INSTITUTO DE COMPUTAÇÃO
UNIVERSIDADE ESTADUAL DE CAMPINAS

# $K_r$-packing of $P_4$-tidy graphs

*V. Pedrotti* *C. P. de Mello*

Technical Report - IC-09-19 - Relatório Técnico

May - 2009 - Maio

# $K_r$-packing of $P_4$-tidy graphs

Vagner Pedrotti* Célia Picinin de Mello†

**Abstract**

The $K_r$-packing problem asks for the maximum number of pairwise vertex-disjoint cliques of size $r$ in a graph, for a fixed integer $r \geq 2$. This problem is NP-hard for general graphs when $r \geq 3$, and even when restricted to chordal graphs. However, Guruswami et al. proposed a polynomial-time algorithm for cographs (when $r$ is fixed). In this work we extended this algorithm to $P_4$-tidy graphs, keeping the same time complexity.

## 1 Introduction

The *$K_r$-packing problem* consists of finding the maximum number of pairwise vertex-disjoint cliques of size $r$ in a graph, for a fixed integer $r \geq 2$. Note that, for $r = 2$ the problem is exactly the maximum matching problem, which has a well-known polynomial time algorithm. However, the $K_r$-packing problem becomes NP-hard for general graphs and even for line, chordal, planar, and total graphs (when $r \geq 3$) [3]. Moreover, the problem remains NP-hard for subclasses of chordal and planar graphs, such that split graphs (when $r \geq 4$) [3] and planar cubic graphs [1]. Recently, Guruswami et al. [3] proposed a polynomial time algorithm for cographs (when $r$ is fixed), which we extend for $P_4$-tidy graphs.

A related problem is the *$K_r$-factor problem*, which asks whether a graph has a partition of its vertices into cliques of size $r$. Note that, for $r = 2$ this problem is the perfect matching problem, and for $r \geq 3$ it is NP-complete for graphs with a maximum clique of $r$ vertices [5], but polynomial for split graphs [3].

In this work, $G$ denotes a simple, finite, and undirected graph with vertex set $V(G)$ and edge set $E(G)$. Let $H \subseteq V(G)$, we denote by $G[H]$ the induced subgraph of $G$ whose vertex set is $H$.

We denote an induced path of $n$ vertices by $P_n$, an induced cycle of $n$ vertices by $C_n$, and the complement of a graph $G$ by $\overline{G}$. A *clique* is a subset $K \subseteq V(G)$ (not necessarily maximal) such that for any $\{u, v\} \in K$, $u$ and $v$ are adjacent in $G$. A *stable set* is a subset $S \subseteq V(G)$ such that for any $\{u, v\} \subseteq S$, $u$ and $v$ are non-adjacent in $G$. We denote by $K_n$ ($S_n$) a clique (stable set) with $n$ vertices.

A subset $M$ of $V(G)$ is called a *module* of $G$ if do not exist $\{a, b\} \subseteq M$ and $c \in (V(G) \setminus M)$ such that $\{a, c\} \in E(G)$ and $\{b, c\} \notin E(G)$. A module $M$ of $G$ is *strong* if does not exist another module $N$ of $G$ such that $N \setminus M \neq \varnothing$, $M \setminus N \neq \varnothing$, and $N \cap M \neq \varnothing$.

*Partially supported by FAPESP, grant 2007/58519-0.

†Partially supported by CNPq, grants 300934/2006-8 and 482521/2007-4.

The *modular decomposition tree* (MDT) of a graph $G$ has one node for each strong module of $G$. The parent of a node, associated with module $M$, is the node that corresponds to the strong module with minimum cardinality which properly contains $M$. Hence, the MDT represents inclusions of strong modules, from isolated vertices (leaves) to the module $V(G)$ (the root). If $N$ is a non-leaf node of the MDT and $M$ is the corresponding module in $G$, then $N$ is called *serial* (*parallel*), if $\overline{G[M]}$ ($G[M]$) is not connected. Otherwise, the node $N$ is called *neighborhood*. A linear-time algorithm that produces the MDT of any graph is given in [6].

A *cograph* is a graph characterized by the absence of neighbourhood nodes in its MDT. A graph is $P_4$*-sparse* [4] if the subgraph induced by each module corresponding to a neighborhood node in its MDT is isomorphic to a spider. A graph $G$ is a *spider* if $V(G)$ can be partitioned into three sets $K$, $S$, and $H$, such that:

- $|K| = |S| \geq 2$, but $H$ can be empty;
- $K$ is a clique and $S$ is a stable set;
- $\{i, j\} \in E(G), \forall i \in K, \forall j \in H$;
- $\{i, j\} \notin E(G), \forall i \in S, \forall j \in H$; and
- There is a bijection $b : K \to S$ such that, $\forall u \in S, \forall v \in K, \{u, v\} \in E(G)$ if, and only if, $b(v) = u$ (*thin spider*) or $b(v) \neq u$ (*thick spider*).

Note that a spider graph $G$ has at least four vertices and, if it is a $P_4$-sparse graph, then $G[H]$ is also a $P_4$-sparse graph. We say that a spider $G$ has a partition $(K, S, H)$.

