into Y, (ii) no vertex in Y has a neighbor outside X. Furthermore, two such sets X and Y and such vertex w can be found in the time that is polynomial in the size of G.*

For our purpose we use the expansion lemma in strengthened form given by Philip et al. [16] which is following.

▶ **Lemma 10** (Strong q-Expansion Lemma [[16], Lemma 5]). *Let $q \in \mathbb{N}$ and G be a bipartite graph with vertex bipartition (P, Q) such that $|Q| > q \cdot t$, where t is the size of a maximum matching in G, and there are no isolated vertices in Q. Then there exist nonempty vertex sets $X \subseteq P$ and $Y \subseteq Q$ such that (i) X has a q-expansion E^* into Y, (ii) no vertex in Y has a neighbor outside X, and (iii) there is a vertex $w \in Y$ such that w is not incident to any edge in E^* (or, E^* does not saturate w). Furthermore, two such sets X and Y and such vertex w can be found in the time that is polynomial in the size of G.*

Note that the statement of Lemma 10 remains valid even for $|Q| > q \cdot |P|$, as $|P| \geq t$. Now, in order to apply the expansion lemma, we first construct an auxiliary bipartite graph where this lemma is applied.

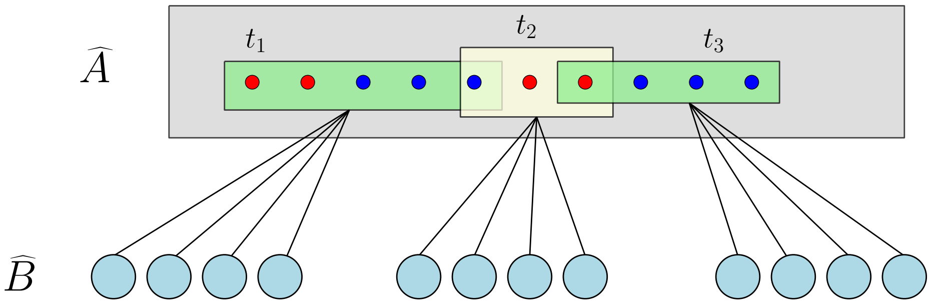

Figure 2 An example of 4-expansion from $\widehat{A}$ into $\widehat{B}$. t_1, t_2, t_3 represents vertices of corresponding tuples in $\mathcal{T}_{\widehat{A}}$. The red-colored vertices denote the solution vertices from the modulator.

Construction of an auxiliary bipartite graph $H = (A, B)$. Let us recall the set $\mathcal{T}$, which consists of all subsets of M with size at most $\ell + 1$, and the corresponding certifying families $\{\mathcal{F}_T : T \in \mathcal{T}\}$. We will now construct a bipartite graph H with vertex partitions A and B using the following procedure:

- For each tuple $T_i \in \mathcal{T}$, we introduce a vertex t_i in the part A.
- For each component $C_j \in \mathcal{C}_\ell$, we include a vertex c_j in the part B.
- For each pair of vertices $t_i \in A$ and $c_j \in B$, we add an edge $t_i c_j$ in H if and only if C_j belongs to the certifying family $\mathcal{F}_{T_i}$.

We are now ready to give the reduction rule. From this point onwards, we fix the following value for q.

$$\boxed{q = (\ell + 2) + c(\ell + 1)}$$

Consider the bipartite graph $H = (A, B)$ that was constructed above. It is important to note that if the instance (G, M, k, t) is reduced using Reduction Rule 1, then there are no isolated vertices in the vertex set B.

▶ **Reduction Rule 2.** *If $|B| > q \cdot |A|$, then call the algorithm provided by the Expansion Lemma to compute sets $\widehat{A} \subseteq A$ and $\widehat{B} \subseteq B$ such that*

- *no vertex in $\widehat{B}$ has a neighbor outside $\widehat{A}$, i.e., $N(\widehat{B}) \subseteq \widehat{A}$,*
- *there is a q-expansion $\widehat{E}$ from $\widehat{A}$ into $\widehat{B}$, and*
- *there is a vertex $b \in \widehat{B}$ such that b is not incident with $\widehat{E}$.*

Consider the component $C \in \mathcal{C}_\ell$ corresponding to the vertex b in B. Then we remove C from the graph G and reduce the value of t by ℓ. The resulting instance is $(G - C, M, k, t - \ell)$.

