Improved Kernels for Signed Max Cut Parameterized Above Lower Bound on (r, ℓ)-graphs*

Luerbio Faria[1]   Sulamita Klein[2]   Ignasi Sau[3,4]   Rubens Sucupira[1]

[1] *Instituto de Matemática e Estatística - UERJ and COPPE/Sistemas - UFRJ, Rio de Janeiro, Brazil*
[2] *Instituto de Matemática and COPPE/Sistemas - UFRJ, Rio de Janeiro, Brazil*
[3] *CNRS, LIRMM, Université de Montpellier, Montpellier, France*
[4] *Departamento de Matemática, Universidade Federal do Ceará, Fortaleza, Brazil*

A graph G is *signed* if each edge is assigned "+" or "−". A signed graph is *balanced* if there is a bipartition of its vertex set such that an edge has sign "−" if and only if its endpoints are in different parts. The Edwards-Erdős bound states that every signed graph with n vertices and m edges has a balanced subgraph with at least $\frac{m}{2} + \frac{n-1}{4}$ edges. In the SIGNED MAX CUT ABOVE TIGHT LOWER BOUND (SIGNED MAX CUT ATLB) problem, given a signed graph G and a parameter k, the question is whether G has a balanced subgraph with at least $\frac{m}{2} + \frac{n-1}{4} + \frac{k}{4}$ edges. This problem generalizes MAX CUT ABOVE TIGHT LOWER BOUND, for which a kernel with $O(k^5)$ vertices was given by Crowston et al. [ICALP 2012, Algorithmica 2015]. Crowston et al. [TCS 2013] improved this result by providing a kernel with $O(k^3)$ vertices for the more general SIGNED MAX CUT ATLB problem. In this article we are interested in improving the size of the kernels for SIGNED MAX CUT ATLB on restricted graph classes for which the problem remains hard. For two integers $r, \ell \geq 0$, a graph G is an (r, ℓ)-*graph* if $V(G)$ can be partitioned into r independent sets and ℓ cliques. Building on the techniques of Crowston et al. [TCS 2013], for any $r, \ell \geq 0$ we provide a kernel with $O((r + \ell)k^2)$ vertices on (r, ℓ)-graphs, and a simple linear kernel on subclasses of split graphs for which we prove that the problem is still NP-hard.

Keywords: max cut, (r, ℓ)-graphs, split graphs, parameterized complexity, parameterization above lower bound, polynomial kernels

1   Introduction

A graph $G = (V, E)$ is a *signed graph* if each edge is assigned *positive* ("+") or *negative* ("−"). In this paper all graphs will be signed graphs. The labels "+" or "−" are the *signs* of the corresponding edges. If (V_1, V_2) is a bipartition of the vertex set of a graph G, we say that G is (V_1, V_2)-*balanced* if the edges with both endpoints in V_1 or both endpoints in V_2 are positive and the edges with endpoints in

distinct parts are negative. A graph G is *balanced* if there exists a partition (V_1, V_2) of $V(G)$ such that G is (V_1, V_2)-balanced. (Note that we can use an algorithm for 2-SAT to determine whether a signed graph is balanced.)

The problem of finding a balanced subgraph of a signed graph with maximum number of edges is called SIGNED MAX CUT and is NP-hard, as it is a generalization of MAX CUT, which is well known to be NP-hard [16] (indeed, MAX CUT is exactly SIGNED MAX CUT when all edges of G are negative). We refer to the work of Crowston et al. [5] and the references therein for some applications of the SIGNED MAX CUT problem.

In this article we study SIGNED MAX CUT from the perspective of Parameterized Complexity; we refer to the monographs [7, 9, 14, 22] for an introduction to the field. Namely, we consider a so-called *parameterization above lower bound* of the problem, and we are particularly interested in obtaining small polynomial kernels on restricted graph classes. Let us first discuss MAX CUT. Edwards [10, 11] showed that any connected graph with n vertices and m edges[(i)] has a cut of size $\frac{m}{2} + \frac{n-1}{4}$, and that this value is *tight* (that is, that there exist infinitely many graphs for which the size of a maximum cut equals this bound). This bound is commonly known as the *Edwards-Erdős* bound, and justifies the following parameterization of MAX CUT recently considered by Crowston et al. [6]: given a graph G and a parameter k, decide whether G has a cut of size at least $\frac{m}{2} + \frac{n-1}{4} + \frac{k}{4}$. They provided FPT-algorithms for this parameterization of the problem and a kernel of size $O(k^5)$ on general graphs.

Coming back to the SIGNED MAX CUT problem, Poljak and Turzík [23] proved that $\frac{m}{2} + \frac{n-t}{4}$ is also a tight lower bound on the number of edges in a balanced subgraph of any signed graph G with t connected components; we denote this lower bound by $\mathrm{pt}(G)$. Motivated by the above result for MAX CUT, Crowston et al. [5] considered the following parameterization of SIGNED MAX CUT, which is the same we consider in this article:

SIGNED MAX CUT ABOVE TIGHT LOWER BOUND
Input: A connected signed graph G and a positive integer k.
Parameter: k.
Question: Does G contain a balanced subgraph with at least $\frac{m}{2} + \frac{n-1}{4} + \frac{k}{4}$ edges?

We call the above problem SIGNED MAX CUT ATLB for short. Crowston et al. [5] proved that SIGNED MAX CUT ATLB is also FPT and provided a kernel of size $O(k^3)$ on general graphs, therefore improving the kernel of size $O(k^5)$ for the particular case of MAX CUT ATLB given by Crowston et al. [6].

Our results. In this article we are interested in improving the size of the cubic kernel for SIGNED MAX CUT ATLB by Crowston et al. [5] on particular graph classes for which the problem remains NP-hard. In particular, we focus on (r, ℓ)-graphs. For two integers $r, \ell \geq 0$, a graph G is an (r, ℓ)-*graph* if $V(G)$ can be partitioned into r independent sets and ℓ cliques. These graph classes contain, for example, split graphs or bipartite graphs, and have been extensively studied in the literature [1, 3, 4, 13, 15, 18, 21]. In particular, it is known that the recognition of (r, ℓ)-graphs is NP-complete if and only if $\max\{r, \ell\} \geq 3$ [3, 13].

Note that for this parameterization of SIGNED MAX CUT to make sense on (r, ℓ)-graphs, the bound $\frac{m}{2} + \frac{n-1}{4}$ should be also *tight* on these graph classes. Fortunately, this is indeed the case: let G be a clique with odd number of vertices, which is clearly an (r, ℓ)-graph as long as $\ell > 0$ (in particular, a split graph), and such that all edges are negative. It is clear that the largest balanced subgraph of G is given by the

---

[(i)] We assume that all input graphs of the problems under consideration have n vertices and m edges.

crossing edges of any bipartition (V_1, V_2) of $V(G)$ with $|V_1| = \frac{n+1}{2}$ and $|V_2| = \frac{n-1}{2}$. The number of edges of such a balanced subgraph, namely $\frac{n^2-1}{4}$, equals $\mathsf{pt}(G) = \frac{n(n-1)}{4} + \frac{n-1}{4}$.

From a parameterized complexity perspective, split graphs have received considerable attention. For instance, Raman and Saurabh [24] proved that DOMINATING SET is W[2]-hard on connected split graphs, and Ghosh et al. [17] provided FPT-algorithms for the problem of deleting at most k vertices to obtain a split graph. Concerning kernelization, Heggernes et al. [20] studied the DISJOINT PATHS problem on split graphs parameterized by the number of pairs of terminals, and provided a kernel of size $O(k^2)$ (resp. $O(k^3)$) for the vertex-disjoint (resp. edge-disjoint) version of the problem. As for general (r, ℓ)-graphs from a parameterized point of view, recently a dichotomy on the parameterized complexity for the problem of deleting at most k vertices to obtain an (r, ℓ)-graph has been independently obtained by Baste et al. [1] and by Kolay and Panolan [21]. To the best of our knowledge, this is the first article that focuses on obtaining polynomial kernels on (r, ℓ)-graphs for arbitrary values of r and ℓ.

Our main result (Theorem 3 in Section 3) is that SIGNED MAX CUT ATLB admits a quadratic kernel on (r, ℓ)-graphs for every fixed $r, \ell \geq 0$. More precisely, the kernel has $O((r + \ell)k^2)$ vertices. Our techniques in order to prove Theorem 3 are strongly based on the ones used by Crowston et al. [5], and for improving their cubic bound we further exploit the structure of (r, ℓ)-graphs in the analysis of the algorithm. In fact, our kernelization algorithm consists in applying exhaustively the reduction rules of Crowston et al. [5], only the analysis changes. As when $\max\{r, \ell\} \geq 3$ the recognition of (r, ℓ)-graphs is NP-complete [3, 13], we stress here that, given an instance (G, k), we do *not* need to obtain any partition of $V(G)$ into r independent sets and ℓ cliques; we only use the *existence* of such partition in the analysis. Since the analysis of the quadratic kernel for the particular case of split graphs is simpler, we prove it separately in Subsection 3.2.