A *pseudo-spider* is a spider or a spider with one vertex $v$ in $S$ or in $K$ split into two new vertices, which are adjacent to the neighbors of $v$ and may either be adjacent to each other or not. A $P_4$*-tidy* graph [2] is a graph in which the subgraph induced by the module associated with each neighborhood node in its MDT is isomorphic to a $P_5$, a $\overline{P_5}$, a $C_5$, or a pseudo-spider.

## 2 $K_r$-packing of some special graphs

In the next subsections, we analyze the solution of the $K_r$-packing problem for some kinds of graphs, which are union graphs, join graphs, spiders, pseudo-spiders, and some special graphs.

### 2.1 Union and join graphs

In this section we recall, for the reader's convenience, the results about $K_r$-packing of cographs given by Guruswami et al. [3], using the a slightly different notation, given bellow. They proposed an algorithm to compute in polynomial time a $K_r$-packing of a cograph, using a dynamic programming technique.

To describe these results, some definitions are necessary. A partition $P$ of $V(G)$ is a $(n_1, n_2, \ldots, n_r)$*-partition* of $G$, for $n_i \in \mathcal{Z}^+, 1 \leq i \leq r$ and $r \geq 2$, if each part of $P$ is a clique of $G$ and there are exactly $n_i$ cliques of size $i$ in $P$. This obviously implies that all cliques in $P$ are vertex-disjoint and $|V(G)| = \sum_{i=1}^{r} in_i$. The $K_r$-packing problem asks for the maximum value of $n_r$ such that $G$ has a $(x, 0, 0, \ldots, n_r)$-partition of $G$, for $x = |V(G)| - rn_r$.

Now, consider the function $f(G, n_3, n_4, \ldots, n_r) = \max\{n_2 : G$ has a $(x,\ n_2,\ n_3,\ n_4, \ldots,$ $n_r)$-partition for $x = |V(G)| - \sum_{i=2}^{r} in_i\}$. This function is undefined when there is no $(x, n_2, n_3, n_4, \ldots, n_r)$-partition of $G$ for all values of $x$ and $n_2$. Note that, if we compute $f(G,\ 0,\ 0, \ldots,\ n_r)$ for all values of $n_r$ between 0 and $|V(G)|/r$, we solve the $K_r$-packing problem for $G$.

We also need two graph operations, the first produces a *union graph* $G = G' \cup G''$ defined by $V(G) = V(G') \cup V(G'')$ and $E(G) = E(G') \cup E(G'')$. The second operation produces a *join graph* $G = G' + G''$ and it is defined by $V(G) = V(G') \cup V(G'')$ and $E(G) = E(G') \cup E(G'') \cup \{\{v, u\} : v \in V(G'), u \in V(G'')\}$.

If $G = G_1 \cup G_2 \cup \ldots \cup G_k$, to compute $f(G, n_3, n_4, \ldots, n_r)$, Guruswami et al. apply repeatedly an algorithm that computes $f$ on a graph $G' \cup G''$. This algorithm returns the maximum of $f(G', n'_3, n'_4, \ldots, n'_r) + f(G'', n''_3, n''_4, \ldots, n''_r)$, for all integers $n'_i \geq 0$ and $n''_i \geq 0$ which satisfy $n_i = n'_i + n''_i$, for $3 \leq i \leq r$.

We define a *configuration* $F$ as a sequence of non-negative integers, which have a particular notation: $F_{i,j}$, $F'_i$, and $F''_i$, for $1 \leq i \leq r$ and $0 \leq j \leq i$. A $(n_1, n_2, \ldots, n_r)$-partition $P$ of $G' + G''$ has configuration $F$ relative to $G'$ if $F_{i,j} = |\{C : C \in P, |C| = i, |C \cap V(G')| = j\}|$, $F'_i = \sum_{k=i}^{r} F_{k,i}$, and $F''_i = \sum_{k=i}^{r} F_{k,k-i}$, for all integers $1 \leq i \leq r$ and $0 \leq j \leq i$.

If $G = G_1 + G_2 + \ldots + G_k$, Guruswami et al. apply an algorithm that computes $f(H, n_3, n_4, \ldots, n_r)$ on a graph $H = G' + G''$. Let $\mathcal{F} = \{F : \exists$ a $(n_1, n_2, \ldots, n_r)$-partition $P$ of $H$ with configuration $F$ relative to $G'\}$. This algorithm returns the maximum of $F_{2,0} + F_{2,1} + F_{2,2}$, for all $F \in \mathcal{F}$. A configuration $F$ is in $\mathcal{F}$ if, and only if, it satisfies:

- $n_i = \sum_{j=0}^{i} F_{i,j}$ and $F_{i,j} \geq 0$, for $1 \leq i \leq r$ and $0 \leq j \leq i$;
- $F'_2 \leq f(G', F'_3, \ldots, F'_r)$ and $F''_2 \leq f(G'', F''_3, \ldots, F''_r)$; and
- $|V(G')| = \sum_{j=1}^{r} jF'_j$ and $|V(G'')| = \sum_{j=1}^{r} jF''_j$.