Before analyzing the safeness of Reduction Rule 2, we look at the following lemma.

▶ **Lemma 11.** *Suppose $\widehat{E}$ represents a q-expansion from $\widehat{A}$ into $\widehat{B}$, and let b be a vertex in $\widehat{B}$ that satisfies the condition of Reduction Rule 2. Further, let S_1 be a solution to the problem $(G-C, M, k, t-\ell)$. Then, there exists another solution S_2 of $(G-C, M, k, t-\ell)$ that satisfies the following properties: (i) $|S_2| \leq |S_1|$; (ii) for each vertex t_i in $\widehat{A}$, S_2 intersects with the corresponding vertex set $T_i \subseteq M$, that is $S_2 \cap T_i \neq \emptyset$.*

Proof. Let $\mathcal{T}_{\widehat{A}}$ be the set of tuples corresponding to the vertices in $\widehat{A}$. Let $\mathcal{T}_1$ be a subset of $\mathcal{T}_{\widehat{A}}$, containing those tuples T for which T intersects with S_1 (i.e., $T \cap S_1 \neq \emptyset$). On the other hand, let $\mathcal{T}_2$ be defined as the set of tuples in $\mathcal{T}_{\widehat{A}}$ that are not in $\mathcal{T}_1$, i.e., $\mathcal{T}_2 = \mathcal{T}_{\widehat{A}} \setminus \mathcal{T}_1$. In other words, $\mathcal{T}_2$ comprises all the tuples T from $\mathcal{T}_{\widehat{A}}$ that satisfy $T \cap S_1 = \emptyset$.

For each vertex t_i in $\widehat{A}$ that corresponds to a tuple in the set $\mathcal{T}_2$, we define two sets: $\mathcal{C}_{1,t_i}$ and $\mathcal{C}_{2,t_i}$. The set $\mathcal{C}_{1,t_i}$ consists of each component C_j in $\mathcal{C}_\ell$ such that $t_i c_j$ is an edge in $\widehat{E}$ and $|V(C_j) \cap S_1| = \ell$. The set $\mathcal{C}_{2,t_i}$ consists of each component C_j in $\mathcal{C}_\ell$ such that $t_i c_j$ is an edge in $\widehat{E}$ and $|V(C_j) \cap S_1| \geq \ell + 1$. For an illustration see Figure 2.

First we show that $|\mathcal{C}_{1,t_i}| < c(\ell+1)$. On the contrary, assume that $|\mathcal{C}_{1,t_i}| \geq c(\ell+1)$. For any component C_j belonging to $\mathcal{C}_{1,t_i}$, we observe that $c_j t_i \in \widehat{E}$ implies $C_j \in \mathcal{F}_{T_i}$. Consequently, it follows that $\texttt{local}(T_i, C_j) > \ell$. However, since $C_j \in \mathcal{C}_{1,t_i}$, we have $|C_j \cap S_1| = \ell$. Hence, we conclude that $N_{C_j}(T_i) \backslash S_1 \neq \emptyset$. Thus there always exists a vertex $x \in T_i$ and $y \in (N_{C_j}(T_i) \backslash S_1)$ such that $xy \in E(G)$. As $|\mathcal{C}_{1,t_i}| \geq c(\ell+1)$, we can select a vertex each from each of the components in $\mathcal{C}_{1,t_i}$ and obtain a set Y containing $c(\ell+1)$ vertices from $G - M$, where $Y \cap S_1 = \emptyset$ and each vertex $y \in Y$ has a neighbor in T_1. Since, $|T_i| \leq \ell+1$ (since $T_i \in \mathcal{T}$), and $T_i \cap S_1 = \emptyset$, we can deduce, by applying the pigeon-hole principle, the existence of a vertex x in T_i such that $\deg_{G-C-S_1}(x) \geq c$. Consequently, we have a component of size at least $c+1$ in $G - S_1$. This contradicts the fact that S_1 is a solution for the c-COC/d-COC problem on the instance $(G - C, M, k, t - \ell)$. Therefore, we can conclude that $|\mathcal{C}_{1,t_i}| \leq c(\ell+1)$.

