Discrete Applied Mathematics 303 (2021) 137–148

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Fixed cardinality stable sets[☆]

Phillippe Samer [*], Dag Haugland

Universitetet i Bergen, Institutt for Informatikk, Postboks 7800, 5020, Bergen, Norway

ARTICLE INFO

Article history:
Received 30 November 2019
Received in revised form 22 July 2020
Accepted 11 January 2021
Available online 2 February 2021

Keywords:
Stable sets
Independent sets
Cardinality constraints
Combinatorial optimization
Integer programming
Graph classes

ABSTRACT

Given an undirected graph $G = (V, E)$ and a positive integer $k \in \{1, \ldots, |V|\}$, we initiate the combinatorial study of stable sets of cardinality exactly k in G. Our aim is to instigate the polyhedral investigation of the convex hull of fixed cardinality stable sets, inspired by the rich theory on the classical structure of stable sets. We introduce a large class of valid inequalities to the natural integer programming formulation of the problem. We also present simple combinatorial relaxations based on computing maximum weighted matchings, which yield dual bounds towards finding minimum-weight fixed cardinality stable sets, and particular cases which are solvable in polynomial time.

1. Introduction

We investigate a problem that is appealing to different research directions around algorithms, combinatorics and optimization. Let $G \stackrel{\text{def}}{=} (V, E)$ be a finite, simple, undirected graph, and denote $n \stackrel{\text{def}}{=} |V|$, and $m \stackrel{\text{def}}{=} |E|$. A stable set (or independent set, or co-clique) in G consists of a subset of pairwise non-adjacent vertices. Given $k \in \{1, \ldots, n\}$ and a vertex-weighting function $w : V \to \mathbb{Q}_+$, the *k stable set problem* consists in finding a minimum weight stable set of cardinality k in G, or deciding that none exists. Note that k is also part of the input to this problem; if it were an arbitrary fixed integer, the enumeration and optimization problems over stable sets of that cardinality could be solved in time polynomially bounded by a function of n.

1.1. Our contribution

The main idea of this work is to initiate the combinatorial study of fixed cardinality stable sets, introducing the first results in selected directions. We consider the problem first from the polyhedral standpoint, then we give efficiently computable dual bounds for the optimization problem, and conclude with graph classes where it can be solved in polynomial time. The different angles from which we study the problem are definitely inviting for further research. Indeed, some basic questions about our results are left open in the form of conjectures throughout the text.

To summarize, the contributions of this article include:

1. We draw attention to the fixed cardinality version of a classical structure in combinatorial optimization and graph theory, shedding light on its appeal to different research directions, besides motivating its application in the MSTCC problem.

☆ A preliminary version of this work appears in the conference proceedings of the 18th Cologne-Twente Workshop on Graphs and Combinatorial Optimization (Samer and Haugland, 2020) [28].

* Corresponding author.
E-mail addresses: samer@uib.no (P. Samer), dag.haugland@uib.no (D. Haugland).

https://doi.org/10.1016/j.dam.2021.01.019

2. We show in Section 2 that the fixed cardinality stable set polytope is not $\frac{1}{p}$-integral, for any integer $p > 1$. Thereafter we introduce an exponential class of valid inequalities to that polytope, whose separation problem is interesting in its own right.
3. We describe a combinatorial relaxation of the optimization problem in Section 3, where lower bounds are calculated using maximum weighted matchings. Given the efficiency of the corresponding algorithm, this technique can be extended as a building block in different solution approaches.
4. We prove in Section 4 that the problem can be solved in polynomial time when the input is restricted to some important graph classes, including cluster, complete multipartite, split, threshold, and line graphs.

1.2. Motivation from conflict-free spanning trees

Our original motivation for considering fixed cardinality stable sets stems from the NP-hard problem of *minimum spanning trees under conflict constraints* (MSTCC). Given a graph $G \stackrel{\text{def}}{=} (V, E)$ and a set of conflicting edge pairs $C \subseteq E \times E$, a *conflict-free spanning tree* in G is a set of edges $T \subseteq E$ inducing a spanning tree in G, such that for each $(e, f) \in C$, at most one of the edges e and f is in T. The MSTCC problem, introduced by Darmann et al. [13,14], asks for such a conflict-free spanning tree of minimum weight.

Different combinatorial and algorithmic results about the MSTCC problem explore the associated conflict graph $H \stackrel{\text{def}}{=} (E, C)$, which has a vertex corresponding to each edge in the original graph G, and we represent each conflict constraint by an (undirected) edge connecting the corresponding vertices in H. Note that each conflict-free spanning tree in G is a subset of E which corresponds both to a spanning tree in G and to a stable set in H. Therefore, one can equivalently search for stable sets in H of cardinality exactly $|V| - 1$ which do not induce cycles in the original graph G.

It is not hard to devise different approaches for studying the MSTCC problem exploring the connection with fixed cardinality stable sets. For the sake of illustration, consider the *relax-and-cut* approach described by Lucena [20] for the fixed cardinality set partitioning problem. The author of that work developed a Lagrangean framework where dual bounds, heuristics and variable fixing tests are computed as a preprocessing phase, resulting in an easier problem to be handled to an integer programming (IP) solver. Note that the Lagrangean bounds are strengthened by dynamically introducing valid constraints, including those from the exponential family of clique inequalities. Now, an analogue towards conflict-free spanning trees could be described as follows. Given the original graph $G = (V, E)$, the conflict graph $H = (E, C)$, and costs $c \in \mathbb{Q}_+^{|E|}$ on edges of G, denote by $\mathfrak{C} \stackrel{\text{def}}{=} \mathfrak{C}(H, |V| - 1)$ the polytope of stable sets in H which have cardinality equal to $|V| - 1$. Using binary variables $x \in \{0, 1\}^{|E|}$, we recast the MSTCC problem as

$$\min \{cx : Ax \leq b, x \in \mathfrak{C}\}, \tag{1}$$

where the system $\{a_i x \leq b_i\}_{i=1}^m$ corresponds to the subtour elimination constraints (SEC): $\left\{\sum_{e \in E(S)} x_e \leq |S| - 1 : S \subset V, S \neq \emptyset\right\}$. Thus, the number m of inequalities is an exponential function of $|V|$. Dualizing all the SEC, with the introduction of multipliers $\lambda \in \mathbb{R}_+^m$, we have a lower bound to (1) given by the *Lagrangean Relaxation Problem*

$$LRP(\lambda) \stackrel{\text{def}}{=} \min \{(c + \lambda A)x - \lambda b : x \in \mathfrak{C}\}, \tag{2}$$

and the best-possible bound is attained by solving the *Lagrangean Dual Problem*

$$LDP \stackrel{\text{def}}{=} \max \left\{LRP(\lambda) : \lambda \in \mathbb{R}_+^m\right\}. \tag{3}$$

There are two main challenges in this approach. First, the issue of dualizing exponentially many inequalities is dealt with (in a subgradient method) by a clever selection of active constraints among those which are currently or previously violated, while arbitrarily setting to zero the subgradient vector entries corresponding to null multipliers; see [20, Section 1.2]. The second issue is how to optimize over $\mathfrak{C}$, to solve $LRP(\lambda)$ in (2). In order of decreasing generality, we note that:

(i) The obvious relaxation would have been to also dualize edge inequalities in H (that is, $x_u + x_v \leq 1$ for $\{u, v\} \in C$), introducing a new set of Lagrangean multipliers $\mu \in \mathbb{R}_+^{|C|}$, and solving instead the easy problem

$$LRP'(\lambda, \mu) \stackrel{\text{def}}{=} \min \left\{(c + \lambda A + \mu M)x - \lambda b - \mu : \sum_{e \in E} x_e = |V| - 1\right\}, \tag{4}$$

where M denotes the incidence matrix of the conflict graph H.