Once their aim is to compute $f$, it is not needed to analyze every possible configuration in $\mathcal{F}$. Lemma 4.2 from [3] (rewriten below, adapted to our notation) is used to reduce the number of analyzed configurations by imposing an additional condition: $\sum_{i=2}^{r} F_{i,j} = 0$ or $\sum_{i=2}^{r} F_{i,0} = 0$.

**Lemma 2.1** (Lemma 4.2 from [3]). *If $P$ is a $(n_1, n_2, \ldots, n_r)$-partition of $G = G' + G''$, then there exists another $(n_1, n_2, \ldots, n_r)$-partition $P'$ of $G$ such that $P'$ does not contain $C'$ and $C''$ such that $C' \subseteq V(G')$ and $C'' \subseteq V(G'')$.*

## 2.2 Spiders

Let $G$ be a spider with partition $(K, S, H)$ and $P$ be a $(n_1, n_2, \ldots, n_r)$-partition of $G$. Then, each $C \in P$ either has one vertex in $S$ or is a subset of $K \cup H$. We now extend the definition of configuration given in Section 2.1 adding the integers $F^S_i$, for $1 \leq i \leq r$. Since $G[K \cup H] = G[K] + G[H]$, we say that $P$ has configuration $F$ if $F_{i,j} = |\{C : C \in P, C \subseteq$

$K \cup H, |C| = i, |C \cap K| = j\}|$, $F_i^S = |\{C : C \in P, |C| = i, |C \cap S| = 1\}|$, $F'_i = \sum_{k=i}^{r} F_{k,i}$, and $F''_i = \sum_{k=i}^{r} F_{k,k-i}$, for all integers $1 \leq i \leq r$ and $0 \leq j \leq i$.

We now describe conditions to pack cliques in any spider.

**Lemma 2.2.** *A spider $G$ has a $(n_1, n_2, \ldots, n_r)$-partition with configuration $F$ if, and only if, the configuration $F$ satisfies:*

1. $F''_2 \leq f(G[H], F''_3, \ldots, F''_r)$;
2. $|S| = \sum_{i=1}^{r} F_i^S$, $|K| = \sum_{i=2}^{r}(i-1)F_i^S + \sum_{i=1}^{r} iF'_i$, *and* $|H| = \sum_{i=1}^{r} iF''_i$;
3. $F_i^S = 0$, *for* $i \geq 3$ $(i \geq |K|+1)$ *when $G$ is a thin (thick) spider; and*
4. $n_i = F_i^S + \sum_{j=0}^{i} F_{i,j}$, *for* $1 \leq i \leq r$.

*Proof.* Let $G$ be a spider and $P$ be a $(n_1, n_2, \ldots, n_r)$-partition of $G$ with configuration $F$. The first condition is true because intersections of elements of $P$ with $H$ give a partition of $H$ into cliques with $F''_i$ cliques of size $i$, for $1 \leq i \leq r$. The second condition must be satisfied because every vertex of each part of the spider must be in exactly one clique of $P$. The third one follows from the fact that no clique can use a vertex in $S$ and all vertices of $K$ for a thick spider or a vertex of $S$ and two or more of $K$ for a thin spider. Finally, the last condition is true because each clique in $P$ must either contain one vertex of $S$ or be contained in $K \cup H$.

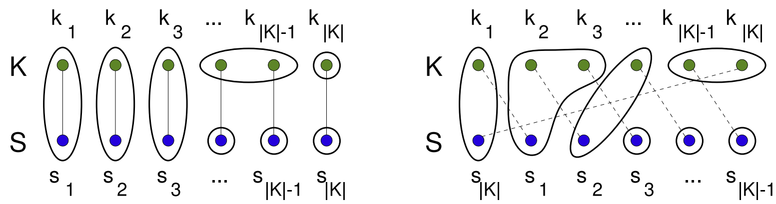
Given a configuration $F$ that satisfy the Lemma's requirements, a partition of $V(G)$ into cliques with that configuration can be produced. To do this, label each vertex of $K$ by $k_i$ for $1 \leq i \leq |K|$ and order the vertices in a sequence $O_K = (k_1, k_2, \ldots, k_{|K|})$. Now, if $G$ is a thin spider, label as $s_i$ the vertex in $S$ which is adjacent to $k_i$ and make the sequence $O_S = (s_1, s_2, \ldots, s_{|S|})$. Otherwise, label as $s_i$ the vertex in $S$ which is non-adjacent to $k_i$ and make the sequence $O_S = (s_{|S|}, s_1, s_2, \ldots, s_{|S|-1})$.

We now partition $K \cup S$ producing a set of cliques $P'$. First, for each $2 \leq i \leq r$, we create $F_i^S$ cliques of size $i$ with one vertex in $S$ each. We remove the first vertex of $O_S$ and the first $i-1$ vertices of $O_K$ and use them to produce each clique. If $G$ is a thick spider, each set uses one vertex of $S$ and at least one vertex of $K$ (but not all), so by the ordering of $O_S$ and $O_K$ and Condition 3, the vertices $s_i$ and $k_i$ are never packed in the same set and, then, each set is a clique. If $G$ is a thin spider, each set contains adjacent vertices $s_i$ and $k_i$, by the ordering of $O_S$ and $O_K$ and Condition 3.