In Section 4 we present a linear kernel for SIGNED MAX CUT ATLB on subclasses of split graphs. Namely, these subclasses are what we call d^*-*split graphs* for every integer $d \geq 1$ (see Section 4 for the definition). We first prove that even MAX CUT is NP-hard on d^*-split graphs for every integer $d \geq 2$, and in Theorem 5 we provide the linear kernel. As discussed later, this kernelization algorithm is the simplest possible algorithm that one could imagine, as it does *nothing* to the input graph; its interest lies on the analysis of the kernel size, which is non-trivial. In particular, the analysis uses a new reduction rule that we introduce in Section 2.

2   Preliminaries

We use standard graph-theoretic notation; see for instance Diestel's book [8]. All the graphs we consider are undirected and contain neither loops nor multiple edges. If $S \subseteq V(G)$, we define $G[S] = (S, \{\{u, v\} \in E(G) \mid u, v \in S\})$ and $G - S = G[V(G) \setminus S]$. A graph $G = (V, E)$ is a *split graph* if there is a partition of V into an independent set I and a clique K. Split graphs can be generalized as follows. Let r, ℓ be two positive integers. A graph $G = (V, E)$ is an (r, ℓ)-*graph* if V can be partitioned into at most r independent sets $I^1, I^2, I^3, \ldots, I^r$ and at most ℓ cliques $K^1, K^2, K^3, \ldots, K^\ell$.

Given a signed graph $G = (V, E)$, a cycle C in G is called *positive* if the number of negative edges in C is even. Otherwise C is called *negative*. Harary [19] proved that the absence of *negative* cycles characterizes balanced graphs.

Theorem 1 (Harary [19]) *A signed graph G is balanced if and only if every cycle in G is positive.*

The following lemma by Crowston et al. [5] is very useful for our purposes, as it gives a lower bound

on the maximum size of a balanced signed subgraph of a graph. Let $G = (V, E)$ be a signed graph and let $U, W \subseteq V$. We denote by $E(U, W)$ the subset of E formed by the edges that have one endpoint in U and the other in W. We also let $\beta(G)$ denote the maximum number of edges in a balanced subgraph of G.

Lemma 1 (Crowston et al. [5]) *Let $G = (V, E)$ be a connected signed graph and let $V = U \cup W$ such that $U \cap W = \emptyset$, $U \neq \emptyset$, and $W \neq \emptyset$. Then $\beta(G) \geq \beta(G[U]) + \beta(G[W]) + \frac{1}{2}|E(U, W)|$. In addition, if $G[U]$ has c_1 components, $G[W]$ has c_2 components, $\beta(G[U]) \geq \mathrm{pt}(G[U]) + \frac{k_1}{4}$, and $\beta(G[W]) \geq \mathrm{pt}(G[W]) + \frac{k_2}{4}$, then $\beta(G) \geq \mathrm{pt}(G) + \frac{k_1 + k_2 - (c_1 + c_2 - 1)}{4}$.*

We say that a problem is *fixed-parameter tractable* (FPT) [7, 9, 14, 22] with respect to parameter k if there exists an algorithm that solves the problem in time $f(k) \cdot n^{O(1)}$, where f is a computable function of k which is independent of n. *Kernelization* is an important technique used to shrink the size of a given problem instance by means of polynomial-time data reduction rules until the size of this instance is bounded by a function of the parameter k. The reduced instance is called a problem *kernel*. Once a problem kernel is obtained, we know that the problem is fixed-parameter tractable, since the running time of any brute force algorithm depends on the parameter k only. The converse is also true: whenever a parameterized problem is FPT, then it admits a kernel [7, 9, 14, 22]. The natural question to be asked about kernels is whether a parameterized problem admits a *polynomial* kernel or not, that is, a kernel of size $k^{O(1)}$.

In their article, Crowston et al. [5] proved that SIGNED MAX CUT ATLB is FPT on general graphs by designing an algorithm running in time $2^{3k} \cdot n^{O(1)}$. The algorithm applies some reduction rules to the input (G, k) that either answer that (G, k) is a YES-instance, or produce a set S of at most $3k$ vertices such that $G - S$ is a forest of cliques, or equivalently *clique-forest*, which is a graph such that every 2-connected component (that is, each block) is a clique without positive edges. (Note that this definition differs from the usage of this term in the context of chordal graphs.) We state this property formally as follows.

Proposition 1 (Crowston et al. [5]) *Let (G, k) be an instance of SIGNED MAX CUT ATLB. In polynomial time we can conclude that (G, k) is a YES-instance or we can find a set S of at most $3k$ vertices such that $G - S$ is a clique-forest without positive edges.*

There are two kinds of reduction rules applied by Crowston et al. [5] to an instance (G, k) in order to obtain the kernel of size $O(k^3)$ for SIGNED MAX CUT ATLB: one-way reduction rules and two-way reduction rules. In a *two-way reduction rule*, the instance (G', k') produced by the reduction rule is *equivalent* to (G, k) (that is, (G, k) is a YES-instance if and only if (G', k') is a YES-instance), so these rules can be safely applied to any instance in order to obtain FPT-algorithms or kernels, as long as the parameter k does not increase, which is always the case in the rules defined by Crowston et al. [5]. A two-way reduction rule is *valid* if it transforms YES-instances into YES-instances and NO-instances into NO-instances, that is, if it works as it should.

On the other hand, in a *one-way reduction rule*, the instance (G', k') produced by the reduction rule does not need to be equivalent to (G, k), but only needs to satisfy that if (G', k') is a YES-instance, then (G, k) is a YES-instance as well. The usefulness of such rules relies on the fact that if after the application of some two-way or one-way reduction rules we obtain an instance (G', k') with $k' \leq 0$, we can safely conclude that the original instance (G, k) is a YES-instance. This fact will be used in the linear kernel provided in Section 4. A one-way reduction rule is *safe* if it does not transform a NO-instance into a YES-instance.

3   A quadratic kernel on (r, ℓ)-graphs

In this section we show that SIGNED MAX CUT ATLB admits a quadratic kernel on (r, ℓ)-graphs for any fixed integers $r, \ell \geq 0$. For this we strongly use the results and the techniques provided by Crowston et al. [5]. The kernelization algorithm applies to the instance (G, k) the algorithm given by Proposition 1. This algorithm uses the two-way reduction rules Rule 8, Rule 9, Rule 10, and Rule 11 defined by Crowston et al. [5]. Namely, when these rules cannot be applied anymore, either one can directly conclude that (G, k) is a YES-instance or, using that the graph cannot be reduced anymore, one can prove that its size is bounded by a function of k only. It is at this point when we exploit the structure of (r, ℓ)-graphs in order to improve the cubic kernel on general graphs and obtain a quadratic kernel on (r, ℓ)-graphs. We would like to stress that we do *not* need to obtain algorithmically any particular partition of $V(G)$ into r independent sets and ℓ cliques, as such a partition will only be used for the *analysis*, and not by the kernelization algorithm.

This section is organized as follows. For completeness, in Subsection 3.1 we state all the reduction rules and the results from Crowston et al. [5] that we need for our purposes. For the sake of providing more intuition on the ideas behind our kernel and because it is simpler, we first present in Subsection 3.2 the quadratic kernel on split graphs (that is, for $r = \ell = 1$), and we describe in Subsection 3.3 its generalization to arbitrary values of r and ℓ. The main difference is that, in the case of split graphs, it can be seen that we do not have to worry about path vertices in $G - S$.

3.1   Reduction rules and known results

Before stating the two-way reduction rules used in the kernelization algorithm, we first need to introduce some definitions from [5]. Let S be a set of vertices as in Proposition 1. For a block C in $G - S$, let $C_{\text{int}} = \{x \in V(C) \mid N_{G-S}(x) \subseteq V(C)\}$ be the *interior* of C, and let $C_{\text{ext}} = V(C) \backslash C_{\text{int}}$ be the *exterior* of C. If a block C has only two vertices and these vertices belong to C_{ext}, then C is called a *path block*. A vertex that belongs only to path blocks is called a *path vertex*. A block C in $G - S$ is called a *leaf block* if $|C_{\text{ext}}| \leq 1$. We denote by N^+ (resp. N^-) the set of neighbors (of a vertex or of a vertex set) adjacent via a positive (resp. negative) edge.

In the following rules, the main idea is that, if some simple local conditions are satisfied, then certain vertices or edges can be deleted from the graph without changing the answer to the problem. Most of these conditions concern blocks or connected components of $G - S$ and their neighborhoods in S.