(ii) If more information on H is available (*e.g.* sparsity, structural decomposition), that could be translated as a better approximation of $\mathfrak{C}$ in the relaxed problem. For instance, if there is a natural decomposition of H into few connected components, one could design instead the special case of a Lagrangean Decomposition, with different primal variables for each component, dualizing the constraints equating the different variables [16, Section 7]. Alternatively, if strong valid inequalities for $\mathfrak{C}$ are known, they could be used towards a relaxation which is between (4) and (2).

(iii) If H belongs to a graph class where the *fixed* cardinality stable set problem becomes solvable in polynomial time, then we can solve problems (2) and (3) as stated above. Note that stronger bounds should follow in this case, since more problem information is embedded in the relaxation.[1]

Our presentation of this first relax-and-cut approach for the MSTCC problem is limited to the above outline. We argue that results of different nature from research on the k stable set problem (*e.g.* integer programming formulations and valid inequalities, well-solved particular cases, primal and dual bounds) could provide fundamental components to advance knowledge on the MSTCC problem as well.

1.3. Further related work

It is surprising that the combinatorics and optimization literature has not addressed the *k stable set problem* problem in depth before. Note, for instance, that the thorough survey on fixed cardinality versions of combinatorial optimization problems by Bruglieri et al. [7] does not mention stable sets, in spite of the major role played by that structure throughout the development of polyhedral combinatorics.

The convex hull of stable sets of cardinality *at most* k was studied by Janssen and Kilakos [18], but only for $k \in \{2, 3\}$. Apart from that article, it has also appeared as part of an algorithm for a variant of the survivable network design problem [6, Chapter 2], where only an alternative proof of one of the original results by Janssen and Kilakos [18] is given.

Thin graphs and frequency assignment problems. The early work of Mannino et al. [21] introduces an interesting class of graphs, as well as a cardinality-constrained stable set problem, and their application in the efficient solution of real-world instances of a frequency assignment problem. We explain next the result which is most relevant to our work, and also derive an initial fact about the problem that we study.

Given an ordering $\{v_1, \ldots, v_n\}$ of the vertices of a graph $G = (V, E)$, and a partition $V = \biguplus_{i=1}^{k} V_i$, let $\overline{N}(v_j, h)_<$ denote the set of vertices in V_h of order lower than j which are non-adjacent to v_j. The ordering and the partition are called *consistent* if the only vertices in V_h of order lower than j and non-adjacent to v_j are the first $\left|\overline{N}(v_j, h)_<\right|$ ones.

A graph G is *k-thin* if there is such an ordering of the vertices and a partition of V into k classes which are consistent. The *thinness* of a graph is the smallest k such that G is k-thin. Now, if $k = 1$, this gives a characterization of *interval graphs* (those graphs for which an intersection model consisting of intervals on a straight line can be defined). Specifically, $G = (V, E)$ is an interval graph if and only if there exists an ordering $\{v_1, \ldots, v_n\}$ which is consistent (with $V = V_1$, the trivial partition).

Then, the authors prove:

Theorem (*2.12 in [21]*)**.** *Given $G = (V, E)$, together with an ordering and a partition $V = \biguplus_{i=1}^{k} V_i$ which are consistent, and $d \in \mathbb{Z}_+^k$, a maximum (minimum) weighted stable set S, with $|S \cap V_i| = d_i$ for each $i \in \{1, \ldots, k\}$, can be determined in time $O\left(|V| \cdot (\rho + 1)^{k-1} \cdot (1 + \max_{1 \le i \le k} d_i)^k\right)$, where ρ denotes the largest amount of neighbours of lower order that a vertex has in some class of the partition (thus, $\rho \le \Delta(G)$, the largest degree of a vertex in the graph).*

Note that they have therefore introduced a different cardinality-constrained version of the stable set problem, more general than the one we study in this work. Thereafter, the excellent computational results on large instances of the frequency-assignment application they describe depend crucially on the efficient solution of this generalized problem on a $|H|$-thin conflict graph, where H is a special set of transmitters in the input.

Finally, we remark that setting $k = 1$, although not interesting in their application, gives an initial result for our problem of interest. For $k = 1$, the consistent ordering in the above theorem implies that G is an interval graph, and that an optimal stable set of fixed cardinality d can be found in time bounded by $O\,(n \cdot (1 + d))$, which is in $O\left(n^2\right)$.

Corollary 1. *If G is an interval graph, the problem of finding a minimum-weight stable set of fixed cardinality in G is in* P.

Extension complexity. The fixed cardinality stable set polytope also appears briefly in the recent and rapidly developing theory of parameterized extension complexity. This line of research aims to develop bounds on the number of inequalities necessary to describe a given polytope as the projection of a higher dimensional one. While that number can be polynomially bounded (as a function of the number of vertices in the input graph) for a few particular cases of the classical stable set and vertex cover polytopes, some striking negative results show how large that number can be in general.

There are two main categories of such results in the current literature. One is proving the hardness of approximating a polytope by an extended formulation, such as the work of Bazzi et al. [2], who prove that for all n sufficiently large, there exist graphs on n vertices such that every linear programming (LP) or even semidefinite programming (SDP) relaxation of polynomial size for the stable set polytope on those graphs has integrality gap of $\omega(1)$.

Another category is designing (exact) extended formulations of fixed-parameter tractable (FPT) size complexity. Still on the negative side, it was recently proved by Gajarský et al. [15] that, regardless of any computational complexity

[1] Contrast with the fact that, even if the classical stable set problem on H can be solved in polynomial time, the MSTCC problem with H as a conflict graph need not be solvable in polynomial time (the original NP-hardness proof of Darmann et al. [14] makes the further assumption that the conflict graph is a collection of disjoint paths of length 2).

assumptions, the stable set polytope cannot have a FPT extension for all graphs (naturally parameterized by the solution size). On the positive side, the authors show that linear size FPT extensions do exist for the class of bounded expansion graphs. Even before, it was proved by Buchanan and Butenko [9] that, when parameterized by the treewidth of G, $\mathbf{tw}(G)$, the extension complexity of the stable set polytope on G is in $O(2^{\mathbf{tw}(G)}n)$. Afterwards, Buchanan [8] proved bounds for FTP extended formulations for vertex cover polytopes parameterized by the solution size. Interestingly, in a lemma towards his main result, the author proves that the fixed cardinality stable set polytope for graphs of largest degree at most 2 is given by edge and odd-cycle inequalities alone.

2. Polyhedral results

For any graph G, we denote by $V(G)$ and $E(G)$ the sets of vertices and edges of G, respectively. For conciseness, we abbreviate 'stable set of cardinality k' as k-stab. The family of all k-stabs in G is denoted $\mathcal{F}(G,k)$. Recall that the incidence vector of any $S \subset V$ is $\chi^S \in \{0,1\}^V$ defined by $\chi^S_i = 1$ if $i \in S$, and $\chi^S_i = 0$ if $i \in V \backslash S$; so the central object of our interest is $\mathfrak{C}(G,k) \overset{\text{def}}{=} \mathbf{conv}\left\{\chi^S : S \in \mathcal{F}(G,k)\right\}$, *i.e.* the convex hull of incidence vectors of all the k-stabs in G.