Now, consider the integers $F'_j$, for $1 \leq j \leq r$. We partition the remaining vertices of $O_K$ into $F'_j$ sets of size $j$, for each $j$. Each set is a clique by definition of a spider. Finally, take the $F_1^S$ vertices remaining in $O_S$ and use each one as a unitary clique. Thus, each vertex of $K \cup S$ was in some clique, by Condition 2.

A way to see what happens is to take vertices of $K$ and $S$ as two queues and apply the above procedure. Examples of the procedure for thin and thick spiders are given in Figure 1.

Finally, we produce the partition $P$ using the previous cliques in $P'$ and cliques in a $(n''_1, n''_2, \ldots, n''_r)$-partition $P''$ of $H$. The following procedure is executed once for each $F_{i,j} > 0$ with $0 < j < i \leq r$. Get $F_{i,j}$ pairs formed by one clique of size $i$ in $P'$ that is contained in $K$ and one clique of size $i-j$ in $P''$. Now, make the union of cliques in each pair to produce $F_{i,j}$ cliques of size $i$ in $G$. By definition of $F'_i$ and $F''_i$ and by Condition 1 there are enough cliques available in $P'$ and $P''$. These new cliques, together with non-used cliques of $P'$ and $P''$ give a $(n_1, n_2, \ldots, n_r)$-partition of $G$ by Condition 4. □




(a) For a thin spider. (b) For a thick spider.

Figure 1: An example of packing cliques in $K \cup S$.

Lemma 2.2 gives necessary and sufficient conditions to partition vertices of a spider into cliques. This makes possible to rearrange cliques in any partition so that some properties are satisfied, as in the next statement.

**Lemma 2.3.** *If a spider $G$ has a $(n_1, n_2, \ldots, n_r)$-partition , then there is another $(n_1, n_2, \ldots, n_r)$-partition $P'$ of $G$ in which every $K_2$ either is contained in $H$ or has a vertex in $S$.*

*Proof.* Let $P$ be a $(n_1, n_2, \ldots, n_r)$-partition of a spider $G$ with partition $(K, S, H)$. Then, the configuration $F$ of $P$ satisfies conditions of Lemma 2.2.

Since any clique contained in $K \cup S$ contains at most one vertex in $S$, then $F_1^S \geq 2F_{2,2} + F_{2,1}$. If $F_{2,2} > 0$, we change the integers, so that $F_2^S \leftarrow F_2^S + F_{2,2}$, $F_1^S \leftarrow F_1^S - F_{2,2}$, $F_{1,1} \leftarrow F_{1,1} + F_{2,2}$, and $F_{2,2} \leftarrow 0$. Moreover, if $F_{2,1} > 0$, we also apply the following changes: $F_2^S \leftarrow F_2^S + F_{2,1}$, $F_1^S \leftarrow F_1^S - F_{2,1}$, $F_{1,0} \leftarrow F_{1,0} + F_{2,1}$, and $F_{2,1} \leftarrow 0$. These transformations change only the integers $F_1^S$, $F_2^S$, $F_{1,0}$, $F_{1,1}$, $F_{2,1}$, and $F_{2,2}$, while they keep $F''_i$ unchanged, for $1 \leq i \leq r$. So, the new configuration satisfy the first condition of Lemma 2.2, as well as the others, which can be verified by inspection.

Therefore, there exists a $(n_1, n_2, \ldots, n_r)$-partition $P'$ of $G$, such that no $K_2$ in $P'$ is contained in $K$ or has a vertex in $K$ and another in $H$. □

## 2.3 Pseudo-spiders

We consider pseudo-spiders which are not spiders. From a spider, we can produce eight distinct kinds of pseudo-spiders (counting thick and thin ones separately). The two new vertices produced splitting a vertex $v$ of the spider are considered to be contained in the same partition ($S$ or $K$) as the original vertex $v$.

We provide conditions for partitioning the vertices of all kinds of pseudo-spiders into cliques in lemmas 2.4, 2.5, and 2.7. Two properties given in lemmas 2.6 and 2.8 may be used to constraint the search for a particular partition, improving the time complexity of an algorithm on the last three cases.

In these lemmas, the configuration of a $(n_1, n_2, \ldots, n_r)$-partition of spiders from Section 2.2 is extended to pseudo-spiders. Also, we denote by $(K, S, H)$ the partition of a pseudo-spider $G$.

**Lemma 2.4.** *Let $G'$ be a spider with partition $(K, S', H)$ and $G$ be a pseudo-spider produced by splitting a vertex $v \in S'$ of $G'$ into two non-adjacent vertices $v'$ and $v''$. Then, $f(G, n_3, n_4, \ldots, n_r) = f(G', n_3, n_4, \ldots, n_r)$ or both are undefined.*

*Proof.* If $G'$ is a thin spider, then at least one of $v'$ and $v''$ is in a unitary set in any partition of $V(G)$ into cliques, since $v'$ and $v''$ have exactly the same neighbor. Thus, any $(n_1, n_2, \ldots, n_r)$-partition $P$ of $G$ may be turned into a $(n_1, n_2, \ldots, n_r)$-partition of $G'$ by removing the unitary set $\{v'\}$ or $\{v''\}$, as well as we can turn a $(n_1, n_2, \ldots, n_r)$-partition$P'$ of $G'$ into a $(n_1, n_2, \ldots, n_r)$-partition of $G$ by relabeling $v$ to $v'$ in $P'$ and adding the clique $\{v''\}$. These transformations preserve the value of $n_i$, for $2 \leq i \leq r$, and so, preserve the value of $f$.