Rule 8. *Let C be a block in $G - S$. If there exists $X \subseteq C_{int}$ such that $|X| > \frac{|V(C)| + |N_G(X) \cap S|}{2} \geq 1$, $N_G^+(x) \cap S = N_G^+(X) \cap S$ and $N_G^-(x) \cap S = N_G^-(X) \cap S$ for all $x \in X$, then delete two arbitrary vertices $x_1, x_2 \in X$ and set $k' = k$.*

Rule 9. *Let C be a block in $G - S$. If $|V(C)|$ is even and there exists $X \subseteq C_{int}$ such that $|X| = \frac{|V(C)|}{2}$ and $N_G(X) \cap S = \emptyset$, then delete a vertex $x \in X$, and set $k' = k - 1$.*

Rule 10. *Let C be a block in $G - S$ with vertex set $\{x, y, u\}$, such that $N_G(u) = \{x, y\}$. If the edge $\{x, y\}$ is a bridge in $G - u$, delete C, add a new vertex z, positive edges $\{\{z, v\} \mid v \in N_{G-u}^+(\{x, y\})\}$, negative edges $\{\{z, v\} \mid v \in N_{G-u}^-(\{x, y\})\}$, and set $k' = k$. Otherwise, delete u and the edge $\{x, y\}$ and set $k' = k - 1$.*

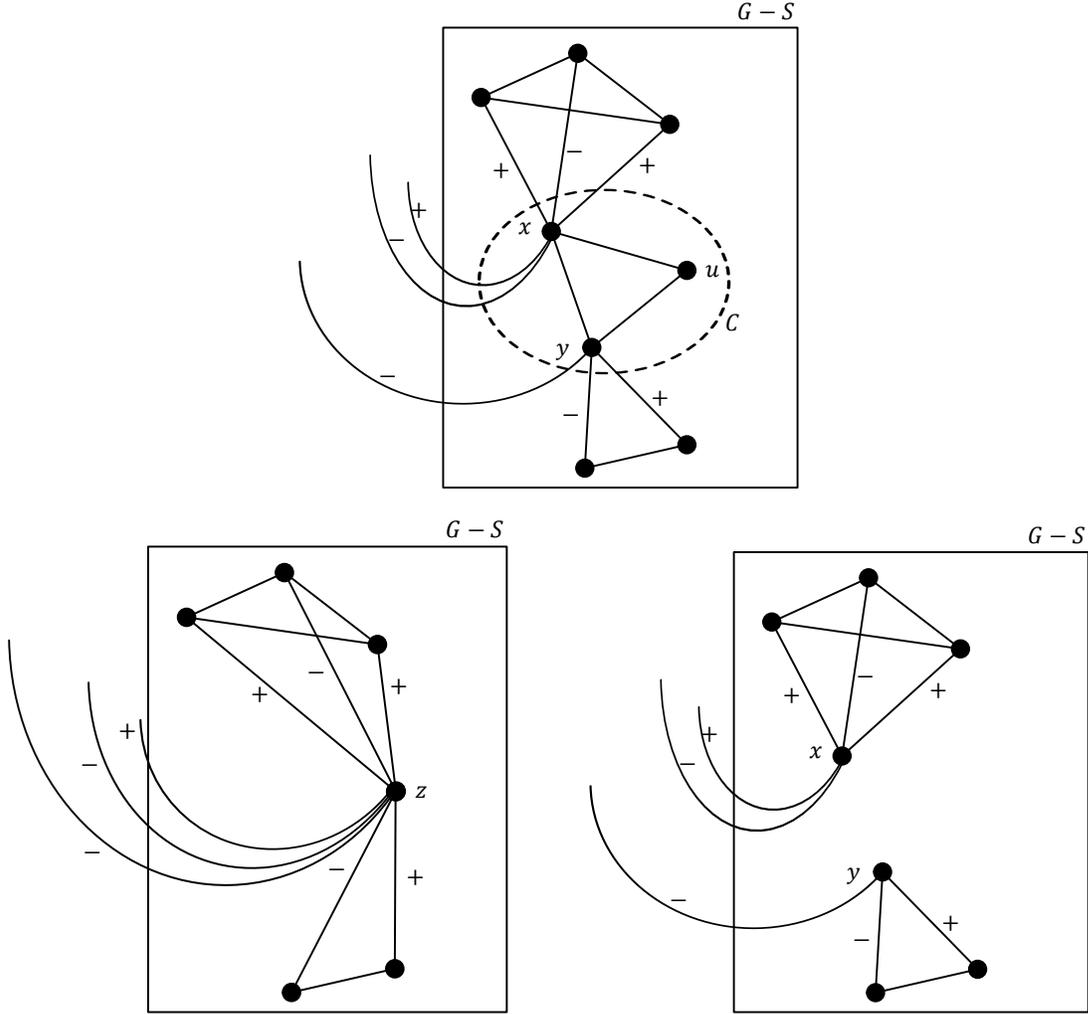An illustration of the application of Rule 10 can be found in Fig. 1.

Fig. 1: Illustration of the application of Rule 10: a block C satisfying the conditions (up), and the two possible resulting graphs after applying the rule (down).

The MAX CUT WITH WEIGHTED VERTICES problem is defined as follows. We are given a graph G with weight functions $w_1 : V(G) \to \mathbb{N}_0$ and $w_2 : V(G) \to \mathbb{N}_0$, and an integer $t \in \mathbb{N}$, and the question is whether there exists an assignment $f : V(G) \to \{1, 2\}$ such that $\sum_{\{x,y\} \in E} |f(x) - f(y)| + \sum_{f(x)=1} w_1(x) + \sum_{f(x)=2} w_1 2(x) \geq t$.

Rule 11. *Let T be a connected component of $G - S$ only adjacent to a vertex $s \in S$. Form a* MAX CUT WITH WEIGHTED VERTICES *instance on T by defining $w_1(x) = 1$ if $x \in N_G^+(s) \cap T$ ($w_1(x) = 0$ otherwise) and $w_2(y) = 1$ if $y \in N_G^-(s) \cap T$ ($w_2(y) = 0$ otherwise). Let $\beta(G[V(T) \cup \{s\}]) =$*

$\mathsf{pt}(G[V(T) \cup \{s\}]) + \frac{p}{4}$. *Then delete T and set $k' = k - p$.*

By [6, Lemma 9], the value of p in Rule 11 can be found in polynomial time.

Crowston et al. [5] proved that the two-way reduction Rules 8-11 are all valid. As mentioned in [5], since it can be easily verified that none of these reduction rules increases the number of positive edges, our proof also implies a kernel of size $O(k^2)$ for SIGNED MAX CUT ATLB on (r, ℓ)-graphs.

We now state the results from Crowston et al. [5] that we will use in the proofs of Theorem 2 and Theorem 3.

Lemma 2 (Lemma 9 in Crowston et al. [5]) *Let T be a connected component of $G - S$. Then for every leaf block C of T, $N_G(C_{int}) \cap S \neq \emptyset$. Furthermore, if $|N_G(S) \cap V(T)| = 1$, then T consists of a single vertex.*

Lemma 3 (Corollary 5 in Crowston et al. [5]) *If $\sum_{C \in \mathcal{B}} |N_G(C_{int}) \cap S| \geq |S|(2|S| - 3 + 2k) + 1$, the instance is a YES-instance. Otherwise, $\sum_{C \in \mathcal{B}} |N_G(C_{int}) \cap S| \leq 3k(8k - 3)$.*

Lemma 4 (Corollary 6 in Crowston et al. [5]) *$G - S$ contains at most $6k(8k - 3)$ non-path blocks and $24k^2(8k - 3)$ path vertices.*

Lemma 5 (Corollary 7 in Crowston et al. [5]) *$G - S$ contains at most $12k(8k - 3)$ vertices in the exteriors of non-path blocks.*

Lemma 6 (Lemma 14 in Crowston et al. [5]) *For a block C, if $|V(C)| \geq 2|C_{ext}| + |N_G(C_{int}) \cap S|(2|S| + 2k + 1)$, then (G, k) is a YES-instance. Otherwise, $|V(C)| \leq 2|C_{ext}| + |N_G(C_{int}) \cap S|(8k + 1)$.*

3.2 *A quadratic kernel on split graphs*

In this subsection we present a quadratic kernel for SIGNED MAX CUT ATLB on split graphs, which contains the main ideas of the kernel described in Theorem 3 for arbitrary (r, ℓ)-graphs. The main simplification is that in the case of split graphs, we do not have to worry about path vertices in $G - S$.

The proof is based on exploiting the structure given by the set S of Proposition 1. We will strongly use the fact that, as the input graph is a split graph and this property is hereditary, both $G[S]$ and $G - S$ are split graphs as well.