The natural integer programming (IP) formulation for minimum-weight k-stabs in G is

$$z \overset{\text{def}}{=} \min \left\{ \sum_{v \in V} w(v) x_v : \mathbf{x} \in \mathcal{P}(G,k) \cap \{0,1\}^n \right\}, \tag{5}$$

where $\mathcal{P}(G,k)$ denotes the polyhedral region defined by:

$$\sum_{v \in V} x_v = k \tag{6}$$

$$x_u + x_v \leq 1 \qquad \forall \{u,v\} \in E \tag{7}$$

$$0 \leq x_v \leq 1 \qquad \forall v \in V \tag{8}$$

Constraints (7) are known as *edge inequalities*, imposing that no two adjacent vertices belong to the selection in $\mathbf{x}$. Together with bounds (8), they determine the *fractional stable set polytope* [29, Section 64.5].

Recall that a vector z is *half-integer* if $2z$ is integer (more generally, we say that z is $\frac{1}{p}$*-integer* if pz is integer). A classical result of Nemhauser and Trotter [24] shows that the *fractional stable set polytope* is half-integer, *i.e.* all its vertices are $\left\{0, \frac{1}{2}, 1\right\}$-valued. Since that is the starting point for a series of both polyhedral and algorithmic advances, one could ask whether that result holds for $\mathcal{P}(G,k)$ as well. Unfortunately, we discovered the negative answer to an even broader question, as we show next.

Theorem 2. *For each $p \geq 2$ and each $k \geq 2$, there exists a graph G such that $\mathcal{P}(G,k)$ is not $\frac{1}{p}$-integer.*

Proof. Given $p \geq 2$ and $k \geq 2$ arbitrary, we determine $n \in \mathbb{Z}_+$, a graph G on n vertices, and a convenient point $z \in \mathbb{R}^n$ such that z is a vertex of the polyhedron $\mathcal{P}(G,k)$ which is not $\frac{1}{p}$-integer.

First, we choose $n = n(p,k)$ such that the point $z \overset{\text{def}}{=} (\overbrace{1/p+1\,, \;\cdots, 1/p+1}^{n-1 \text{ entries}}\,, \; p/p+1)$ satisfies the equality constraint (6): $\sum_{v \in V} x_v = k$. That is,

$$z_1 + \cdots + z_n = \frac{1}{p+1} + \cdots + \frac{1}{p+1} + \frac{p}{p+1} = (n-1)\frac{1}{p+1} + \frac{p}{p+1} = k, \tag{9}$$

and we therefore set $n \overset{\text{def}}{=} p(k-1) + k + 1$. Consider next $G \overset{\text{def}}{=} S_{n-1} = K_{1,n-1}$, the star on n vertices (illustrated in Fig. 1).

We can show that z is a vertex of $\mathcal{P}(S_{n-1},k) \subset \mathbb{R}^n$ using the equivalence of vertices, basic feasible solutions and extreme points of polyhedra; see *e.g.* [4, Section 2.2]. Besides satisfying all equality constraints, a *basic solution* of a polyhedron embedded in $\mathbb{R}^n$ must have n constraints

(i) which are active (equiv. satisfied at equality) at z, and
(ii) whose corresponding vectors in $\mathbb{R}^n$ are linearly independent.

The equality constraint (6) is satisfied by construction of the point z in (9). The graph $G = S_{n-1}$ has an edge $\{v_i, v_n\} \in E(G)$ for each $i \in \{1,\ldots,n-1\}$, and the corresponding edge inequality (7) is active at z: $x_{v_i} + x_{v_n} = \frac{1}{p+1} + \frac{p}{p+1} = 1$. It remains to verify that the coefficient vectors of those n constraints are linearly independent. Indeed, arranging the vectors as rows of matrix $A_{n\times n}$,

$$A_{n\times n} = \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \cdots & a_{1,n-1} & a_{1,n} \\ a_{2,1} & a_{2,2} & a_{2,3} & \cdots & a_{2,n-1} & a_{2,n} \\ a_{3,1} & a_{3,2} & a_{3,3} & \cdots & a_{3,n-1} & a_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n,1} & a_{n,2} & a_{n,3} & \cdots & a_{n,n-1} & a_{n,n} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 & 1 \\ 1 & 0 & 0 & \cdots & 0 & 1 \\ 0 & 1 & 0 & \cdots & 0 & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 1 \end{pmatrix},$$

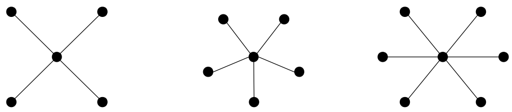

Fig. 1. The star graphs S_4, S_5 and S_6.

it follows that $\det A = (-1)^n(n-2) \neq 0$, and the row vectors are indeed linearly independent. Since z is also feasible (as the bounds $0 < 1/p+1 < p/p+1 < 1$ are satisfied by construction), it is a basic feasible solution, and thus a vertex of $\mathcal{P}(S_{n-1}, k)$.

Finally, $p \cdot \frac{1}{p+1} = \frac{p}{p+1}$ is not an integer (but a proper fraction), and it is clear that pz is not an integer point. Therefore, the vertex z is not $\frac{1}{p}$-integer, and the result follows. □

2.1. The unsuitable neighbourhood inequalities (UNI)

We introduce next a class of valid inequalities for $\mathfrak{C}(G, k)$, exploring the relationship between k, the size of the neighbourhood $N(S) \stackrel{\text{def}}{=} \{u \in V \backslash S : \exists \{u, v\} \in E \text{ for some } v \in S\}$ of any set $S \subset V$, and how many vertices from S can appear in any k-stab. First, denoting the set of neighbours of a vertex $v \in V$ by $\delta(v)$, that is $\delta(v) = N(\{v\})$, one can immediately observe that no vertex which has too many neighbours to still build a k-stab can be chosen. This gives the following simple preprocessing test.

Proposition 3. *If* $\mathbf{x}$ *is the incidence vector of any k-stab, and* $v \in V$ *is such that* $|\delta(v)| > n - k$, *then* $x_v = 0$.

In an attempt to enforce an algebraic expression that enough vertices are left upon choosing a set $S \subset V$ towards building a k-stab, we introduce a class of exponentially-many constraints, which we refer to as *unsuitable neighbourhood inequalities* (UNI).

Theorem 4. *For each* $S \subset V$ *such that* $1 \leq |S| < k$ *and* $|N(S)| > n - k$, *inequality* $\sum_{v \in S} x_v \leq |S| - 1$ *is valid for* $\mathfrak{C}(G, k)$.

Proof. From $|S| < k$, it follows that S is not a k-stab in itself. If S were a subset of any k-stab, there should be at least $k - |S|$ vertices left to choose from, while no neighbour in $N(S)$ can be selected towards building a stable set. That is

$$\begin{aligned} & n - |S| - |N(S)| \geq k - |S| \,, && \forall S \subset V, 1 \leq |S| < k, \\ \Leftrightarrow \quad & |N(S)| \leq n - k \,, && \forall S \subset V, 1 \leq |S| < k. \end{aligned}$$

Since $|N(S)| > n - k$ by hypothesis, S cannot be part of a k-stab. Therefore no incidence vector $\mathbf{x}$ of a k-stab induces the selection of all the vertices in S, and the result follows. □

While Proposition 3 is clearly a special case of Theorem 4, one could ask whether the UNI indeed give a stronger condition. The positive answer follows next.

Theorem 5. *For any graph* G *and* $k > 1$, *the UNI imply the condition enforced by Proposition 3 in the description of* $\mathfrak{C}(G, k)$, *but the converse does not hold.*

Proof. Let $\mathbf{x}$ be a vector satisfying all UNI. The inequalities in Proposition 3 are implied by the UNI with $|S| = 1$. Suppose that $S = \{u\}$ and $|N(S)| = |\delta(u)| > n - k$. Then u cannot be extended to a k-stab and the UNI include $x_u = \sum_{v \in S} x_v \leq |S| - 1 = 0$, which is the condition on the former proposition.