If $G'$ is a thick spider and $P$ is a $(n_1, n_2, \ldots, n_r)$-partition of $G$, we can always make another $(n_1, n_2, \ldots, n_r)$-partition $P'$ of $G$, in which $v'$ or $v''$ is in a unitary set to use the previous argument. If neither $\{v'\} \in P$ nor $\{v''\} \in P$, then let $\{C', C''\} \subseteq P$ such that $v' \in C'$ and $v'' \in C''$. Since $|S| > |K|$, there exists a vertex $u \in S$ such that $\{u\} \in P$. By definition of a thick spider, at least one of $C' \setminus \{v'\}$ or $C'' \setminus \{v''\}$ are all neighbors of $u$. Without loss of generality, consider $C'$ such that $C' \setminus \{v'\}$ are neighbors of $u$ (otherwise, swap $C'$ with $C''$ and $v'$ with $v''$) and let $P' = (P \setminus \{C', \{u\}\}) \cup \{\{v'\}, \{u\} \cup (C' \setminus \{v'\})\}$. $\square$

Now, we consider a pseudo-spider $G'$ produced by splitting a vertex $v$ from the partition $S$ of a spider $G$ into two adjacent vertices $v'$ and $v''$. In this case, both vertices $v'$ and $v''$ can be used together with vertices from $K$ to form a clique. If $P$ is a $(n_1, n_2, \ldots, n_r)$-partition of $G$, we extend the definition of configuration by adding the non-negative integers $F_i^e$, for $2 \leq i \leq r$. We also say $P$ has configuration $F$ if the integers $F_{i,j}$, $F_i^S$, and derivatives are computed as they are for spiders and $F_i^e$ is 1 if $P$ contains a clique of size $i$ which includes both $v'$ and $v''$, and is 0, otherwise.

**Lemma 2.5.** *Let $G'$ be a spider with partition $(K, S', H)$ and $G$ be a pseudo-spider produced by splitting a vertex $v \in S'$ of $G'$ into two adjacent vertices $v'$ and $v''$. So, $G$ has a $(n_1, n_2, \ldots, n_r)$-partition with configuration $F$ if, and only if, the configuration $F$ satisfies:*

1. $F_2'' \leq f(G[H], F_3'', \ldots, F_r'')$;
2. $|S| = \sum_{i=1}^{r} F_i^S + 2\sum_{i=2}^{r} F_i^e$, $|K| = \sum_{i=2}^{r}((i-1)F_i^S + (i-2)F_i^e) + \sum_{i=1}^{r} iF_i'$, *and* $|H| = \sum_{i=1}^{r} iF_i''$;
3. $F_i^S = 0$ *if* $i \geq 3$ $(i \geq |K|+1)$ *and $G$ is a thin (thick) spider;*
4. $n_i = F_i^S + F_i^e + \sum_{j=0}^{i} F_{i,j}$, *for* $1 \leq i \leq r$;
5. $\sum_{i=2}^{r} F_i^e \leq 1$; *and*
6. $F_i^e = 0$ *if* $i \geq 4$ $(i \geq |K|+2)$ *and $G$ is a thin (thick) spider.*

*Proof.* Let $G'$ be a spider and $G$ obtained from $G'$ by splitting a vertex of $S'$ into two adjacent vertices $v'$ and $v''$. The Lemma's conditions are those found on Lemma 2.2, adapted to consider the integers $F_i^e$. It is possible to verify that the configuration of any $(n_1, n_2, \ldots, n_r)$-partition of $G$ must satisfy these conditions with similar arguments of Lemma 2.2. Note that conditions 5 and 6 enforce the proper use of vertices $v'$ and $v''$ to produce cliques and conditions 2 and 4 take into account a clique with both $v'$ and $v''$ when $F_i^e = 1$ for some $i$.

To produce a $(n_1, n_2, \ldots, n_r)$-partition $P$ for $G$ with a configuration $F$ satisfying the requirements of this lemma, we create a new configuration $\overline{F} = F$. Then, we proceed to one of the following cases, that will change $\overline{F}$.