Given that $q = (\ell+2) + c(\ell+1)$, and $|\mathcal{C}_{1,t_i}| \leq c(\ell+1)$, we can deduce that $|\mathcal{C}_{2,t_i}| \geq \ell+2$. We denote the set of all vertices contained in some tuple in $\mathcal{T}_2$ as $V(\mathcal{T}_2)$, defined formally as $V(\mathcal{T}_2) := \{v \mid \exists\, T \in \mathcal{T}_2 : v \in M \cap T\}$. Now, we propose a new solution denoted as S_2. Let $U = \bigcup_i V(\mathcal{C}_{2,t_i})$ and $\mathcal{C}' = \bigcup_i \mathcal{C}_{2,t_i}$.

$$S_2 = \big(S_1 \setminus U\big) \bigcup V(\mathcal{T}_2) \bigcup_{C_j \in \mathcal{C}'} Z_j,$$

Here, $Z_j \subseteq V(C_j)$ represents a set (the exact choice of Z_j is deferred to later in the proof) that corresponds to $\texttt{coc}_c(C_j)$. Here we want to mention that we do not want to choose an arbitrary ℓ-size coc in $G[C_j]$: rather we want to choose one that contains the neighborhood of $(M \setminus (S_1 \cup V(\mathcal{T}_2))$ in C_j. We define a set X to *correspond to* $\texttt{coc}_c(H)$ if $|X| = \texttt{coc}_c(H)$ and $\texttt{comp}(H - X) \leq c$. Recall that $\texttt{coc}_c(C_j) = \ell$.

Towards the proof, we need to establish two conditions: firstly, $|S_2| \leq |S_1|$, and secondly, that S_2 is a solution of $(G - C, M, k, t - \ell)$.

(i) $|S_2| \leq |S_1|$**.** In this comparison, we are examining the sizes of S_1 and S_2. Observe that we are only editing (deleting or adding) vertices that appear in the tuples in $\mathcal{T}_2$ and the components $\mathcal{C}_{2,t_i}$ where t_i corresponds to a specific tuple T_i in $\mathcal{T}_2$. Let us define the size of the solution outside $\mathcal{T}_2$ and $\mathcal{C}_{2,t_i}$ as f. We also define r as the sum of the sizes of $\mathcal{C}_{2,t_i}$ for all relevant tuples $T_i \in \mathcal{T}_2$, i.e., $r = \sum_i |\mathcal{C}_{2,t_i}|$. Since $|\mathcal{C}_{2,t_i}| \geq \ell+2$, we can conclude that $r \geq (\ell+2) \cdot |\mathcal{T}_2|$. Based on these definitions, we can establish that $|S_1| \geq f + r(\ell+1)$ and $|S_2| = f + |V(\mathcal{T}_2)| + r\ell$. Now according to the definition of a certifying family, we have $|V(\mathcal{T}_2)| \leq (\ell+1) \cdot |\mathcal{T}_2|$.
Now,

$$\begin{aligned}
|S_2| &= f + r\ell + |V(\mathcal{T}_2)| \\
&\leq f + r\ell + (\ell+1) \cdot |\mathcal{T}_2| \\
&\leq f + r\ell + (\ell+2) \cdot |\mathcal{T}_2| \\
&\leq f + r\ell + r \\
&\leq f + r(\ell+1) \\
&\leq |S_1|
\end{aligned}$$

(ii) S_2 **is a solution of** $(G - C, M, k, t - \ell)$**.** Next, we show that S_2 serves as a solution for $(G - C, M, k, t - \ell)$. We analyze a component $C_j \in \mathcal{C}_{2,t_i}$. There are two possible scenarios depending on the size of $\texttt{bdry}_{M \setminus S_2}(C_j)$.