Now the converse does not hold, *i.e.* even if $|\delta(v)| \leq n - k$ for each $v \in V$, the UNI need not be automatically satisfied, as the following counterexample shows (see Fig. 2). Consider the graph $G \stackrel{\text{def}}{=} 2P_3$, which consists of two copies of the path graph on 3 vertices put together, so that $n = 6$, and suppose that $k \stackrel{\text{def}}{=} 3$. Since all vertices have degree 1 or 2, it follows that $|\delta(u)| \leq n - k = 3$ for each vertex u. On the other hand, with a test set S consisting of the two vertices of degree 2 in the middle of the paths, we have $1 \leq |S| < k$ and $|N(S)| = 4 > n - k$, thus yielding the unsuitable neighbourhood inequality given by $\sum_{v \in S} x_v \leq |S| - 1 = 1$ which separates from the convex hull $\mathfrak{C}(G, k)$ any vector selecting those two vertices. □

Proposition 6. *In either of the following two conditions, the corresponding unsuitable neighbourhood inequality is redundant in* $\mathfrak{C}(G, k)$: *(i) if* $S \subset V$ *is not independent, or (ii) if* $S \subset V$ *is not minimal with respect to the condition* $|N(S)| > n - k$.

Proof. If $u, v \in S$ are adjacent vertices, the edge inequality $x_u + x_v \leq 1$ implies $\sum_{v \in S} x_v \leq |S| - 1$.

Otherwise, let $S \subset V$ with $1 \leq |S| < k$ and $N(S) > n - k$ be a given independent set, and suppose that $T \subsetneq S$ is such that $|N(T)| > n - k$. The UNI corresponding to T is $\sum_{v \in T} x_v \leq |T| - 1$. Combined with $x_v \leq 1$ for each $v \in S \backslash T$, it implies the UNI corresponding to S, *i.e.* $\sum_{v \in S} x_v \leq |S| - 1$, which is thus redundant in the description of $\mathfrak{C}(G, k)$. □

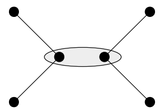

Fig. 2. The graph $2P_3$ and the selection of its two central vertices.

Recall that the *domination number* $\gamma(G)$ gives the least cardinality of a dominating set in $G = (V, E)$, *i.e.* a subset $D \subset V$ such that every vertex $u \in V\backslash D$ has a neighbour in D. If a lower bound on the domination number of G is known, the following result might be useful.

Proposition 7. *If $\gamma(G) \geq k$, then there exists no UNI for $\mathfrak{C}(G, k)$.*

Proof. Suppose there were $S \subset V$ with $1 \leq |S| < k$ and $|N(S)| > n - k$, and denote $T \stackrel{\text{def}}{=} V\backslash\{S \cup N(S)\}$. Note that any vertex belongs to exactly one among S, $N(S)$, or T; then

$$|S| + |N(S)| + |T| = n \implies |S| + |T| = n - |N(S)| \implies |S| + |T| < n - [n - k] = k,$$

since $|N(S)| > n - k$. Now, $S \cup T$ would be a dominating set of cardinality strictly less than k, contradicting the hypothesis that $\gamma(G) \geq k$. □

On the algorithmic side, it is in general impractical to include *a priori* all minimal UNI in an IP formulation for a black-box solver, since the number of those inequalities may grow exponentially with the size of the input (n, k). The natural approach in this case is to try to cut off successive solutions x^* to a linear programming (LP) relaxation, by finding cutting planes corresponding to UNI violated at x^*, *i.e.* separating x^* from $\mathfrak{C}(G, k)$, or deciding that none exists. Answering that question is known as the *separation problem* for a class of valid inequalities.

Definition 8 (*Separation Problem for UNI*)**.** Given a graph $G = (V, E)$, with $n \stackrel{\text{def}}{=} |V|$, $k \in \{2, \ldots, n-1\}$, and $x^* \in [0, 1]^n$ satisfying the conditions that $\sum_{v \in V} x^*_v = k$ and that $x^*_u + x^*_v \leq 1$ for each $\{u, v\} \in E$, determine

i. either a set $S \subset V$, with $1 \leq |S| \leq k-1$ and $|N(S)| \geq n - (k-1)$, such that $\sum_{v \in S} x^*_v > |S| - 1$, in which case the unsuitable neighbourhood inequality corresponding to S separates x^* from $\mathfrak{C}(G, k)$,
ii. or that no such set exists, in which case all UNI are satisfied at x^*.

We give next a slight reformulation of the separation problem which might be useful in future work. Given the input $[G, k, x^*]$ corresponding to Definition 8, define $y^* \in [0, 1]^n$ such that $y^*_v \stackrel{\text{def}}{=} 1 - x^*_v$. Note now that $\sum_{v \in S} x^*_v > |S| - 1$ if and only if $\sum_{v \in S} y^*_v < 1$. We thus have the following equivalent statement of the problem.

Definition 9 (*Equivalent Formulation of the Separation Problem for UNI*)**.** Given a graph $G = (V, E)$, with $n \stackrel{\text{def}}{=} |V|$, $k \in \{2, \ldots, n-1\}$, and $y^* \in [0, 1]^n$ satisfying the conditions that $\sum_{v \in V} y^*_v = n - k$ and that $y^*_u + y^*_v \geq 1$ for each $\{u, v\} \in E$, determine

i. either a set $S \subset V$, with $|N(S)| \geq n-(k-1)$ and $\sum_{v \in S} y^*_v < 1$, in which case the unsuitable neighbourhood inequality corresponding to S separates $x^* \stackrel{\text{def}}{=} \mathbf{1} - y^*$ from $\mathfrak{C}(G, k)$,
ii. or that no such set exists, in which case all UNI are satisfied at $x^* \stackrel{\text{def}}{=} \mathbf{1} - y^*$.

We consider this statement of the problem to be particularly appealing. Note that if S has size exactly $k - 1$, then $|N(S)| \geq n - (k-1)$ implies that it would be a dominating set. Given the condition that adjacent vertices have y^* values summing up to at least 1, and that we require $\sum_{v \in S} y^*_v < 1$, we would actually have an *independent dominating set* if $|S| = k - 1$, *i.e.* a subset of vertices which is both dominating and independent (stable). Now, allowing $|S| \leq k-1$ means that there might be $q \in \{0, 1, \ldots, k-2\}$ vertices neither in S nor dominated by it. If we define a *q-quasi dominating set* in a graph $G = (V, E)$ to be a subset of vertices which is dominating in $G[V\backslash X]$, for some $X \subset V$, $|X| \leq q$, our separation problem corresponds to finding a $(k-2)$-quasi dominating set of weight at most 1, or deciding that none exists. (Recall that, for any graph G and $U \subset V(G)$, the *induced subgraph* $G[U]$ is a graph with vertex set U and all of the edges in $E(G)$ which have both endpoints in U.)

We leave the open question of establishing the complexity of that problem.

Conjecture 1. *The separation problem for UNI is* NP*-hard.*

2.2. UNI separation with MIP heuristics

We discuss next an alternative to actually use the UNI in a branch-and-cut solver. This part of the text is only interesting under the assumption that the above conjecture is true.

Besides the natural strategies of designing separation heuristics or including *a priori* some UNI corresponding to sets S of small cardinality, it might prove useful to explore an IP formulation of the separation problem. One can actually use good but not necessarily optimal solutions to that auxiliary IP, which give very effective cutting planes, for instance, in the context of an example of optimizing over the first Chvátal closure [5, Section 5.4]. Most MIP solvers include a collection of general purpose heuristics to accelerate the availability of integer feasible solutions, like local branching, feasibility pump and neighbourhood diving methods; see [17] for a recent survey.