Theorem 2 SIGNED MAX CUT ATLB *on split graphs admits a kernel with $O(k^2)$ vertices.*

Proof: Let (G, k) be the given instance, where $G = (V, E)$ is a split graph. Applying the polynomial-time algorithm of Proposition 1, we can either conclude that (G, k) is a YES-instance, or we obtain a set $S \subseteq V(G)$ such that $G - S$ is a clique-forest without positive edges. Let $S = K^1 \cup I^1$ and $G - S = K^2 \cup I^2$ be such that $K^1 \cup K^2$ is a clique and $I^1 \cup I^2$ is an independent set. We partition I^2 into three subsets: I_0^2 containing only isolated vertices in $G - S$, I_1^2 with all vertices of degree one in $G - S$, and I_u^2 containing all vertices $u \notin I_0^2 \cup I_1^2$ such that $N_{K^2}(u) = K^2$; see Fig. 2 for an illustration. Note that I_u^2 can contain at most one vertex. Indeed, if $|K^2| \leq 1$, then the set I_u^2 is empty by definition. Otherwise, if $|K^2| \geq 2$, if I_u^2 contains two distinct vertices u_1 and u_2, then $G - S$ is not a clique-forest, contradicting Proposition 1.

These sets indeed define a partition of I^2, as if there exists a vertex v in I^2 such that $1 < d_{K^2}(v) < |V(K^2)|$, we could find a block in $G - S$ that is not a clique. On the other hand, due to the structure of the clique-forest, two maximal cliques can intersect only in one vertex.
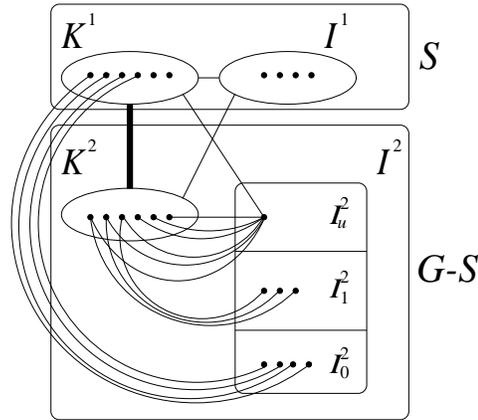
Fig. 2: Illustration of the proof of Theorem 2.

Observe that $G - S$ has at most one path block. Indeed, it is easily seen that the only possible case in which a path block exists in $G - S$ is when $K^2 = K_2$ (the complete graph with two vertices), $I_u^2 = \emptyset$, and each vertex of K_2 has at least one adjacent vertex in I_1^2. Hence, $G - S$ has no path vertices since in $G - S$ there is only one possible path block (namely, K_2) adjacent to at least two leaf blocks; see Fig. 3 for an illustration.
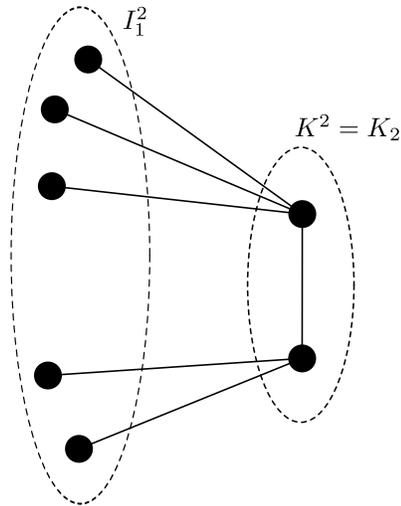

Fig. 3: Illustration of the unique possible path block in $G - S$.

For each vertex x in the set I_1^2 there is a leaf block C^x with $|V(C^x)| = 2$ and $C_{\text{int}}^x = \{x\}$. So, Rule 9 could be successively applied to each vertex in I_1^2 with no neighbor in S, deleting all these vertices and reducing the parameter k accordingly. We could also delete all the vertices of I_0^2 with no neighbor in S, because they are isolated vertices and make no contribution to the cut. So in this case we set $k' = k$.

We note that Rule 10 could be applied only in two cases. In the first case $K^1 = \emptyset$ and K^2 is a triangle, and in the second case, $K^1 = \emptyset$, $K^2 = \{x, y\}$, and $I_u^2 = \{u\}$.

Using Lemma 2, it can be easily seen that after applying Rule 9, the remaining vertices of I_1^2 are adjacent to some vertex in S. Indeed, otherwise Rule 9 could be applied again, deleting the vertices $x \in C_{\text{int}}^x$ of I_1^2 with $N_G(C_{\text{int}}^x) \cap S = \emptyset$, contradicting the hypothesis that the graph G is reduced under all two-way reduction rules; see Fig. 4 for an illustration.
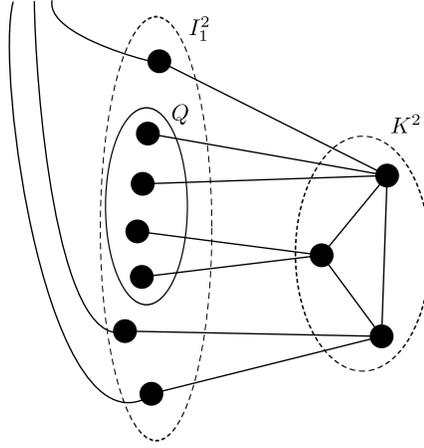

Fig. 4: The set $Q \subseteq I_1^2$, if it were non-empty, would be deleted by Rule 9.

Let $\mathcal{B}$ be the set of non-path blocks of $G - S$. As discussed before, in a split graph G all blocks of $G - S$, but possibly one, are non-path blocks. Since if there is some path block, we have that $|K^2| = |K_2| = 2$, we may assume in what follows that all blocks of $G - S$ are in $\mathcal{B}$.

By Lemma 3, either we can identify that G is a Yes-instance or it follows that $\sum_{C \in \mathcal{B}} |N_G(C_{\text{int}}) \cap S| \le 3k(8k - 3)$. Using Lemma 6 it follows that for all blocks C in $G - S$, we have

$$|V(C)| \le 2|C_{\text{ext}}| + |N_G(C_{\text{int}}) \cap S| \cdot (8k + 1).$$

Then $|V(G)| \le |S| + \sum_{C \in \mathcal{B}} |V(C)|$ since we may assume that are no path blocks in $G - S$. As G is a split graph, $G - S$ is also a split graph and our previous analysis shows that $\mathcal{B}$ is composed by one big block C_0 defined by K^2, leaf blocks that are edges which have at least one neighbor in S, and possibly isolated vertices. Let $\mathcal{B}_*$ be the set of leaf blocks in $\mathcal{B}$. We can rewrite the previous inequality as

$$|V(G)| \le |S| + |V(C_0)| + \sum_{C \in \mathcal{B}_*} |V(C)|.$$

In particular, for the big block C_0 we have $|V(C_0)| \le 2|C_{0\text{ext}}| + |N_G(C_{0\text{int}}) \cap S|(8k+1) \le 2|C_{0\text{ext}}| + |S|(8k+1)$. And for all leaf blocks $C \in \mathcal{B}_*$ we have $|V(C)| \le 2$. Then

$$\sum_{C \in \mathcal{B}_*} |V(C)| \le 2|\mathcal{B}_*| \le 2 \sum_{C \in \mathcal{B}_*} |N_G(C_{\text{int}}) \cap S| \le 2 \cdot 3k(8k-3)$$

as a consequence of Lemma 3, since every block $C \in \mathcal{B}_*$ has a vertex $v \in C_{\text{int}}$ such that $N_S(v) \ne \emptyset$. Finally, the number of vertices in G can be bounded as follows:

$$
\begin{aligned}
|V(G)| &\le& |S| + |S|(8k+1) + 6k(8k-3) + 2|C_{0\text{ext}}| \\
&\le& 3k + 3k(8k+1) + 6k(8k-3) + 2|\bigcup_{C \in \mathcal{B}} C_{\text{ext}}| \\
&\le& 3k + 3k(8k+1) + 6k(8k-3) + 12k(8k-3) \\
&=& 168k^2 - 48k = O(k^2),
\end{aligned}
$$

where the last inequality is due to Lemma 5.   □

3.3 Generalization to arbitrary (r, ℓ)-graphs

In this subsection we show how the ideas of the previous subsection can be generalized to arbitrary (r, ℓ)-graphs. We follow the same strategy as in the proof of Theorem 2, but we need a number of extra arguments, mostly to deal with the path vertices in $G - S$.

Namely, we will again exploit the structure given by the set S of Proposition 1, and the fact that, as the input graph is an (r, ℓ)-graph and this property is hereditary, both $G[S]$ and $G - S$ are (r, ℓ)-graphs as well. In order to bound the size of the set of path vertices in $G - S$, which we call $\mathcal{P}$ in the proof, we will need three technical claims.

Theorem 3 *For every two integers $r, \ell \ge 0$,* SIGNED MAX CUT ATLB *on (r, ℓ)-graphs admits a kernel with $O((r + \ell)k^2)$ vertices.*

Proof: Let G be an (r, ℓ)-graph for two integers $r, \ell \ge 0$. Denote by $K^1, K^2, \dots, K^\ell$ the cliques and $I^1, I^2, \dots, I^r$ the independent sets of G. We stress again that we just use the partition of G for the analysis, but we do not need to know exactly this partition. By Proposition 1, we can find a set S with at most $3k$ vertices such that the subgraph $G - S$ is a forest of cliques without positive edges. For an (r, ℓ)-graph, we have $S = K_1^1 \cup \cdots \cup K_1^\ell \cup I_1^1 \cup \cdots \cup I_1^r$ and $G - S = K_2^1 \cup \cdots \cup K_2^\ell \cup I_2^1 \cup \cdots \cup I_2^r$ with $K^i = K_1^i \cup K_2^i$ and $I^j = I_1^j \cup I_2^j$ for $1 \le i \le \ell$ and $1 \le j \le r$. In contrast to split graphs (see Subsection 3.2), we may have path vertices in $G - S$.