- $\texttt{bdry}_{M\setminus S_2}(C_j) \leq \ell$.
  Based on Lemma 5, we can have two possibilities. Either there exists a tuple $T_i \subseteq M \setminus S_2$ such that $|T_i| \leq (\ell+1)$ and C_j belongs to the certifying family $\mathcal{F}_T$, or there exists a vertex set $U \subseteq V(C_j)$ that satisfies the following conditions: $|U| = \ell$, $\texttt{bdry}_{M\setminus S_2}(C_j) \subseteq U$, and $\texttt{comp}(C_j - U) \leq c$.
  - If there is a T_i belonging to $\mathcal{T}$ such that C_j is in $\mathcal{F}_{T_i}$, it and $C_j \in \mathcal{C}_{2,t_i}$ implies that t_i must be a part of $\widehat{A}$. However, this leads to a contradiction because it means T_i has a non-empty intersection with S_2, which contradicts the fact that T_i is a subset of $M \setminus S_2$.
  - If there exists a subset U of the vertex set $V(C_j)$ such that $|U| = \ell$, $\texttt{bdry}_{M\setminus S_2}(C_j) \subseteq U$ and $\texttt{comp}(C_j - U) \leq c$, then we can define the set Z_j as U, which represents the set corresponding to $\texttt{coc}_c(C_j)$.
- $\texttt{bdry}_{M\setminus S_2}(C_j) \geq \ell + 1$.
  Lemma 4 guarantees the existence of $T_i \in \mathcal{T}$ that fulfills the following conditions: $T_i \subseteq M \setminus S_2$, $|T_i| \leq (\ell+1)$, and $C_j \in \mathcal{F}_T$. As a result, t_i must belong to $\widehat{A}$ according to Lemma 10. However, this implies that $T_i \cap S_2$ cannot be empty, which contradicts the fact that T_i is a subset of $M \setminus S_2$.

when $C_j \in \mathcal{C}_{1,t_i}$, the vertices $S_1 \cap C_j \subseteq S_2$ and no neighbor of $C_j \setminus S_1$ have been added to S_1. So we are fine for this case. Hence the proof follows. ◀

The correctness of Reduction Rule 2 follows from the lemma below.

▶ **Lemma 12.** *Reduction Rule 2 is safe.*

Proof. The forward direction is straightforward. Suppose S is a solution to the instance (G, M, k, t). Given that $G - C$ is a subgraph of G and $\texttt{coc}_c(C) = \ell$, we can conclude that S contains at least ℓ vertices from C. Hence, $S \setminus C$ forms a solution for $(G - C, M, k, t - \ell)$.

In the backward direction, let S_1 represent a solution for $(G - C, M, k, t - \ell)$. We will show that there exists a vertex set $Z \subseteq V(C)$ such that $S \cup Z$ forms a solution for (G, M, k, t). Consider the sets $\widehat{A}$, $\widehat{B}$, and $\widehat{E}$ that satisfy the assumptions outlined in Reduction Rule 2. Within these assumptions, there exists a vertex $b \in \widehat{B}$ associated with a vertex $w \in \widehat{A}$. Specifically, $T \subseteq N(C)$ and $\texttt{local}(T, C) \geq \ell + 1$, where C represents the component in $\mathcal{C}_\ell$ corresponding to the vertex b in B, and $T \subseteq M$ is the tuple associated with the vertex w. However, due to the property of $\widehat{E}$, there is no edge $e \in \widehat{E}$ in which w and b are the endpoints. At this point, we invoke the algorithm provided by Lemma 11 to compute the set S_2. As per Lemma 11, for each $t' \in \widehat{A}$, we have $S_2 \cap T' \neq \emptyset$, where $T' \subseteq M$ represents the vertex set associated with the vertex t' in $\widehat{A}$. Depending on the size of $\texttt{bdry}_{M\setminus S_2}(C)$, we encounter two cases.

Case 1. $\texttt{bdry}_{M\setminus S_2}(C) \geq \ell + 1$. By applying Lemma 4, we can establish the existence of $T_i \in \mathcal{T}$ that satisfies the following conditions: $T_i \subseteq M \setminus S_2$, $|T_i| \leq (\ell+1)$, and $C \in \mathcal{F}_T$. Consequently, t_i must belong to $\widehat{A}$ (based on the Expansion Lemma, $b \in \widehat{B}$ and $N(\widehat{B}) \subseteq \widehat{A}$). However, this implies that $T_i \cap S_2 \neq \emptyset$, which contradicts the fact that $T_i \subseteq M \setminus S_2$.

Case 2. $\texttt{bdry}_{M\setminus S_2}(C) \leq \ell$. Using Lemma 5, we can conclude that one of the following two cases holds: Either there exists $T_i \subseteq M \setminus S_2$, where $|T_i| \leq (\ell+1)$ and $C \in \mathcal{F}_T$, or, there exists a vertex set $U \subseteq V(C)$ satisfying $|U| = \ell$, $\texttt{bdry}_{M\setminus S_2}(C) \subseteq U$, and $\texttt{comp}(C - U) \leq c$.