The following is described in light of Definition 9, with input $[G, k, \mathbf{y}^*]$. We suppose further that the input is preprocessed by the reduction rules:

(i) Remove any vertex v such that $y^*_v = 1$
(ii) Remove isolated vertices

Those operations do not change the problem answer, since a UNI is automatically satisfied if it contains a vertex with $y^*_v = 1$, and since isolated vertices are not contained in a minimal set S corresponding to a UNI.

For each $v \in V$, let variables $z_v \in \{0, 1\}$ be such that $z_v = 1$ if and only if $v \in S$, and $w_v \in \{0, 1\}$ be such that $w_v = 1$ if and only if $v \in N[S] = S \cup N(S)$, the *closed neighbourhood* of $S \subset V$. Then, we have to determine

$$\rho = \min \left\{ \sum_{v \in V} y^*_v \cdot z_v : (\mathbf{z}, \mathbf{w}) \in \mathcal{P}_{\mathrm{UNI}}(G, \mathbf{y}^*) \cap \{0, 1\}^{2n} \right\}, \tag{10}$$

where $\mathcal{P}_{\mathrm{UNI}}(G, \mathbf{y}^*)$ denotes the polyhedral region:

$$\sum_{v \in V} (w_v - z_v) \geq n - (k - 1) \tag{11}$$

$$z_u \leq w_v \qquad \forall u \in V, \forall v \in N[u] \tag{12}$$

$$\sum_{u \in N[v]} z_u \geq w_v \qquad \forall v \in V \tag{13}$$

$$z_u + z_v \leq 1 \qquad \forall \{u, v\} \in E \tag{14}$$

$$0 \leq z_v \leq 1 \qquad \forall v \in V \tag{15}$$

$$0 \leq w_v \leq 1 \qquad \forall v \in V \tag{16}$$

The objective function in (10) accounts for the used $\mathbf{y}^*$ budget, as prescribed in Definition 9. Inequality (11) guarantees the minimum number of vertices dominated by S (excluding those which are in S). Inequalities (12) and (13) bind the binary variables $\mathbf{w}$ and $\mathbf{z}$, to enforce the domination condition that $w_v = 1$ if and only if $z_u = 1$ for some $u \in N[v]$.

Inequalities (14) are redundant, being implied at integer points in $\mathcal{P}_{\mathrm{UNI}}(G, \mathbf{y}^*)$ by (11) and the fact the input parameter satisfies $y^*_u + y^*_v \geq 1$ for each $\{u, v\} \in E$. Still, adding those inequalities is likely to tighten the LP relaxation bounds, and hence speed up the overall optimization procedure.

The exact separation problem thus reduces to deciding if $\rho < 1$. The MIP heuristic, on the other hand, consists of searching (*e.g.* allowing a MIP solver to run with a prescribed time limit) for any integer feasible solution $(\mathbf{z}', \mathbf{w}')$ with an objective value less than 1, which determines the UNI $\sum_{v \in S'} x_v \leq |S'| - 1$, with $S' = \left\{ v \in V : z'_v = 1 \right\}$, violated at $x^* = \mathbf{1} - y^*$.

3. Combinatorial dual bounds

We concern next the possibility to compute dual bounds to problem (5) via a *combinatorial relaxation*, *i.e.* computing a lower bound to $z \stackrel{\text{def}}{=} \min \left\{ \sum_{v \in V} w(v) x_v : \mathbf{x} \in \mathcal{P}(G, k) \cap \{0, 1\}^n \right\}$ through a relaxation which is a new combinatorial optimization problem, and which is more tractable or interesting, for some reason. For instance, a key ingredient in a recent matheuristic for a class of generalized set partitioning problems [27] is an efficiently computable combinatorial bound similar to the ones introduced here, even though the actual bounds are weaker than LP relaxation ones.

In this section, we write $V = \{v_1, \ldots, v_n\}$, with the vertices indexed by non-decreasing weight, so that $w(v_1) \leq w(v_2) \leq \cdots \leq w(v_n)$. Note that the most naïve lower bound corresponds to the selection of the k vertices of least weight in G. That is,

$$z \geq \sum_{i=1}^{k} w(v_i), \tag{17}$$

which corresponds to relaxing all of the edge inequalities (7) in the definition of $\mathcal{P}(G, k)$. We introduce a simple way of relaxing fewer of those inequalities.

Recall that a *matching* in a graph is a subset of pairwise non-adjacent edges, that is, a subset of edges without common vertices. While the facial structure of the matching polytope and combinatorial algorithms to find a maximum weighted matching in a graph are well-known, the following result is less frequently used.

Remark 10. Finding a minimum-weight matching of a specified cardinality in a graph is a well-solved problem. More generally, for any $l, u \in \mathbb{Z}_+$, $l \leq u$, the convex hull of incidence vectors of matchings $M \subset E(G)$ such that $l \leq |M| \leq u$ is equal to the set of those vectors in the matching polytope of G satisfying $l \leq \mathbf{1}^\top x \leq u$, that is, $l \leq \sum_{e \in E(G)} x(e) \leq u$ [29, Section 18.5f].

Theorem 11. *Suppose that $\mathcal{P}(G, k) \cap \{0, 1\}^n \neq \emptyset$, so that z is well-defined in problem (5), and let $S \stackrel{def}{=} \{v_1, \dots, v_k\}$.*

(i) Let $M \subset E$ be any matching in the induced subgraph $G[S]$. Then

$$b_1(M) \stackrel{def}{=} \sum_{i=1}^{k} w(v_i) + \sum_{\substack{\{v_i, v_j\} \in M, \\ i<j}} \left[w(v_{k+1}) - w(v_j)\right]$$

is a lower bound on z.

(ii) Let $\nu \in \left\{1, \dots, \left\lfloor \frac{k}{2} \right\rfloor\right\}$ denote the maximum cardinality of a matching in $G[S]$. For $1 \leq q \leq \nu$, let $M_q \subset E$ be any matching in $G[S]$ such that $|M_q| = q$. Then

$$b_2\left(M_1, \dots, M_\nu\right) \stackrel{def}{=} \max_{1 \leq q \leq \nu} \left\{ b_1(M_q) + \sum_{h=2}^{q} \left[w(v_{k+h}) - w(v_{k+1})\right] \right\}$$

is a lower bound on z.

Proof. Note first that S is the vertex selection giving the trivial bound (17), which is also the first summand in the definition of b_1. If there exists $\{v_i, v_j\} \in E$, for any $1 \leq i < j \leq k$, then S is not feasible and (17) is not tight. Since some (possibly all) vertices in S need to be *replaced* towards finding an optimal k-stab, an optimistic approach would be to consider disjoint pairs of vertices which are not compatible with each other in S, *i.e.* a matching in $G[S]$, and that we could form a k-stab by exchanging the vertex with larger weight in each pair with a hypothetical vertex in $G \backslash S$ with the least possible weight: $w(v_{k+1})$. Surely, there might not exist enough vertices with such weight in $G \backslash S$, and even if that is the case, the new pair could be incompatible (that is, adjacent in G). Still, to assume an additional value of $w(v_{k+1}) - w(v_j)$ to make feasible each matched pair of vertices $\{v_i, v_j\}$, $i < j$, yields a lower bound on the optimal value of z. This proves *(i)*, where only those additional values are added to the naïve bound.