If $F_{j+1}^e = 1$ for some $2 \leq j \leq r$, we set $\overline{F}_j^S = F_j^S + 1$, so that the new configuration satisfies the requirements of Lemma 2.2 for $G'$ (by inspection, ignoring the integers $\overline{F}_i^e$) and, then, we have a $(n_1, n_2, \ldots, n_j + 1, n_{j+1} - 1, \ldots, n_r)$-partition $P'$ of $G'$ (note that $n_{j+1} \geq 1$, since $F_{j+1}^e = 1$). Then we select any $C \in P$, $|C| = j$, such that $C \cap S = \{u\}$ ($C$ must exist, since $\overline{F}_j^S \geq 1$) and relabel vertices of $G'$ such that $u = v$. This can be done by swapping $v$ with $u$ and their counterparts $b^{-1}(v)$ and $b^{-1}(u)$ in $K$ (see definition of spider on Section 2.2). Finally, we make $P = (P' \setminus \{C\}) \cup \{(C \setminus \{v\}) \cup \{v', v''\}\}$.

If $F_i^e = 0$ for all $2 \leq i \leq r$, we set $\overline{F}_1^S = F_1^S - 1$ ($F_1^S \geq 1$ since $|S| > |K|$ and no two vertices of $S$ would be in the same clique). The new configuration satisfies the conditions of Lemma 2.2 for $G'$, so it has a $(n_1 - 1, n_2, \ldots, n_r)$-partition $P'$. Let $C \in P'$ such that $v \in C$ and make $P = (P' \setminus \{C\}) \cup \{(C \setminus \{v\}) \cup \{v'\}, \{v''\}\}$. $\square$

**Lemma 2.6.** *Let $G'$ be a spider with partition $(K, S', H)$ and $G$ be a pseudo-spider produced by splitting a vertex $v \in S'$ into two adjacent vertices $v'$ and $v''$. If $G$ has a $(n_1, n_2, \ldots, n_r)$-partition $P$ with $n_2$ maximum (there is no $(n_1 - 2, n_2 + 1, \ldots, n_r)$-partition of $G$), then there exists another $(n_1, n_2, \ldots, n_r)$-partition $P'$ of $G$ such that vertices $v'$ and $v''$ are both in the same clique.*

*Proof.* Suppose $\{C', C''\} \subseteq P$ such that $v' \in C'$ and $v'' \in C''$. If $|C'| = |C''| = 1$, then $n_2$ is not maximum, since $(P \setminus \{C', C''\}) \cup \{\{v', v''\}\}$ is a $(n_1 - 2, n_2 + 1, \ldots, n_r)$-partition of $G$. Thus, without loss of generality, let $|C'| > 1$ and $u \in (C' \setminus \{v'\})$. Since $u \in K$, we produce $P'$ as $(P \setminus \{C', C''\}) \cup \{(C' \cup \{v''\}) \setminus \{u\}, (C'' \cup \{u\}) \setminus \{v''\}\}$. $\square$

Lemma 2.6 can be used on packing algorithms for this kind of pseudo-spider, adding the constraint $\sum_{i=2}^{r} F_i^e = 1$ and, therefore, reducing the computed possibilities.

**Lemma 2.7.** *Let $G'$ be a spider with partition $(K', S, H)$ and $G$ be a pseudo-spider produced by splitting a vertex $v \in K'$ of $G'$ into two vertices $v'$ and $v''$. So, $G$ has a $(n_1, n_2, \ldots, n_r)$-partition $P$ with configuration $F$ if, and only if, the configuration $F$ satisfies conditions of Lemma 2.2, and the following additional conditions if $v'$ is not adjacent to $v''$:*

1. $F_j' = 0$, *for $j \geq |K|$; and*

2. $F_i^S = 0$, *for $i \geq |K|$.*

*Proof.* Let $G$ be a pseudo-spider where $\{v', v''\} \subset K$ and suppose $P$ is a $(n_1, n_2, \ldots, n_r)$-partition of $G$ with configuration $F$. If vertices $v'$ and $v''$ are not adjacent, they can not be used in the same clique and the maximum clique contained in $S \cup K$ has size at most $|K|-1$. So, conditions 1 and 2 are satisfied. The arguments used in the proof of Lemma 2.2 show that $P$ obeys the conditions of that lemma.

To produce a $(n_1, n_2, \ldots, n_r)$-partition $P$ for $G$, with a configuration $F$ that satisfies the requirements of this lemma, we create a new configuration $\overline{F} = F$ and go through one of the following cases, in which we adjust some integers of $\overline{F}$.

First (case a), if $F_{k,l} \geq 1$ for some $1 \leq l \leq k \leq r$, we choose $k$ and $l$ and make $\overline{F}_{k,l} = F_{k,l} - 1$ and, if $k \geq 2$, $\overline{F}_{k-1,l-1} = F_{k-1,l-1} + 1$. Otherwise (case b), there is some $k \geq 2$ such that $F^S_k \geq 1$. We choose one and make $\overline{F}^S_k = F^S_k - 1$ and $\overline{F}^S_{k-1} = F^S_{k-1} + 1$.

It is easy to see that after these transformations, the new configuration $\overline{F}$ satisfies the requirements of Lemma 2.2 for $G'$ and it has either a $(n_1, n_2, \ldots, n_{k-1} + 1, n_k - 1, \ldots, n_r)$-partition $P'$ (if $k \geq 2$) or a $(n_1 - 1, n_2, \ldots, n_r)$-partition $P'$ (otherwise). Note that if $G'$ is a thin spider, it always fits case (a).