- If there exists $T_i \in \mathcal{T}$ such that $C \in \mathcal{F}_{T_i}$, then it follows that t_i must be in $\widehat{A}$. However, this implies that $T_i \cap S_2 \neq \emptyset$, which contradicts the fact that $T_i \subseteq M \setminus S_2$.

- In the case where there exists a vertex set $U \subseteq V(C)$ that satisfies the conditions $|U| = \ell$, $\texttt{bdry}_{M \setminus S_2}(C) \subseteq U$, and $\texttt{comp}(C - U) \leq c$, we can set Z equal to U.

This completes the proof. ◀

Putting them all together, we get the following theorem.

▶ **Theorem 1.** *c-COC/d-COC admits a kernel with $\mathcal{O}(k^{d-c+1})$ vertices and $\mathcal{O}(k^{d-c+2})$ edges.*

Proof. Consider the instance of the c-COC/d-COC problem denoted as (G, M, k, t). We begin by partitioning all the components $\mathcal{C}$ into a maximum of $d - c$ parts denoted as $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_{d-c}$. Each part $\mathcal{C}_\ell$ is defined as the collection of components C in $G - M$ satisfying $\texttt{coc}_c(C) = \ell$, where $\ell \in [d - c]$. In other words, $\mathcal{C}_\ell$ contains components C for which the minimum-sized set $X \subseteq V(C)$ that guarantees $\texttt{comp}(C - X) \leq c$ is exactly ℓ. Towards solving the problem, we initially focus on each set $\mathcal{C}_\ell$ individually, aiming to reduce the number of components contained within each set.

We exhaustively apply Reduction Rules 1 and 2 to the set $\mathcal{C}_\ell$ for each $\ell \leq d - c$. Each reduction rule is capable of removing at least one vertex from the graph and can be executed in polynomial time. The running time of the algorithm takes into account the time required for applying Reduction Rule 2 as well as constructing the auxiliary bipartite graph, which facilitates the application of the expansion lemma. The construction of the bipartite graph can be accomplished in $k^{d-c+1} \cdot n^{\mathcal{O}(1)}$ time. Consequently, the entire kernelization procedure runs within polynomial time, specifically $n^{\mathcal{O}(1)}$. The output of the algorithm is the resulting instance (G', M', k', t'), which is guaranteed to be a kernel, meaning that no further reduction can be applied to it. The correctness of the algorithm is derived from the proofs establishing the safeness of the reduction rules (Lemmas 6 and 12).

We now argue about the size of the kernel. When Reduction Rules 1 and 2 are not applicable, we can establish that $|B| \leq q \cdot |A|$, where $q = (\ell + 2) + c(\ell + 1)$, and A and B represent the vertex sets of the auxiliary bipartite graph. Recall that each vertex $a \in A$ corresponds to a set of at most $\ell + 1 \leq (d - c) + 1$ vertices from M, and each vertex $b \in B$ corresponds to a set of at most d vertices. Consequently, we have $|A| \leq \mathcal{O}(k^{d-c+1})$ and $|B| \leq \mathcal{O}(k^{d-c+1})$. By combining these bounds, we can deduce that the size of the vertex set in the reduced instance (G', M', k', t') is upper bounded by $\mathcal{O}(k^{d-c+1})$. Additionally, the degree of each vertex in M' is bounded by $k + (d \cdot \mathcal{O}(k^{d-c+1}))$, while the degree of each vertex in $G' - M'$ is bounded by $k + d$. As a result, the size of the edge set in the reduced instance is upper bounded by $\mathcal{O}(k^{d-c+2})$. In conclusion, the size of the vertex set in the reduced instance is upper bounded by $\mathcal{O}(k^{d-c+1})$, and the size of the edge set is upper bounded by $\mathcal{O}(k^{d-c+2})$. Hence the proof follows. ◀

4 Kernel Lower bound

In this section, we show a lower bound for the size of the kernel of the problem we considered in this paper, under some complexity-theoretic assumptions. We prove it by giving a *parameter preserving transformation* from d-CNF-SAT to c-COC/d-COC and using the Vertex Cover kernelization lower bound due to Dell and Van Melkebeek [4].

Given a CNF formula where each clause has at most d literals, the d-CNF-SAT problem asks to find a boolean assignment of values to the variables such that each clause is satisfiable. The following two theorems are known due to Dell and Van Melkebeek [4].