The proof of *(ii)* follows from the same reasoning, while performing slightly less optimistic exchanges. Previously, we assumed the availability of enough vertices with weight $w(v_{k+1})$ in $G \backslash S$ to replace the one with larger weight on each matched pair in M. Now, we still get a lower bound on z if we use the actual weight of that many vertices among the ones with lowest weight in $G \backslash S$. More precisely: given matching M_q, to assume that replacing $\left\{v_j : \text{ for each } \{v_i, v_j\} \in M_q, i < j\right\}$ by $\{v_{k+h} : \text{ for } 1 \leq h \leq q\}$ would give a k-stab is still a relaxation of problem (5). We can thus increase bound $b_1(M_q)$, where we assumed $w(v_{k+1}) - w(v_j)$ would suffice to replace each v_j, by the accumulated differences $\sum_{h=2}^{q} \left[w(v_{k+h}) - w(v_{k+1})\right]$, and still get a lower bound on z. Finally, since we cannot anticipate which matching gives the greatest weight increase, we take the maximum bound among the ones attained by different matching cardinalities in $G[S]$. □

Remark 12. For each edge $\{v_i, v_j\} \in E(G[S])$, with $i < j$, let $c(\{v_i, v_j\}) \stackrel{\text{def}}{=} \left[w(v_{k+1}) - w(v_j)\right]$. Then, taking M to be a maximum weighted matching in $G[S]$ with edge weights given by c gives the strongest bound $b_1(M)$ in Theorem 11. Analogously, taking all M_q, $q \in \{1, \dots, \nu\}$, to be maximum weighted matchings gives the strongest bound $b_2\left(M_1, \dots, M_\nu\right)$.

It is worth remarking that the graph $G[S]$ would no longer be a model for the pairwise compatibility of the new selection of vertices after even a single such exchange operation. Therefore, in the case of both bounds b_1 and b_2, we cannot accumulate the additional value for non-disjoint conflicting pairs of vertices and still get a lower bound on z. That is the rationale behind searching for matchings, and attaining dual bounds for z via a well-solved combinatorial problem.

We can generalize the reasoning behind the relaxations yielding bounds b_1 and b_2 in Theorem 11 by considering matchings in the whole graph G, that is, not only in a proper induced subgraph. Since each k-stab contains at most one vertex from each edge in a matching, we can simply pick the k vertices of lowest weight among: (i) the cheapest vertex in each matched edge, and (ii) the remaining vertices not covered by the matching. So we have the following result.

Theorem 13. *Suppose that $\mathcal{P}(G, k) \cap \{0, 1\}^n \neq \emptyset$, so that z is well-defined in problem (5). Let $M \subset E$ be any matching in G. Define $c_e \stackrel{def}{=} w(v_i)$ for each edge $e = \{v_i, v_j\} \in M$, with $i < j$. Also define $c_u \stackrel{def}{=} w(v_u)$ for any vertex v_u not covered by the matching M. Then, the sum of the k lowest values among the $c(\cdot)$ is a lower bound on z. That is, given an order $c_1 \leq c_2 \cdots \leq c_{(n-|M|)}$ on $\{c_e\}_{e \in M} \cup \{c_u\}_{u \in V \backslash V_M}$, where V_M corresponds to the set of vertices covered by M, we have $z \geq \sum_{i=1}^{k} c_i$.*

The drawback involved in this statement is that, while the actual algorithm to compute the bounds referring to Theorem 11 is immediate (following Remark 12), the choice of a specific matching M yielding the strongest bound in

Theorem 13 is not clear. A first approach would be to evaluate different greedy constructions. Alternatively, a stronger bound should follow from computing minimum-weight matchings in G with cardinality at least $l \in \{1, \dots, k\}$, using the edge-weight function corresponding to $c(\cdot)$ in the latter theorem, that is, for each $\{v_i, v_j\} \in E(G)$, with $i < j$, define $c(\{v_i, v_j\}) \stackrel{\text{def}}{=} w(v_i)$. Note that we can find such a matching of least weight in polynomial time, as we note in Remark 10.

Finally, since every matching in a proper induced subgraph is also considered by Theorem 13, it follows that experimenting with the selection of the matching M yielding the latter bounds should never be weaker than bounds b_1 or b_2 from Theorem 11.

3.1. Application towards balanced branching trees

A fundamental component for the performance of branch-and-cut algorithms for the classical stable set problem is the balanced branching rule of Balas and Yu [1]; see also [25] and [22]. Its original motivation also applies to the fixed cardinality setting: avoiding unbalanced branch-and-bound trees when branching on a fractional variable x_v, since fixing $x_v = 1$ has the larger impact of implying $x_u = 0$ for each $u \in N(v)$, while fixing $x_v = 0$ has no impact on the neighbourhood.

The general branching scheme can be adapted to find minimum weight k-stabs with little effort. Suppose that, on a given node of the enumeration tree, $G' = (V', E')$ denotes the subgraph induced by vertices not fixed in this subproblem, and that $\bar{z}$ is the best primal bound available. Let $W \subseteq V'$ be such that we can determine *efficiently* that the minimum weight of a k-stab in the subgraph induced by W, denoted $z(W)$, is such that $z(W) \geq \bar{z}$. Note that, if $W = V'$, the subproblem is fathomed and the whole subtree rooted on this node can be pruned. Otherwise, if the search on this subtree is to eventually find that $z(V') < \bar{z}$, any bound-improving solution must intersect $V' \backslash W = \{v_1, \dots, v_p\}$. That is, we can partition the search space into the sets

$$V_i' = \{v_i\} \bigcup V' \backslash \left(N(v_i) \cup \{v_{i+1}, \dots, v_p\}\right)$$

for $1 \leq i \leq p$. The enumeration can therefore branch on p subproblems, each fixing $x_{v_i} = 1$, and fixing at 0 those variables corresponding to $N(v_i) \cup \{v_{i+1}, \dots, v_p\}$.

Now, there are different strategies to determine subgraph W. The standard one is to find a collection of cliques in G', *e.g.* with as many cliques as the currently available lower bound, when searching for maximum cardinality stable sets. For minimum-weight k-stabs, the natural idea would be to greedily find k cliques, such that the combined weight of the *cheapest* vertices in each exceed $\bar{z}$.

The combinatorial bounds that we introduce give an alternative approach tailored for optimizing over k-stabs. Using the weight function corresponding to $c(\cdot)$ in Theorem 13, we can determine candidate subgraphs W by inspecting, for each $l \in \{1, \dots, k\}$:

1. A minimum-weight matching in G' with cardinality l
2. A suitable choice of $k - l$ vertices not covered by the matching

We leave for future work the task of comparing those two strategies, whether theoretically or according to computational experience.

4. Particular cases solvable in polynomial time

A major research topic in combinatorial optimization is the study of particular cases of an NP-hard problem which admit a solution algorithm with polynomially-bounded worst case complexity. As indicated before, the rich theory on the classical stable set problem suggests that research in this direction is also promising. The work of Dabrowski et al. [11,12] in parameterized complexity parallels our contributions here. Instead of NP-completeness, their work builds on the W[1]-completeness of the classical stable set problem, cf. [10, Section 13.3], to give fixed-parameter tractable algorithms for an input restricted to some graph classes which extend that of graphs of bounded clique number.

We note that the recognition problem for all classes we discuss next can be solved in polynomial time, *i.e.* given an arbitrary graph G, there exists an algorithm with polynomially-bounded worst-case time complexity which decides if G belongs to that class of graphs. We refer the interested reader to the ISGCI encyclopaedia of graph classes [26]. Throughout this section, we denote by perfect the set of all perfect graphs (*i.e.* those graphs in which the chromatic number of every induced subgraph equals its clique number), and follow similar notation for any graph class.