If we applied case (a) with $k = 1$, then let $C' \in P'$ such that $v \in C'$ and make $P = (P' \setminus \{C'\}) \cup \{\{v'\}, (C' \setminus \{v\}) \cup \{v''\}\}$.

For the other possibilities, we first consider when $v'$ is non-adjacent to $v''$. If we applied case (a) with $k \geq 2$, there is a $C' \in P'$ such that $C' \subset (K' \cup H)$ and $|C'| = k - 1$. Also, there is a $u \in (K' \setminus C')$ by Condition 1. We rename vertices of $G'$ so that $u = v$ and let $C'' \in P'$, $v \in C''$. Finally, make $P = (P' \setminus \{C', C''\}) \cup \{C' \cup \{v'\}, (C'' \setminus \{v\}) \cup \{v''\}\}$, which is a valid $(n_1, n_2, \ldots, n_r)$-partition since $v'$ and $v''$ are in different cliques.

Otherwise, we applied case (b), $G'$ is a thick spider and there is a $C' \in P'$ such that $|C'| = k - 1$ and $C' \cap S = \{u\}$. We note that $|K' \setminus C'| \geq 2$ since the transformation (b) was applied with $k \leq |K| - 1$ (by Condition 2), leading to $|C'| \leq |K'| - 1$ and $|C' \cap K'| \leq |K'| - 2$. Thus, there is a $t \in K'$ such that $t \notin C'$ and $t$ is adjacent to $u$. Let $C'' \in P'$ such that $t \in C''$, relabel vertices so that $t = v$, and make $P = (P' \setminus \{C', C''\}) \cup \{C' \cup \{v'\}, (C'' \setminus \{v\}) \cup \{v''\}\}$.

Finally, we analyze the cases when $v'$ is adjacent to $v''$. Since it was applied case (a) with $k \geq 2$ or case (b), there is a $C' \in P'$ satisfying $|C'| = k - 1$ and $C' \cap K' \neq \varnothing$ or $C' \cap S \neq \varnothing$. If $C' \cap K' \neq \varnothing$, let $u \in C' \cap K'$, relabel vertices such that $u = v$, and make $P = (P' \setminus \{C'\}) \cup \{(C' \setminus \{v\}) \cup \{v', v''\}\}$. Otherwise, $C' = \{u\}$ and $u \in S$. Take any $t \in K$ adjacent to $u$ and let $C'' \in P'$ such that $t \in C''$. Now, relabel vertices of $G'$ to have $t = v$ and make $P = (P' \setminus \{C', C''\}) \cup \{C' \cup \{v'\}, (C'' \setminus \{v\}) \cup \{v''\}\}$. □

**Lemma 2.8.** *If $G'$ is a spider with partition $(K', S', H)$, $G$ is a pseudo-spider derived from $G'$ by splitting a vertex of $S'$ into a $K_2$ or a vertex of $K'$ into a $S_2$ or a $K_2$, and $P$ is a $(n_1, n_2, \ldots, n_r)$-partition of $G$, then there is a $(n_1, n_2, \ldots, n_r)$-partition $P'$ of $G$ with configuration $F$ that satisfies $F_{2,2} = 0$ and $F_{2,1} \leq 1$.*

*Proof.* Given a $(n_1, n_2, \ldots, n_r)$-partition $P$ of $G$ with configuration $F$ we can rearrange cliques so that no $K_2$ is contained in $K$ and at most one $K_2$ intercepts both $K$ and $H$. This may occur when $G$ derives from $G'$ by splitting a vertex of $K$ into two or a vertex of $S$ into a $K_2$.

In all these cases, the configuration of $P$ satisfies $F_1^S \geq 2F_{2,2} + F_{2,1} - 1$. This allows us to apply the following transformations (which are equal to the first set of transformations of Lemma 2.3) and make $F_{2,2} = 0$: $F_2^S \leftarrow F_2^S + F_{2,2}$, $F_1^S \leftarrow F_1^S - F_{2,2}$, $F_{1,1} \leftarrow F_{1,1} + F_{2,2}$, and $F_{2,2} \leftarrow 0$.

Now we make $F_{2,1} \leq 1$ by these transformations: $x \leftarrow \min\{F_1^S, F_{2,1}\}$, $F_2^S \leftarrow F_2^S + x$, $F_1^S \leftarrow F_1^S - x$, $F_{1,0} \leftarrow F_{1,0} + x$, and $F_{2,1} \leftarrow F_{2,1} - x$.