▶ **Theorem 13** (Lower Bound for d-CNF-SAT [4]). *Let $d \geq 3$ be an integer. For any $\epsilon > 0$, the d-CNF-SAT problem parameterized by the number of variables (n) does not admit a polynomial compression with size $\mathcal{O}(n^{d-\epsilon})$, unless co-NP $\subseteq$ NP/poly.*

▶ **Theorem 14** (Lower Bound for Vertex Cover [4]). *For any $\epsilon > 0$, the Vertex Cover problem parameterized by the solution size (k) does not admit a polynomial compression with size $\mathcal{O}(k^{2-\epsilon})$, unless co-NP $\subseteq$ NP/poly.*

▶ **Definition 15** (Parameter preserving transformation (PPT)). *Let Π_1 and Π_2 be two parameterized problems. We say that there exists a parameter preserving transformation from Π_1 to Π_2 if there exists a polynomial time algorithm $\mathcal{B}$ that given an instance (x, k) of Π_1, constructs an instance (x', k') of Π_2 such that*

- *$(x, k) \in \Pi_1$ if and only if $(x', k') \in \Pi_2$, and*
- *$k' \leq \mathcal{O}(k)$.*

Below we provide a parameter preserving transformation from d-CNF-SAT to c-COC/d-COC when $c = 1$. In particular, we prove the following lemma.

▶ **Lemma 16** (Reduction from d-CNF-SAT to Vertex Cover/d-COC). *There exists a parameter preserving transformation from the d-CNF-SAT parameterized by the number of variables to Vertex Cover/d-COC. In the Vertex Cover/d-COC problem, the size of the modulator is twice the number of variables present in the d-CNF-SAT formula.*

Proof. Let Φ be a d-CNF formula, an instance of d-CNF-SAT, consisting of n variables, denoted as $\{x_1, x_2, \ldots, x_n\}$, and m clauses $\{C_1, C_2, \ldots, C_m\}$. Since Φ is a d-CNF formula, each clause contains at most d literals. We construct an instance (G, k, t) for Vertex Cover/d-COC using the following construction:

- For each variable x, we introduce two vertices denoted as x^1 and x^2, and connect them with an edge (x^1, x^2).
- For a clause C_j consisting of d_j literals, we include a clique of size d_j. Within the clique, we label the vertices as follows: a vertex is named $v_{i,j}$ if the literal x_i or $\overline{x_i}$ is present in clause C_j.
- For every $i \in [n], j \in [m]$, if x_i is a literal in clause C_j, we add the edge $(x_i, v_{i,j})$. Similarly, if $\overline{x_i}$ is a literal in clause C_j, we add the edge $(\overline{x_i}, v_{i,j})$.

▷ Claim 17. There exists a vertex subset $S \subseteq V(G)$ of size $2n$ such that $\texttt{comp}(G - S) \leq d$.

Proof. Let S be defined as the set containing elements x_i and $\overline{x_i}$ for all $i \in [n]$. Clearly $|S| = 2n$. Considering the construction of graph G, it can be observed that every component in $G - S$ forms a clique with a maximum size of d. Therefore, we have $\texttt{comp}(G - S) \leq d$. ◁

▷ Claim 18. Φ is satisfiable if and only if G has a vertex cover of size $n - m + \sum_{i=1}^{m} |C_i|$.

Proof. In the forward direction, assuming that Φ is satisfiable, we show the existence of a vertex cover in G with a size of $n - m + \sum_{i=1}^{m} |C_i|$. To construct this vertex cover, we proceed as follows:

- For every variable x, we include x in the set S if it is assigned the value `true` in the satisfying assignment. Otherwise, we add $\overline{x}$ to S.
- For each clause C_j, assuming that variable x_a makes clause C_j satisfiable, we add all vertices from the corresponding clique to S except for the vertex $v_{a,j}$.

It is evident that the set S constructed as described above forms a vertex cover. Furthermore, the size of S is bounded by $n - m + \sum_{i=1}^{m} |C_i|$ due to the construction process.