Remark 14. Consider the unweighted problems corresponding to the classical stable set problem and the fixed cardinality version. If $G = (V, E)$ is such that the stability number $\alpha(G)$ can be found in polynomial time (*i.e.* the classical problem over G is in complexity class P), then we also have that deciding if there exists a k-stab in G is also in P. More precisely: for $k \in \{1, \dots, \alpha(G)\}$, the answer for the latter problem is `yes`; for $k > \alpha(G)$, the answer is `no`. Nevertheless, the same is not true regarding the weighted version of the problems. Even if a maximum-weight stable set in G can be found in polynomial time, it is not obvious how to find a k-stab in G of optimal weight, in general. In principle, there can be a number of optimal solutions for the classical problem, from which a k-stab might be retrieved or not; and, conversely, there might exist optimal-weight k-stabs in G which are not contained in any optimal solution to the classical problem.

Recall that a graph is *k-partite* (or *k-colourable*) if its vertices can be partitioned into k different stable sets. Now, a *complete k-partite* graph is a k-partite graph containing an edge between all pairs of vertices from different stable sets. A *complete multipartite graph* is complete k-partite for some k. The following result is rather straightforward.

Theorem 15. *If G is a complete multipartite graph, the problem of finding a minimum-weight k-stab in G is in* P.

Proof. Let $G = (V_1 \uplus \cdots \uplus V_c, E)$ be an arbitrary complete c-partite graph, so that each V_i induces a stable set in G, for $1 \le i \le c$. Clearly, no stable set in G contains vertices from more than one set in the partition. For each V_i such that $|V_i| \ge k$, then, we inspect the least-weight subset of cardinality k, *i.e.* we find $S_i \in \arg\min_{S \subset V_i, |S|=k} \sum_{v \in S} w(v)$. A minimum-weight k-stab in G is therefore one of minimum weight among all S_i. □

Complete multipartite graphs are a subclass of *cographs*, or *complement-reducible graphs*: those which can be constructed from isolated vertices by disjoint union and complementation operations alone. The class cograph is equivalent to that of P_4-free graphs, and a number of other characterizations are known [23, Sec. 7.9]. It follows from the definition that the class of graphs which are the complement of some complete multipartite graph corresponds to another subclass of cographs. These are known as *cluster graphs*. Thus G is a cluster graph if and only if G is the disjoint union of cliques; equivalently, G is a cluster graph if and only if it is P_3-free.

Theorem 16. *If G is cluster graph, the problem of finding a minimum-weight k-stab in G is in* P.

Proof. Let $G = \biguplus_{i=1}^{q} K_{n_i}$, where each K_{n_i} induces a clique on n_i vertices. Clearly, at most one vertex from each K_{n_i} can be part of a k-stab. If $k > q$, there cannot exist a k-stab in G. Now, assuming $k \le q$, the set of k-stabs in G corresponds to subsets of k vertices from different cliques each, since G is a *disjoint* union of the q cliques K_{n_i}, $1 \le i \le q$. In particular, we can restrict our attention to the set $S \overset{\text{def}}{=} \biguplus_{i=1}^{q} \left\{ v_i \in \arg\min_{v \in K_{n_i}} w(v) \right\}$ of least weight vertices in each clique, and a minimum-weight k-stab in G can be found by choosing k vertices of least weight in S. □

Note that cograph is contained in perfect. We consider next another subclass of perfect graphs, not contained in that of cographs. We say that G is a *split graph* if there exists a partition of its vertices into two sets, one of which induces a clique in G, the other inducing a stable set. A noteworthy result is that almost all *chordal graphs* are in split. (Recall that chordal or triangulated graphs are those in which every cycle of length at least 4 has a chord.) Precisely, the probability that a chordal graph chosen uniformly at random from the set of all chordal graphs on n vertices is split goes to 1 as $n \to \infty$ [3].

Theorem 17. *If G is a split graph, the problem of finding a minimum-weight k-stab in G is in* P.

Proof. Suppose $V(G) = \mathrm{C} \uplus \mathrm{I}$ is such that C induces a clique and I induces a stable set in G. Note first that at most one vertex from C belongs to any k-stab. Then, if $|\mathrm{I}| < k-1$, or if $|\mathrm{I}| = k-1$ and $C = \emptyset$, there is no k-stab in G. Suppose now that $|\mathrm{I}| \ge k-1$ and $C \ne \emptyset$. For each $v_i \in \mathrm{C}$ such that $|\mathrm{I} \backslash N(v_i)| \ge k-1$, let S_i denote a subset of $k-1$ vertices in $\mathrm{I} \backslash N(v_i)$ of least weight, that is, $S_i \in \arg\min_{S \subseteq \mathrm{I} \backslash N(v_i), |S|=k-1} \sum_{v \in S} w(v)$. Now,

$$\mathcal{S} \overset{\text{def}}{=} \{\{v_i\} \cup S_i : \text{ for each } v_i \in \mathrm{C} \text{ such that } |\mathrm{I} \backslash N(v_i)| \ge k-1\}$$

is thus an enumeration of all the k-stabs in G which include a vertex from C, and that those amount to at most $|\mathrm{C}|$ k-stabs. If $|\mathrm{I}| \ge k$, define also $S_0 \in \arg\min_{S \subseteq \mathrm{I}, |S|=k} \sum_{v \in S} w(v)$, *i.e.* a k-stab contained in I of least weight. Therefore, a minimum-weight k-stab in G can be found by inspection among those in $\mathcal{S}$ and S_0 thus defined. □

The class cograph ∩ split is equivalent to the class of *threshold graphs*. G is a threshold graph if it is possible to define a constant $t \in \mathbb{R}$ and a function $f : V(G) \to \mathbb{R}$ in such a way that $\{u, v\} \in E(G)$ if and only if $f(u) + f(v) \ge t$. An equivalent definition is that G is a threshold graph if it can be constructed from the empty graph by repeatedly adding either an isolated vertex or a universal vertex. It is therefore a consequence of Theorem 17 that our problem of interest is well-solved over threshold graphs.

Corollary 18. *If G is a threshold graph, the problem of finding a minimum-weight k-stab in G is in* P.

From our results, we have algorithms with polynomial worst-case time complexity to find minimum-weight k-stabs in some representative subclasses of cograph: complete graphs, complete multipartite graphs, cluster graphs and threshold graphs. A natural question that we pose as a conjecture, then, is whether the positive results could be generalized to the whole class of cographs.

Conjecture 2. *Given an arbitrary cograph G, weights $w : V(G) \to \mathbb{R}_+$ and $k \in \mathbb{Z}_+$, the problem of finding a minimum-weight k-stab in G is in* P.

To conclude, we mention that the problem is also well-solved over a class of graphs which is not contained in perfect. An equivalent result was already shown by Buchanan [8], but we include its simple proof for the sake of completeness. The *line graph* $L(G)$ of a given graph $G = (V, E)$ is the intersection graph of the edges of G, that is, the graph containing a vertex for each element in E, and where two vertices are connected if and only if the corresponding edges in G share an endpoint.

Theorem 19. *If H is a line graph, the problem of finding a minimum-weight k-stab in H is in* P.

Proof. Suppose that $H = L(G)$ is an arbitrary line graph, with G being an underlying root graph. Note that G is uniquely defined, provided $H \notin \{K_3, K_{1,3}\}$ (in which cases the result would follow immediately), as proved by Whitney [30] cf. [23, Example 1.4]. Moreover, the original graph G can be determined from H in linear time [19]. Now, $S \subset V(H)$, induces a stable set in H if and only if $S \subset E(G)$ is a matching in G, and the bijection obviously preserves cardinality and weight. Therefore, a minimum-weight k-stab in H corresponds to a minimum-weight matching of cardinality k in G. The result, then, follows from the fact that finding such a matching is a well-solved problem, as described in Remark 10. □

CRediT authorship contribution statement

Phillippe Samer: Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Writing - original draft, Writing - review & editing. **Dag Haugland:** Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Writing - original draft, Writing - review & editing.

Acknowledgement

This research is partly supported by the Research Council of Norway through the research project 249994 CLASSIS. All authors approved the version of the manuscript to be published.