In order to bound the size of $V(G)$, we bound separately the total size of non-path blocks and the number of path vertices (that is, the only vertices that do not belong to non-path blocks) in $G - S$. Denote again by $\mathcal{B}$ the set of non-path blocks, by $\mathcal{B}_b$ the set of non-path blocks containing at least two vertices from the set $K^1 \cup K^2 \cup \cdots \cup K^\ell$, and $\mathcal{B}_s = \mathcal{B} \backslash \mathcal{B}_b$. The structure of clique-forests implies that $|\mathcal{B}_b| \le \ell$, and by Lemma 6, for every block $C \in \mathcal{B}_b$ either we can conclude that (G, k) is a YES-instance or we have that

$$|V(C)| \le 2|C_{\text{ext}}| + |N_G(C_{\text{int}}) \cap S| \cdot (8k+1) \le 2|C_{\text{ext}}| + |S|(8k+1).$$

Then we can write

$$
\begin{aligned}
\sum_{C \in \mathcal{B}_b} |V(C)| &\leq 2 \sum_{C \in \mathcal{B}_b} |C_{\text{ext}}| + \sum_{C \in \mathcal{B}_b} |S|(8k+1) \\
&\leq 2\ell \cdot |\bigcup_{C \in \mathcal{B}_\ell} C_{\text{ext}}| + \ell |S|(8k+1) \\
&\leq 24\ell k(8k-3) + 3\ell k(8k+1),
\end{aligned}
$$

where in the second inequality we have used twice that $|\mathcal{B}_b| \leq \ell$, and the last inequality follows from Lemma 5.

For each block $C \in \mathcal{B}_s$ we have $|V(C)| \leq r + 1$, as C contains only vertices of the r independent sets $I^1, I^2, \ldots, I^r$ plus possibly one vertex of some of the cliques. Indeed, otherwise if $|V(C)| > r + 1$ then there are two adjacent vertices in the same independent set, which is impossible. Therefore, by Lemma 4, we can bound the sum of the number of vertices in $C \in \mathcal{B}_s$ as $\sum_{C \in \mathcal{B}_s} |V(C)| \leq (r + 1) \cdot 6k(8k-3)$.

Therefore, by combining the bounds from the two above paragraphs we conclude that

$$
\sum_{C \in \mathcal{B}} |V(C)| = \sum_{C \in \mathcal{B}_b} |V(C)| + \sum_{C \in \mathcal{B}_s} |V(C)| \leq (2r + 9\ell)3k(8k+1) =: g(k, r, \ell). \tag{1}
$$

It just remains to bound the number of path vertices in the blocks of $G - S$. Note that we cannot use directly the bound on the number of path vertices in $G - S$ given by Lemma 4, as this bound is $O(k^3)$, which is too much for our purposes.

Let $\mathcal{P}$ be the set of path vertices of $G - S$. In order to bound the size of $\mathcal{P}$, we argue separately about the components in $G[\mathcal{P}]$ having at most two vertices (Claim 1) and the remaining components (Claim 2), where $G[\mathcal{P}]$ denotes the subgraph of G induced by $\mathcal{P}$.

We call an *isolated vertex* (resp. *isolated edge*) a connected component of $G[\mathcal{P}]$ of size 1 (resp. 2). We show in Claim 1 that the number of isolated vertices and isolated edges in $G[\mathcal{P}]$ are both bounded by the number of non-path blocks, which is at most $6k(8k-3)$ by Lemma 4.

Claim 1 *The number of isolated vertices in $G[\mathcal{P}]$ and the number of isolated edges in $G[\mathcal{P}]$ are upper-bounded by the number of non-path blocks.*

Proof: Let us first deal with the number of isolated vertices in $G[\mathcal{P}]$, which we denote by p_0. We show by induction on p_0 that this number is at most the number of non-path blocks minus one. Recall that a vertex is isolated if and only if it is exclusively adjacent to non-path vertices. We contract each non-path block into a black vertex and represent each path vertex by a white vertex. So the forest of cliques in $G - S$ is now represented by a simple forest colored by two colors: black and white. If $p_0 = 1$, then there are at least two black vertices adjacent to the single white vertex of $G[\mathcal{P}]$. So the property is valid and the basis of the induction is proved. Suppose that the property is valid for $p_0 = i > 1$, and consider $G - S$ such that the number of isolated vertices is $i + 1$. Choose arbitrarily one of these isolated vertices. Without loss of generality, we can suppose that $G - S$ is connected. Otherwise we can deal with the connected components of $G - S$ separately to achieve the same result. The removal of this white vertex disconnects the tree, increasing the number of connected components. For all connected components the property holds by induction hypothesis, that is, the number of isolated vertices in each connected component is at most the number of non-path blocks in this component minus one. So, the number of isolated vertices in

the original graph is at most the total number of non-path blocks minus one, and the first part of the claim follows.

As for the number of isolated edges in $G[\mathcal{P}]$, similarly to above, we contract each non-path block into a black vertex and represent each path vertex as a white vertex. An isolated edge is represented by an edge that has white endpoints. We identify all these edges and contract each of them into a red vertex. As these edges are isolated, the red vertices have only black neighbors. Then, proceeding by induction as we did for the isolated vertices, we can show that the number of isolated edges is upper-bounded by the number of non-path blocks.   □

Let $\mathcal{P}_1$ be the subset of $\mathcal{P}$ formed by the non-isolated vertices. Note that the minimum degree of the vertices in $G[\mathcal{P}_1]$, which we denote by $\delta(G[\mathcal{P}_1])$, is such that $\delta(G[\mathcal{P}_1]) \geq 1$, so clearly the number of edges of $G[\mathcal{P}_1]$ is at least $\frac{|\mathcal{P}_1|}{2}$; this value is attained when $G[\mathcal{P}_1]$ consists of disjoint edges. We need to improve this bound of $\frac{|\mathcal{P}_1|}{2}$ for being able to state Equation (2) below, and we can easily do it by forbidding this latter case.

Claim 2 *If each connected component of $G[\mathcal{P}_1]$ has at least three vertices, then $|E(G[\mathcal{P}_1])| \geq \frac{2|\mathcal{P}_1|}{3}$.*

Proof: Without loss of generality, we can suppose that $G[\mathcal{P}_1]$ is connected by the same reason mentioned in the proof of Claim 1. Let $p = |\mathcal{P}_1|$. As $G[\mathcal{P}_1]$ is a tree, we have $|E(G[\mathcal{P}_1])| = p - 1$. If $p \geq 3$ then $\frac{p}{3} \geq 1$, that is, $p - \frac{2p}{3} \geq 1$ and $p - 1 \geq \frac{2p}{3}$ as desired.   □

The induced subgraph $G[\mathcal{P}_1]$ is a forest, so by Theorem 1 $G[\mathcal{P}_1]$ is balanced. Let e_p denote the number of edges in $G[\mathcal{P}_1]$. Then $\beta(G[\mathcal{P}_1]) = e_p$ and $\mathsf{pt}(G[\mathcal{P}_1]) = \frac{e_p}{2} + \frac{p-1}{4}$ if we suppose $G[\mathcal{P}_1]$ to be connected, where $p = |\mathcal{P}_1|$. Claim 2 implies that

$$\beta(G[\mathcal{P}_1]) - \mathsf{pt}(G[\mathcal{P}_1]) = e_p - \frac{e_p}{2} - \frac{p-1}{4} = \frac{2e_p - p + 1}{4} \geq \frac{\frac{4p}{3} - p + 1}{4} \geq \frac{\frac{p}{3}}{4}. \tag{2}$$

Claim 3 *Let (G, k) be an instance of* SIGNED MAX CUT ATLB. *If $\mathcal{P}$ is the set of path vertices in $G - S$, then the number of connected components of $G[\mathcal{P}]$ is at most $6k(8k - 3)$. If $\mathcal{W} = G - \mathcal{P}$, then the number of connected components of $\mathcal{W}$ is at most $6k(8k - 2)$.*

Proof: The same ideas used in the proof of Claim 1 easily imply that the number of connected components of $G[\mathcal{P}]$ is at most $6k(8k - 3)$. Since $\mathcal{W} \cap (G - S)$ is the complement of $G[\mathcal{P}]$ in $G - S$, each vertex in $\mathcal{W} \cap (G - S)$ belongs to a non-path block. And as the size of S is at most $3k$, by Lemma 4 the number of connected components of $\mathcal{W}$ is at most $3k + 6k(8k - 3) \leq 6k(8k - 2)$.   □

We are now ready to piece everything together and conclude the proof of the theorem.