In the backward direction, let R be a vertex cover of G with a size of $n - m + \sum_{i=1}^{m} |C_i|$. Since any vertex cover in C_i must have size at least $|C_i| - 1$, it follows that for each i, exactly one of x_i and $\overline{x_i}$ is present in R. Now, we construct an assignment β for the variables in Φ as follows: we set x_i to `true` if x_i is in R, and we set x_i to `false` otherwise. Our objective is to show that β satisfies all the clauses. Consider the clique corresponding to clause C_j. We observe that exactly $d_j - 1$ vertices are present in R. Therefore, there exists a vertex, denoted as $v_{a,j}$, which is not in R. However, since R is a vertex cover, it must contain the vertex x_a in order to cover the edge $(v_{a,j}, x_a)$. Since x_a is assigned the value `true`, the literal x_a satisfies the clause C_j. Hence, every clause in Φ is satisfied by the assignment β. ◁

As the transformation of d-CNF-SAT to Vertex Cover/d-COC can be performed in $n^{\mathcal{O}(1)}$ time, the lemma follows from the above two claims. ◀

Now we have the following theorem.

▶ **Theorem 2.** *For every $\epsilon > 0$ and every positive integer d,* Vertex Cover/d-COC *has no compression of vertex size $\mathcal{O}(k^{d-\epsilon})$ unless* co-NP $\subseteq$ NP/poly.

Proof. Our proof is divided into three cases in order to prove that, for any $\epsilon > 0$ and an integer $d \in \mathbb{N}$, there exists no polynomial time algorithm that can transform a given instance of Vertex Cover/d-COC to an equivalent instance of any arbitrary problem with $\mathcal{O}(k^{d-\epsilon})$ bits, unless co-NP $\subseteq$ NP/poly.

Case 1. $d = 1$. In the case where $d = 1$, the problem known as Vertex Cover/1-COC refers to the Vertex Cover problem parameterized by solution size. In this particular case, the result stated in Theorem 14 proves the theorem.

Case 2. $d = 2$. We can observe that the size of a 2-COC set is at most the size of a minimum vertex cover of the graph. As a result, 2-COC can be considered a parameter smaller than 1-COC (vertex cover). Therefore, if Vertex Cover/2-COC admits a compression of $\mathcal{O}(k^{2-\epsilon})$ bits, it would imply that the Vertex Cover problem parameterized by the solution size also has a compression of $\mathcal{O}(k^{2-\epsilon})$ bits. However, this contradicts the result stated in Theorem 14.

Case 3. $d \geq 3$. Suppose we have an instance (G, k, t) of Vertex Cover/d-COC, where $d \geq 3$, and there exists a polynomial time algorithm $\mathcal{A}$ that can transform (G, k, t) into an equivalent instance I of an arbitrary problem L such that I can be represented using $\mathcal{O}(k^{d-\epsilon})$ bits. To demonstrate the implications of this assumption, let us consider an instance Π of d-CNF-SAT with n variables and m clauses. First, we utilize the polynomial time algorithm described in Lemma 16 to transform Π into an instance $(G, 2n, n + \sum_{j=1}^{m} d_j - 1)$ of Vertex Cover/d-COC. Next, we apply the algorithm $\mathcal{A}$ to this transformed instance $(G, 2n, n + \sum_{j=1}^{m} d_j - 1)$, resulting in an equivalent instance I of problem L. Based on our assumption, we know that I can be represented using $\mathcal{O}(2n^{d-\epsilon}) = \mathcal{O}(n^{d-\epsilon})$ bits. However, this leads to a contradiction with co-NP $\subseteq$ NP/poly, as stated in Theorem 13.

This completes the proof. ◀

5 Conclusion

In this paper, we show that c-COC/d-COC admits a polynomial kernel with $\mathcal{O}(k^{d-c+1})$ vertices and $\mathcal{O}(k^{d-c+2})$ edges, where k is the size of the minimum d-coc set. Importantly, we observe that the degree of the polynomial in the kernel size is solely determined by the difference between d and c, and is independent of the specific values of d and c. Furthermore, we establish that obtaining a uniform kernel for the problem, where the exponent of k is independent of $d-c$, is unlikely under reasonable complexity assumptions. This result contributes valuable insights to the field of kernelization for Vertex Cover, particularly regarding c-Component Order Connectivity, when considering parameterizations smaller than the conventional solution size.

References

1 Marin Bougeret, Bart M. P. Jansen, and Ignasi Sau. Bridge-depth characterizes which minor-closed structural parameterizations of vertex cover admit a polynomial kernel. *SIAM J. Discret. Math.*, 36(4):2737–2773, 2022. doi:10.1137/21m1400766.