References

[1] E. Balas, C.S. Yu, Finding a maximum clique in an arbitrary graph, SIAM J. Comput. 15 (4) (1986) 1054–1068, http://dx.doi.org/10.1137/0215075.
[2] A. Bazzi, S. Fiorini, S. Pokutta, O. Svensson, No small linear program approximates vertex cover within a factor 2 - ϵ, Math. Oper. Res. 44 (1) (2019) 147–172, http://dx.doi.org/10.1287/moor.2017.0918.
[3] E.A. Bender, L.B. Richmond, N.C. Wormald, Almost all chordal graphs split, J. Aust. Math. Soc. Ser. A. Pure Math. Stat. 38 (2) (1985) 214–221, http://dx.doi.org/10.1017/S1446788700023077.
[4] D. Bertsimas, J.N. Tsitsiklis, Introduction to Linear Optimization, Athena Scientific Belmont, MA, 1997.
[5] D. Bertsimas, R. Weismantel, Optimization Over Integers, Dynamic Ideas, Belmont, MA, USA, 2005.
[6] Q. Botton, Survivable Network Design with Quality of Service Constraints: Extended Formulations and Benders Decomposition, Ph.D. thesis, Université Catholique de Louvain, Louvain-la-Neuve, 2010, URL http://hdl.handle.net/2078.1/33300.
[7] M. Bruglieri, M. Ehrgott, H.W. Hamacher, F. Maffioli, An annotated bibliography of combinatorial optimization problems with fixed cardinality constraints, Discrete Appl. Math. (ISSN: 0166-218X) 154 (9) (2006) 1344–1357, http://dx.doi.org/10.1016/j.dam.2005.05.036.
[8] A. Buchanan, Extended formulations for vertex cover, Oper. Res. Lett. (ISSN: 0167-6377) 44 (3) (2016) 374–378, http://dx.doi.org/10.1016/j.orl.2016.03.008.
[9] A. Buchanan, S. Butenko, Tight extended formulations for independent set, 2014, URL http://www.optimization-online.org/DB_FILE/2014/09/4540.pdf, Unpublished manuscript.
[10] M. Cygan, F.V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, S. Saurabh, Parameterized Algorithms, Springer International Publishing, 2015, http://dx.doi.org/10.1007/978-3-319-21275-3.
[11] K. Dabrowski, V. Lozin, H. Müller, D. Rautenbach, Parameterized algorithms for the independent set problem in some hereditary graph classes, in: C.S. Iliopoulos, W.F. Smyth (Eds.), International Workshop on Combinatorial Algorithms (IWOCA) 2010, in: Lecture Notes in Computer Science, vol. 6460, Springer, Berlin Heidelberg, ISBN: 978-3-642-19222-7, 2011, pp. 1–9, http://dx.doi.org/10.1007/978-3-642-19222-7_1.
[12] K. Dabrowski, V. Lozin, H. Müller, D. Rautenbach, Parameterized complexity of the weighted independent set problem beyond graphs of bounded clique number, J. Discrete Algorithms (ISSN: 1570-8667) 14 (2012) 207–213, http://dx.doi.org/10.1016/j.jda.2011.12.012.
[13] A. Darmann, U. Pferschy, J. Schauer, Determining a minimum spanning tree with disjunctive constraints, in: F. Rossi, A. Tsoukias (Eds.), Algorithmic Decision Theory, in: Lecture Notes in Computer Science, vol. 5783, Springer, Berlin Heidelberg, 2009, pp. 414–423, http://dx.doi.org/10.1007/978-3-642-04428-1_36.
[14] A. Darmann, U. Pferschy, J. Schauer, G.J. Woeginger, Paths, trees and matchings under disjunctive constraints, Discrete Appl. Math. 159 (16) (2011) 1726–1735, http://dx.doi.org/10.1016/j.dam.2010.12.016.
[15] J. Gajarský, P. Hliněný, H.R. Tiwary, Parameterized extension complexity of independent set and related problems, Discrete Appl. Math. 248 (2018) 56–67, http://dx.doi.org/10.1016/j.dam.2017.04.042.
[16] M. Guignard, Lagrangean relaxation, Top 11 (2) (2003) 151–200, http://dx.doi.org/10.1007/BF02579036.
[17] S. Hanafi, R. Todosijević, Mathematical programming based heuristics for the 0–1 MIP: a survey, J. Heuristics 23 (4) (2017) 165–206, http://dx.doi.org/10.1007/s10732-017-9336-y.
[18] J. Janssen, K. Kilakos, Bounded stable sets: Polytopes and colorings, SIAM J. Discrete Math. (ISSN: 0895-4801) 12 (2) (1999) 262–275, http://dx.doi.org/10.1137/S089548019630978X.
[19] P.G.H. Lehot, An optimal algorithm to detect a line graph and output its root graph, J. Assoc. Comput. Mach. (ISSN: 0004-5411) 21 (1974) 569–575, http://dx.doi.org/10.1145/321850.321853.
[20] A. Lucena, Non delayed relax-and-cut algorithms, Ann. Oper. Res. 140 (1) (2005) 375–410, http://dx.doi.org/10.1007/s10479-005-3977-1.
[21] C. Mannino, G. Oriolo, F. Ricci, S. Chandran, The stable set problem and the thinness of a graph, Oper. Res. Lett. 35 (1) (2007) 1–9, http://dx.doi.org/10.1016/j.orl.2006.01.009.

[22] C. Mannino, A. Sassano, Edge projection and the maximum cardinality stable set problem, in: D.S. Johnson, M. Trick (Eds.), Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge, October 11-13, 1993, in: DIMACS series in discrete mathematics and theoretical computer science, vol. 26, American Mathematical Society, 1996, pp. 205–219.

[23] T. McKee, F. McMorris, Topics in Intersection Graph Theory, Society for Industrial and Applied Mathematics, 1999, http://dx.doi.org/10.1137/1.9780898719802.

[24] G.L. Nemhauser, L.E. Trotter, Properties of vertex packing and independence system polyhedra, Math. Program. 6 (1) (1974) 48–61, http://dx.doi.org/10.1007/BF01580222.

[25] S. Rebennack, G. Reinelt, P.M. Pardalos, A tutorial on branch and cut algorithms for the maximum stable set problem, Int. Trans. Oper. Res. 19 (1–2) (2012) 161–199, http://dx.doi.org/10.1111/j.1475-3995.2011.00805.x.

[26] H.N. de Ridder, et al., Information system on graph classes and their inclusions (ISGCI), 2001–2019, http://www.graphclasses.org.

[27] P. Samer, E. Cavalcante, S. Urrutia, J. Oppen, The matching relaxation for a class of generalized set partitioning problems, Discrete Appl. Math. (ISSN: 0166-218X) 253 (2019) 153–166, http://dx.doi.org/10.1016/j.dam.2018.05.033.

[28] P. Samer, D. Haugland, The unsuitable neighbourhood inequalities for the fixed cardinality stable set polytope, in: Claudio Gentile, Giuseppe Stecca, Paolo Ventura (Eds.), Graphs and Combinatorial Optimization: from Theory to Applications – CTW2020 Proceedings, in: AIRO Springer Series, 5, Springer International Publishing, 2021, https://doi.org/10.1007/978-3-030-63072-0.

[29] A. Schrijver, Combinatorial Optimization: Polyhedra and Efficiency, in: Algorithms and Combinatorics., vol. 24, Springer-Verlag, Berlin Heidelberg, 2003.

[30] H. Whitney, Congruent graphs and the connectivity of graphs, Amer. J. Math. 54 (1) (1932) 150–168, http://dx.doi.org/10.2307/2371086.