Due to Claim 1, by incurring an additive term of $18k(8k-3)$ to the size of the kernel we may assume that each component of $G[\mathcal{P}]$ has at least three vertices, so the hypothesis of Claim 2 holds. We apply Lemma 1 to the graph G with $U = \mathcal{P}$. We have c_1 connected components of $G[\mathcal{P}]$, c_2 connected components of W, and $\beta(G) \geq \mathsf{pt}(G) + \frac{k_1 + k_2 - (c_1 + c_2 - 1)}{4}$, where $\beta(G[\mathcal{P}]) \geq \mathsf{pt}(G[\mathcal{P}]) + \frac{k_1}{4}$ and $\beta(W) \geq \mathsf{pt}(W) + \frac{k_2}{4}$. It holds of course that $k_2 \geq 0$. By Equation (2) we have $k_1 \geq \frac{p}{3}$ for $p = |\mathcal{P}_1|$ and, by Claim 3, it holds that $c_1 + c_2 \leq 12k(8k - 2) \leq 96k^2$. Then $\beta(G) \geq \mathsf{pt}(G) + \frac{\frac{p}{3} - 192k^2 + 1}{4}$. Therefore, if $\frac{p}{3} - 192k^2 + 1 \geq k$, then (G, k) is a YES-instance. Otherwise, $\frac{p}{3} \leq k - 1 + 192k^2$, that is, we may assume that $p \leq 576k^2 + 3k - 3$.

Therefore, combining Equation (1) with the above discussion we can conclude that

$$|V(G)| \leq |S| + \sum_{C \in \mathcal{B}} |V(C)| + |\mathcal{P}| \leq 3k + g(k, r, \ell) + 18k(8k - 3) + p = O(k^2).$$

Finally, note that in the whole proof, when we use some result of Crowston et al. [5] in which one of the possible outputs is that (G, k) is a YES-instance (like Lemma 14 or to check whether p is at most $576k^2 + 3k - 3$ or not), the condition to be checked concerns just the blocks of $G - S$, which we can obtain in polynomial time by Proposition 1. Thus, as we claimed, we do *not* need to compute any partition of $V(G)$ into cliques $K^1, K^2, \ldots, K^\ell$ and independent sets $I^1, I^2, \ldots, I^r$. This concludes the proof of the theorem. □

4   A linear kernel on subclasses of split graphs

With the objective of improving the quadratic kernel on (r, ℓ)-graphs presented in Section 3, in Subsection 4.2 we provide a linear kernel on a smaller graph class, namely on the subclass of d^*-split graphs for every integer $d \geq 1$, which is defined as follows. A graph $G = (V, E)$ is a d^*-*split graph* if V can be partitioned into a clique K and an independent set I such that every vertex in K has at least one neighbor in I, and every vertex in I has degree at most d.

We first prove in Subsection 4.1 that the problem remains NP-hard restricted to this class of graphs for every $d \geq 2$. In fact, we prove that even MAX CUT is NP-hard, and this result is tight in terms of d. Indeed, for $d = 0$, a 0^*-split graph is an independent set and therefore MAX CUT is trivial. For $d = 1$, the claim is given in the following simple observation.

Remark 1 MAX CUT *on* 1^*-*split graphs can be solved in polynomial time.*

Proof: Note that by the definition of 1^*-split graphs, such a graph has a very precise structure. Namely, it consists of an arbitrarily large clique K such that every vertex v in K has a private neighbor v' in I (that is, a vertex of degree one). Then, an optimal solution of MAX CUT consists of a balanced partition of the clique, and for each $v \in K$, we place its private neighbor v' in the opposite part of v. □

4.1 *Max Cut is NP-hard on* d^*-*split graphs*

In this subsection we prove that MAX CUT, which is a particular case of SIGNED MAX CUT, is NP-hard on d^*-split graphs for every fixed $d \geq 2$. We first prove easily that MAX CUT is NP-hard on graphs without universal vertices (a *universal* vertex is a vertex adjacent to every other vertex in the graph). Then, using a proof from Bodlaender and Jansen [2] on the complexity of MAX CUT, we prove that this problem is still NP-hard on 2^*-split graphs and, consequently, NP-hard on d^*-split graphs for every fixed $d \geq 2$.

Lemma 7 MAX CUT *is* NP-*hard on graphs without universal vertices.*

Proof: We use the well known fact that MAX CUT is NP-hard on general graphs [16] to show that this problem is still NP-hard on graphs without universal vertices. For this, we consider a graph G as instance of MAX CUT, and construct a new graph $G' = 2G$ made of two disjoint copies of G. Obviously G' has no universal vertices because it is disconnected. Let MAXCUT(G) be the size of a maximum cut for G and

MAXCUT(G') be the size of a maximum cut for G'. We claim that MAXCUT$(G') = 2 \cdot$ MAXCUT(G). Indeed, MAXCUT$(G') \geq 2 \cdot$ MAXCUT(G) because if $[V_1, V_2]$ is a cut in G of maximum size, then $2 \cdot$ MAXCUT(G) is the size of the cut $[2V_1, 2V_2]$; see Fig. 5 for an illustration. On the other hand, if MAXCUT$(G') > 2 \cdot$ MAXCUT(G), then as G' is made of two disjoint copies of G, necessarily G contains a cut of size greater than MAXCUT(G), a contradiction. $\square$
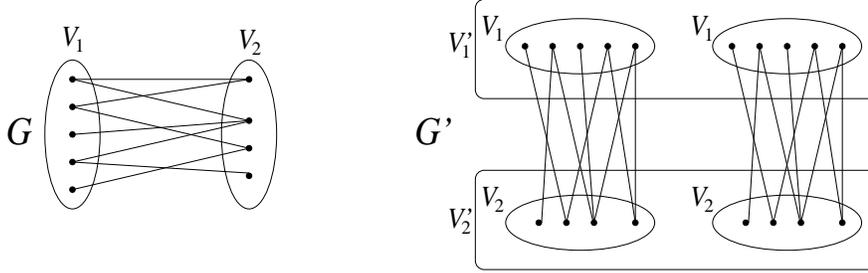

Fig. 5: Reduction to show that MAX CUT is NP-hard on graphs without universal vertices.

The construction given in the proof of the next theorem is exactly the same as the one given by Bodlaender and Jansen in [2, Theorem 3.1], but we reproduce it here because we prove that if we start with a graph without universal vertices (we can do so thanks to Lemma 7), then the graph constructed in the reduction is a 2^*-split graph.

Theorem 4 MAX CUT *is* NP-*hard on* 2^*-*split graphs.*

Proof: Let a graph $G = (V, E)$ be given and let $\overline{G} = (V, \overline{E})$ be the complement of G. Let $H = (V \cup \overline{E}, F)$, where $F = \{(v, w) \mid v, w \in V, v \neq w\} \cup \{(v, e) \mid v \in V, e \in \overline{E}, v$ is an endpoint of edge $e\}$. Then V forms a clique, $\overline{E}$ forms an independent set in H, and every edge-representing vertex e is connected to the vertices that represent its endpoints. Therefore H is a split graph in which all the vertices in the independent set have degree exactly two. We claim that G allows a partition with at least p cut edges if and only if H allows a partition with at least $2|\overline{E}| + p$ cut edges.

Suppose first we have a partition (W_1, W_2) of G with at least p cut edges. We partition the vertices of H as follows: partition V as in the partition of G; for every $e \in \overline{E}$, if both endpoints of e belong to W_1, then put e in W_2, otherwise put e in W_1. It is easy to see that this partition gives the desired number of cut edges.

Now suppose we have a partition (W_1, W_2) of H with at least $2|\overline{E}| + p$ cut edges. Partition the vertices of G into two subsets: $W_1 \cap V$ and $W_2 \cap V$. This partition gives the desired number of cut edges. This can be noted as follows: for every edge $\{v, w\} \in E$, we have one cut edge in H if $\{v, w\}$ is a cut edge in G, otherwise we have no cut edge. For every edge $e = \{v, w\} \in \overline{E}$, we have that out of the three edges $\{v, w\}, \{e, v\}, \{e, w\}$, exactly two will be cut edges. Hence, the total number of cut edges in H equals the number of cut edges in G plus $2|\overline{E}|$. Note that H can clearly be constructed from G in polynomial time.

By Lemma 7, we can assume that the graph $G = (V, E)$ has no universal vertices. Then the graph $\overline{G}$ constructed above has no isolated vertices, and therefore every vertex $v \in V$ has at least one neighbor in $\overline{E}$. Since every vertex in $\overline{E}$ has degree two, the graph H constructed in the above reduction is indeed a 2^*-split graph, and the theorem follows.   □

Since for every $d \geq 0$, the class of d^*-split graphs contains the class of 2^*-split graphs, the following corollary is a direct consequence of Theorem 4.