The new configuration satisfies the requirements of lemmas 2.5 or 2.7, depending on the type of the pseudo-spider (since $P$ must satisfy the same conditions). So, there is a $(n_1, n_2, \ldots, n_r)$-partition of $G$ with the desired properties. □

### 2.4 Some other graphs

The other graphs that may show up as induced graphs by a strong module related to a neighborhood module of the MDT of a $P_4$-tidy graph are finite, and thus, easily recognized. The function $f$ is precomputed for these graphs:

- $G \simeq C_5$ or $G \simeq P_5$:

$$f(G, n_3, n_4, \ldots, n_r) = \begin{cases} 2 & \text{if } n_i = 0 \text{ for all } 3 \leq i \leq r; \\ \nexists & \text{otherwise.} \end{cases}$$

- $G \simeq \overline{P_5}$:

$$f(G, n_3, n_4, \ldots, n_r) = \begin{cases} 2 & \text{if } n_i = 0 \text{ for all } 3 \leq i \leq r; \\ 1 & \text{if } n_3 = 1 \text{ and } n_i = 0 \text{ for all } 4 \leq i \leq r; \\ \nexists & \text{otherwise.} \end{cases}$$

## 3 An algorithm for $P_4$-tidy graphs

If $G$ is a $P_4$-tidy graph, we compute the function $f$ in each node of the MDT of $G$, processing serial and parallel nodes as in Section 2.1. For neighborhood nodes, if the subgraph induced by the corresponding module is a $P_5$, $\overline{P_5}$, or $C_5$, then $f$ is computed as defined in Section 2.4. Otherwise, the induced subgraph is a spider or a pseudo-spider and $f$ is computed as the maximum of $F_{2,0} + F_2^S + F_{2,1} + F_2^e$ for every configuration $F$ satisfying the conditions given in lemmas 2.2, 2.4, 2.5, or 2.7 (according to the type of spider or pseudo-spider), plus the condition $\sum_{i=2}^{r} F_{i,i} = 0$ or $\sum_{i=2}^{r} F_{i,0} = 0$.

Optimizations given in lemmas 2.3, 2.6, and 2.8 are used to reduce the number of considered configurations when they are applicable. Moreover, a dynamic programming table is associated with each node of the MDT to store previously computed values of $f$ and speed up the computation.

**Theorem 3.1.** *There is a polynomial-time algorithm that solves the $K_r$-packing problem for $P_4$-tidy graphs, for any fixed $r$.*

*Proof.* The algorithm described above solves the $K_r$-packing problem for any $P_4$-tidy graph $G$. Since the class is hereditary, any induced subgraph of $G$ is also a $P_4$-tidy graph. The processing of serial and parallel nodes corresponds to processing join and union graphs, respectively, and it happens as in the work of Guruswami et al. for cographs. Moreover, the three simple cases of neighborhood nodes ($C_5$, $P_5$, and $\overline{P_5}$) are analyzed case by case. The only remaining possible neighborhood nodes of a $P_4$-tidy graph are spiders and pseudo-spiders, which are processed in a set of cases given by lemmas 2.2, 2.4, 2.5, and 2.7. The maximized expression is a combination of expressions in each of these lemmas ($F_2^e$ is considered 0 when not applicable). The additional condition comes from the fact that $G[K \cup H] = G[K] + G[H]$ (see Section 2.1).

The MDT of any graph is obtained in linear time [6]. We can identify if a graph induced by a module associated with a neighborhood node of the MDT of a graph is a $C_5$, $P_5$, $\overline{P_5}$, spider, or pseudo-spider in polynomial time [2], as well as identify the partition of a (pseudo-)spider in the three sets $K$, $S$, and $H$. Since the number of possibilities evaluated for spiders and pseudo-spiders is a subset of the possibilities evaluated for join graphs, the proposed algorithm has polynomial time complexity likewise the algorithm for cographs [3]. $\square$

Note that this result solves the $K_r$-factor problem for $P_4$-tidy graphs, since it suffices to check if we can pack exactly $V(G)/r$ cliques in a graph $G$.

## 4 Conclusion

We extended the algorithm proposed by Guruswami et al. [3] to solve the $K_r$-packing problem for $P_4$-tidy graphs, a superclass of cographs. The new algorithm also has polynomial time complexity for any fixed $r$.

The $P_4$-tidy graph class contains many other graph classes, so we proposed an algorithm that has a broader application scope. As a natural continuation for this work, we could investigate other classes and try to extend the algorithm even further. Unfortunately, we know few graph classes which contain the $P_4$-tidy class and have manageable neighborhood nodes on their modular decomposition trees. For instance, the subgraph induced by the module associated with a neighborhood node of the MDT of a $P_4$-laden graph may be any split graph, for which the problem is intractable.

## References

[1] M.R. Cerioli, L. Faria, T.O. Ferreira, C.A.J. Martinhon, F. Protti, and B. Reed. Partition into cliques for cubic graphs: Planar case, complexity and approximation. *Discrete Applied Mathematics*, 156:2270–2278, 2008.

[2] V. Giakoumakis, F. Roussel, and H. Thuillier. On $P_4$-tidy graphs. *Discrete Mathematics & Theoretical Computer Science*, 1(1):17–41, 1997.

[3] V. Guruswami, C. Pandu Rangan, M. S. Chang, G. J. Chang, and C. K. Wong. The $K_r$-Packing Problem. *Computing*, 66:79–89, 2001.

[4] C.T. Hoàng. A class of perfect graphs. Master's thesis, School of Computer Science, Montreal, 1983.

[5] D. G. Kirkpatrick and P. Hell. On the complexity of general graph factor problems. *SIAM J. Computing*, 12:601–609, 1983.

[6] R.M. McConnell and J.P. Spinrad. Modular decomposition and transitive orientation. *Discrete Mathematics*, 201(1-3):189–241, 1999.