2 Marin Bougeret and Ignasi Sau. How much does a treedepth modulator help to obtain polynomial kernels beyond sparse graphs? *Algorithmica*, 81(10):4043–4068, 2019. doi:10.1007/s00453-018-0468-8.

3 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.

4 Holger Dell and Dieter van Melkebeek. Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses. *J. ACM*, 61(4):23:1–23:27, 2014. doi:10.1145/2629620.

5 Fedor V. Fomin, Daniel Lokshtanov, Neeldhara Misra, Geevarghese Philip, and Saket Saurabh. Hitting forbidden minors: Approximation and kernelization. *SIAM J. Discret. Math.*, 30(1):383–410, 2016. doi:10.1137/140997889.

6 Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. *Kernelization: Theory of Parameterized Preprocessing: Theory of parameterized preprocessing*. Cambridge University Press, United Kingdom, 2019. Publisher Copyright: © Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi 2019. doi:10.1017/9781107415157.

7 Fedor V. Fomin and Torstein J. F. Strømme. Vertex cover structural parameterization revisited. In Pinar Heggernes, editor, *Graph-Theoretic Concepts in Computer Science - 42nd International Workshop, WG 2016, Istanbul, Turkey, June 22-24, 2016, Revised Selected Papers*, volume 9941 of *Lecture Notes in Computer Science*, pages 171–182, 2016. doi:10.1007/978-3-662-53536-3_15.

8 Archontia C. Giannopoulou, Bart M. P. Jansen, Daniel Lokshtanov, and Saket Saurabh. Uniform kernelization complexity of hitting forbidden minors. *ACM Trans. Algorithms*, 13(3):35:1–35:35, 2017. doi:10.1145/3029051.

9 Eva-Maria C. Hols and Stefan Kratsch. Smaller parameters for vertex cover kernelization. In Daniel Lokshtanov and Naomi Nishimura, editors, *12th International Symposium on Parameterized and Exact Computation, IPEC 2017, September 6-8, 2017, Vienna, Austria*, volume 89 of *LIPIcs*, pages 20:1–20:12. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017. doi:10.4230/LIPIcs.IPEC.2017.20.

10 Bart M. P. Jansen and Hans L. Bodlaender. Vertex cover kernelization revisited - upper and lower bounds for a refined parameter. *Theory Comput. Syst.*, 53(2):263–299, 2013. doi:10.1007/s00224-012-9393-4.

11 Stefan Kratsch. A randomized polynomial kernelization for vertex cover with a smaller parameter. *SIAM J. Discret. Math.*, 32(3):1806–1839, 2018. doi:10.1137/16M1104585.

12 Stefan Kratsch and Magnus Wahlström. Representative sets and irrelevant vertices: New tools for kernelization. *J. ACM*, 67(3):16:1–16:50, 2020. doi:10.1145/3390887.

13 Mithilesh Kumar and Daniel Lokshtanov. A 2lk kernel for l-component order connectivity. In Jiong Guo and Danny Hermelin, editors, *11th International Symposium on Parameterized and Exact Computation, IPEC 2016, August 24-26, 2016, Aarhus, Denmark*, volume 63 of *LIPIcs*, pages 20:1–20:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPIcs.IPEC.2016.20.

14 Michael Lampis. A kernel of order 2 k-c log k for vertex cover. *Inf. Process. Lett.*, 111(23-24):1089–1091, 2011. doi:10.1016/j.ipl.2011.09.003.

15 Diptapriyo Majumdar, Venkatesh Raman, and Saket Saurabh. Polynomial kernels for vertex cover parameterized by small degree modulators. *Theory Comput. Syst.*, 62(8):1910–1951, 2018. doi:10.1007/s00224-018-9858-1.

16 Geevarghese Philip, Varun Rajan, Saket Saurabh, and Prafullkumar Tale. Subset feedback vertex set in chordal and split graphs. *Algorithmica*, 81(9):3586–3629, 2019. doi:10.1007/s00453-019-00590-9.

17 Stéphan Thomassé. A $4k^2$ kernel for feedback vertex set. *ACM Trans. Algorithms*, 6(2):32:1–32:8, 2010. doi:10.1145/1721837.1721848.

18 Mingyu Xiao. Linear kernels for separating a graph into components of bounded size. *J. Comput. Syst. Sci.*, 88:260–270, 2017. doi:10.1016/j.jcss.2017.04.004.