Corollary 1 MAX CUT *is* NP-*hard on* d^*-*split graphs for every fixed* $d \geq 2$.

4.2   The kernelization algorithm

As we will see in the proof of Theorem 5 below, our kernelization algorithm is the simplest possible algorithm that one could imagine, as it does *nothing* to the input graph; its interest lies on the analysis of the kernel size, which uses the following two new reduction rules.

The first one is a one-way reduction rule that is a generalization of the one-way Rule 6 given by Crowston et al. [5].

Rule 6$^+$. *Let G be a connected graph. If $v \in V(G)$ and $u_1, \ldots, u_c$ are pairwise non-adjacent neighbors of v such that $c \geq 2$ and $G - \{v, u_1, \ldots, u_c\}$ is connected, then delete $v, u_1, \ldots, u_c$ and set $k' = k - c + 1$.*

Note that Rule 6 of Crowston et al. [5] corresponds exactly to the case $c = 2$ of Rule 6$^+$ above.

Lemma 8 *Rule* 6$^+$ *is safe.*

Proof: Let $v \in V(G)$ and $u_1, \ldots, u_c$ be as in the description of Rule 6$^+$ and let $P = G[\{v, u_1, \ldots, u_c\}]$. Note that $\mathsf{pt}(P) = \frac{c}{2} + \frac{c}{4} = \frac{3c}{4}$ and since P is a tree (namely, a star), P is a balanced graph by Theorem 1 whatever the signs of its edges. Therefore, $\beta(P) = c = \mathsf{pt}(P) + \frac{c}{4}$. Let G' be the graph obtained from G by the deletion of $v, u_1, \ldots, u_c$, so G' is connected by hypothesis. Suppose that $\beta(G') \geq \mathsf{pt}(G') + \frac{k'}{4}$, where $k' = k - c + 1$. Then, by Lemma 1, $\beta(G) \geq \mathsf{pt}(G) + \frac{k' + c - 1}{4} = \mathsf{pt}(G) + \frac{k}{4}$, and therefore G is a YES-instance.   □

The second new rule is a simple two-way reduction rule, which just eliminates vertices of degree 1.

Rule A. *Let G be a connected graph, and let v be a vertex of degree 1 in G. Then delete vertex v and set $k' = k - 1$.*

Lemma 9 *Rule A is valid.*

Proof: Let G be a connected graph, let v be a vertex of degree 1 in G, and let G' be the graph obtained from G by applying Rule A on vertex v. Since the edge containing v in belongs to every optimal balanced subgraph of G (regardless of its sign), it holds that $\beta(G) = \beta(G') + 1$. On the other hand, we have that $\mathsf{pt}(G) = \frac{m}{2} + \frac{n-1}{4}$ and $\mathsf{pt}(G') = \frac{m-1}{2} + \frac{n-2}{4}$, and therefore $\beta(G) \geq \mathsf{pt}(G) + \frac{k}{4}$ if and only if $\beta(G') = \beta(G) - 1 \geq \mathsf{pt}(G) + \frac{k}{4} - 1 = \mathsf{pt}(G') + \frac{k'}{4}$. Thus, (G, k) is a YES-instance of SIGNED MAX CUT ATLB if and only if (G', k') is.   □

We are now ready to prove the main result of this subsection.

Theorem 5 *For any fixed integer $d \geq 1$,* SIGNED MAX CUT ATLB *on d^*-split graphs admits a kernel with at most $4(d+1)k$ vertices.*

Proof: Given a signed d^*-split graph $G = (V, E)$ and an integer k, let (K, I) be an arbitrary partition of V into a clique K and an independent set I such that every vertex in I has degree at most d and every vertex in K has at least one neighbor in I. We note that we do not even need to compute this partition (K, I) of V, as we will just use it for the analysis.

Let K_h be the set of vertices in K that have at least two neighbors in I, let $K_s = K \setminus K_h$ (so by hypothesis, every vertex in K_s has exactly one neighbor in I), let $I_h = N_I(K_h)$, and let $I_s = I \setminus I_h$. Note that since we assume that G is connected, it holds that $|I_s| \leq |K_s| \leq |K|$. We now state two claims that will allow us to certificate in some cases that (G, k) is a YES-instance.

Claim 4 *If $|K| \geq (d+1)k$, then (G, k) is a YES-instance.*

Proof: We apply to the input (G, k) the following procedure, assuming that G is connected (we stress that this algorithm is only used for the analysis; as mentioned before we do not modify our input graph at all):

1. If the current graph contains a vertex v in the clique with $c \geq 2$ neighbors in the independent set, let N be this set of neighbors, and do the following:

   ○ Apply Rule 6^+ to $\{v\} \cup N$, thus removing $\{v\} \cup N$ from the current graph and setting $k \leftarrow k - c + 1$. Note that the resulting graph is clearly connected, so Rule 6^+ can indeed be applied to $\{v\} \cup N$.

   ○ Go back to Step 1.

2. Apply exhaustively Rule A to the current graph, removing all vertices of degree 1. Clearly this operation preserves connectivity, and by Lemma 9 it produces an equivalent instance.

3. If the current graph contains a vertex v in the clique with exactly 1 neighbor u in the independent set, do the following:

   ○ Let w be a vertex in the clique that is non-adjacent to u; since the degree of u is at most d, such a vertex w is guaranteed to exist as long as the size of the current clique is at least $d + 1$.

   ○ Note that the removal of the vertices $\{v, u, w\}$ does not disconnect the current graph. Indeed, assume for contradiction that there exists a vertex z in the independent set that gets disconnected after the removal of $\{v, u, w\}$. Since Rule A cannot be applied anymore to the current graph, z cannot have degree 1, hence necessarily z is adjacent to both w and v. But then v has at least 2 neighbors u and z in the independent set, and this contradicts the fact that we are in Step 3, since Step 1 could be applied to v together with its neighborhood in the independent set.

   ○ Therefore, we can apply Rule 6^+ to $\{v, u, w\}$, thus removing them from the current graph and setting $k \leftarrow k - 1$.

   ○ Go back to Step 2.

4. Stop the procedure.

We now claim that if $|K| \geq (d+1)k$, then the instance (G', k') output by the above procedure satisfies that $k' \leq 0$, and since (G', k') is obtained from (G, k) by applying the one-way reduction Rule 6^+ and the two-way reduction Rule A, Lemma 8 and Lemma 9 imply that (G', k') is also a YES-instance, as we wanted to prove. Indeed, suppose that Step 1 has been applied p_1 times, that Rule A in Step 2 has been applied p_2 times, and that Step 3 has been applied p_3 times. Since in Step 1 it holds that $c \geq 2$, it follows that $k' \leq k - (p_1 + p_2 + p_3)$, so it suffices to prove that $p_1 + p_2 + p_3 \geq k$.

With slight abuse of notation, let us denote by K^* the set of vertices in the current clique that have some neighbor in the independent set. Note that as long as $|K^*| > 0$, Step 1, 2, or 3 can be applied to the current graph. For each application of Step 1, the size of K^* decreases by 1. For each application of Rule A in Step 2, the size of K^* decreases by at most 1. Finally, for each application of Step 3, the size of K^* decreases by at most $d + 1$ (at most d neighbors of vertex u together with vertex w may not belong to K^* anymore). Since by assumption G is a d^*-split graph, initially we have that $|K^*| \geq (d+1)k$, and therefore it follows that $p_1 + p_2 + p_3 \geq k$, concluding the proof. $\square$

Claim 5 *If* $|I_h| \geq 2dk$*, then* (G, k) *is a* YES-*instance.*

Proof: We prove that if $|I_h| \geq 2dk$, then we could iteratively apply Rule 6^+ to (G, k) until obtaining an instance (G', k') with $k' \leq 0$, which implies that G is a YES-instance.

We apply to (G, k) the following procedure (we stress again that this algorithm is only used for the analysis; as mentioned before we do not modify our input graph at all). We set $i := 1$, $G^1 := G$, $K_h^1 := K_h$, $I_h^1 := I_h$, and $k_1 := k$. Note that G^1 is connected by hypothesis. For the sake of readability, we want to stress that the sets K_h^i and I_h^i defined below will *not* correspond to the intersection of G^i with K_h and I_h, respectively. Proceed as follows:

1. Let v_i be an arbitrary vertex in K_h^i, let $N^i := N_{I_h^i}(v_i)$, and let $c_i := |N^i|$. Note that $c_i \geq 2$ and that $G^i - (\{v_i\} \cup N^i)$ is connected.

2. Apply Rule 6^+ to $\{v_i\} \cup N^i$, and let $G^{i+1} := G^i - (\{v_i\} \cup N^i)$.

3. Let D_K^i be the set of vertices in K_h^i having at most one neighbor in $I_h^i \setminus N^i$, and let $D_I^i := N_{I_h^i \setminus N^i}(D_K^i)$; see Fig. 6 for an example of these sets for $d = 3$. Since every vertex in K_h^i has at least two neighbors in I_h^i and each vertex in I_h^i has degree at most d, it follows that $|D_K^i| \leq (d-1)|N^i| = (d-1)c_i$. On the other hand, by definition it holds that $|D_I^i| \leq |D_K^i| \leq (d-1)c_i$.

4. Let $K_h^{i+1} := K_h^i \setminus (\{v_i\} \cup D_K^i \cup N_{K_h^i}(D_I^i))$, $I_h^{i+1} := I_h^i \setminus (N^i \cup D_I^i)$, and $k_{i+1} := k_i - c_i + 1$. Note that $|I_h^i| - |I_h^{i+1}| = |N^i| + |D_I^i| \leq c_i + (d-1)c_i = dc_i$, and that by construction, each vertex in K_h^{i+1} has at least two neighbors in I_h^{i+1}.

5. If $|I_h^{i+1}| > 0$, update $i \leftarrow i + 1$ and go back to Step 1. Otherwise, stop the procedure.

Assume that the above procedure has been applied $p \geq 1$ times, and let $k' := k_{p+1}$. Note that $k' = k - \sum_{i=1}^{p}(c_i - 1)$. Since k' is the parameter obtained from k by iteratively applying Rule 6^+ in G to the sets described in Step 2 above, our objective is to prove that $k' \leq 0$.
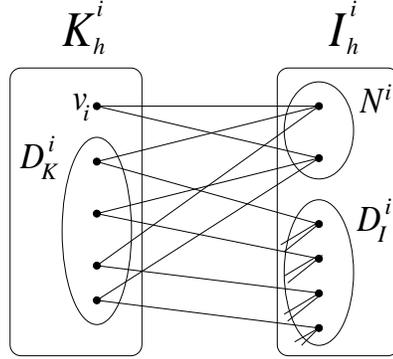
Fig. 6: Example for $d = 3$ of the sets defined in the proof of Theorem 5.

By the condition in Step 5 above, necessarily $|I_h^{p+1}| = 0$. Since for every $1 \leq i \leq p$ we have that, as discussed in Step 4 above, $|I_h^i| - |I_h^{i+1}| \leq dc_i$, using the hypothesis that $|I_h^1| = |I_h| \geq 2dk$ it follows that

$$2dk \leq |I_h^1| = (|I_h^1| - |I_h^2|) + (|I_h^2| - |I_h^3|) + \ldots + (|I_h^p| - |I_h^{p+1}|) = \sum_{i=1}^p (|I_h^i| - |I_h^{i+1}|) \leq d \cdot \sum_{i=1}^p c_i,$$

and therefore it follows that $\sum_{i=1}^p c_i \geq 2k$. As for every $1 \leq i \leq p$ it holds that $c_i \geq 2$ (see Step 1 above) and the function $x \mapsto x - 1$ is greater than or equal to the function $x \mapsto x/2$ for $x \geq 2$, we have that $\sum_{i=1}^p (c_i - 1) \geq \sum_{i=1}^p c_i/2 \geq k$. Finally,

$$k' = k - \sum_{i=1}^p (c_i - 1) \leq k - k = 0,$$

and therefore we can conclude that (G, k) is a YES-instance, as claimed. $\square$

Assume now that the hypothesis of Claim 4 and Claim 5 are both false, that is, that $|K| < (d+1)k$ and $|I_h| < 2dk$. Then we have that

$$|V| = |K| + |I| = |K| + |I_h| + |I_s| < (d+1)k + 2dk + (d+1)k < 4(d+1)k.$$

Thus, if the above inequality is not true, that is, if $|V| \geq 4(d+1)k$, then necessarily at least one of the hypothesis of Claim 4 or Claim 5 is true, and in any case we can safely conclude that (G, k) is a YES-instance. Therefore, our linear kernel is extremely simple: if $|V| \geq 4(d+1)k$, we report that (G, k) is a YES-instance, and otherwise we have that $|V| < 4(d+1)k$, as desired. $\square$

5 Conclusions

In this article we presented a quadratic kernel for SIGNED MAX CUT ATLB when the input graph is an (r, ℓ)-graph, for any fixed positive value of r and ℓ. Etscheid and Mnich [12] recently presented, among other results, a *linear* kernel for this problem on *general* graphs, improving the results of the current paper, as well as those of Crowston et al. [6]. Like us, the linear kernel of Etscheid and Mnich [12] also relies on a number of one-way reduction rules and results given by Crowston et al. [6]. The key idea is to use two new two-way reduction rules (similar to those of Crowston et al. [6]) that, roughly speaking, maintain the connectivity among the blocks of $G - S$. The analysis of the kernel size in [12] is considerably more involved that ours and entails, in particular, an elaborated "discharging" argument on the number of vertices removed by the rules.

Concerning FPT-algorithms, it may be possible that SIGNED MAX CUT ATLB can be solved in *subexponential* time on (r, ℓ)-graphs or, at least, on split graphs. We leave this question as an open problem.

Acknowledgements

References

[1] J. Baste, L. Faria, S. Klein, and I. Sau. Parameterized Complexity Dichotomy for (r, ℓ)-Vertex Deletion. *CoRR*, abs/1310.6205, 2015. To appear in *Theory of Computing Systems*, DOI: 10.1007/s00224-016-9716-y.

[2] H. L. Bodlaender and K. Jansen. On the complexity of the maximum cut problem. *Nordic Journal of Computing*, 7(1):14–31, 2000.

[3] A. Brandstädt. Partitions of graphs into one or two independent sets and cliques. *Discrete Mathematics*, 152(1-3):47–54, 1996.

[4] A. Brandstädt, V. B. Le, and J. P. Spinrad. *Graph Classes: a Survey*. SIAM, 1987.

[5] R. Crowston, G. Gutin, M. Jones, and G. Muciaccia. Maximum balanced subgraph problem parameterized above lower bound. *Theoretical Computer Science*, 513:53–64, 2013.

[6] R. Crowston, M. Jones, and M. Mnich. Max-Cut Parameterized Above the Edwards-Erdős Bound. *Algorithmica*, 72(3):734–757, 2015.

[7] M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*. Springer, 2015.

[8] R. Diestel. *Graph Theory*. Springer-Verlag, Berlin, 3rd edition, 2005.

[9] R. G. Downey and M. R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

[10] C. S. Edwards. Some extremal properties of bipartite subgraphs. *Canadian Journal of Mathematics*, 25:475–485, 1973.

[11] C. S. Edwards. An improved lower bound for the number of edges in a largest bipartite subgraph. *Recent Advances in Graph Theory*, pages 167–181, 1975.

[12] M. Etscheid and M. Mnich. Linear kernels and linear-time algorithms for finding large cuts. In *Proc. of the 27th International Symposium on Algorithms and Computation (ISAAC)*, volume 64 of *LIPIcs*, pages 1–13, 2016.

[13] T. Feder, P. Hell, S. Klein, and R. Motwani. List partitions. *SIAM Journal on Discrete Mathematics*, 16(3):449–478, 2003.

[14] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer-Verlag, 2006.

[15] S. Földes and P. Hammer. Split graphs. *Congressus Numerantium*, 19:311–315, 1977.

[16] M. R. Garey, D. S. Johnson, and L. Stockmeyer. Some simplified NP-complete graph problems. *Theoretical Computer Science*, 1(3):237–267, 1976.

[17] E. Ghosh, S. Kolay, M. Kumar, P. Misra, F. Panolan, A. Rai, and M. S. Ramanujan. Faster parameterized algorithms for deletion to split graphs. *Algorithmica*, 71(4):989–1006, 2015.

[18] M. Golumbic. *Algorithmic Graph Theory and Perfect Graphs*, volume 57 of *Annals of Discrete Mathematics*. Elsevier, 2004.

[19] F. Harary. On the notion of balance of a signed graph. *The Michigan Mathematical Journal*, 2(2):143–146, 1953.

[20] P. Heggernes, P. van 't Hof, E. J. van Leeuwen, and R. Saei. Finding disjoint paths in split graphs. *Theory of Computing Systems*, 57(1):140–159, 2015.

[21] S. Kolay and F. Panolan. Parameterized Algorithms for Deletion to (r, ℓ)-graphs. In *Proc. of the 35th IARCS Annual Conference on Foundation of Software Technology and Theoretical Computer Science (FSTTCS)*, volume 45 of *LIPIcs*, pages 420–433, 2015.

[22] R. Niedermeier. *Invitation to Fixed-Parameter Algorithms*. Oxford University Press, 2006.

[23] S. Poljak and D. Turzík. A polynomial time heuristic for certain subgraph optimization problems with guaranteed worst case bound. *Discrete Mathematics*, 58(1):99–104, 1986.

[24] V. Raman and S. Saurabh. Short Cycles Make W-hard Problems Hard: FPT Algorithms for W-hard Problems in Graphs with no Short Cycles. *Algorithmica*, 52(2):203–225, 2008.