On P_4-transversals of Chordal Graphs

Juraj Stacho

School of Computing Science, Simon Fraser University,
Burnaby, B.C., Canada V5A 1S6

Abstract

A P_4-transversal of a graph G is a set of vertices T which meets every P_4 of G. A P_4-transversal T is called stable if there are no edges in the subgraph of G induced by T. It has been previously shown by Hoàng and Le that it is NP-complete to decide whether a comparability (and hence perfect) graph G has a stable P_4-transversal. In the following we show that the problem is NP-complete for chordal graphs. We apply this result to show that two related problems of deciding whether a chordal graph has a P_3-free P_4-transversal, and deciding whether a chordal graph has a P_4-free P_4-transversal (also known as a *two-sided* P_4-transversal) are both NP-complete. Additionally, we strengthen the main results to strongly chordal graphs.

Key words: chordal graphs, P_4-transversal

1 Introduction and results

A graph is *perfect* if for every induced subgraph H of G, the chromatic number $\chi(H)$ is equal to the clique number $\omega(H)$. The Strong Perfect Graph Theorem states that a graph is perfect if and only if it has no induced odd cycle or its complement [2]. This result had been conjectured by Berge [1]. In the long history of this conjecture, the study of the structure of P_4's in a graph has been found to play an important role. In [8], the authors define the notion of a P_4-*transversal* to be a subset of vertices of a graph meeting every P_4. They show that if a graph has a P_4-transversal with certain properties it is guaranteed to be perfect. They also investigate the complexity of finding a P_4-transversal with various properties. In particular they investigate stable P_4-transversals, i.e., P_4-transversals which form a *stable set* – a set of vertices inducing a subgraph with no edges. They show that for comparability graphs

Email address: `jstacho@cs.sfu.ca` (Juraj Stacho).

(and therefore also for perfect graphs) it is NP-complete to decide whether a graph has a stable P_4-transversal. In [9], the authors consider a related problem of P_4-colourings. They show that finding a P_4-free P_4-transversal (a "P_4-free 2-colouring" in their terminology) is NP-complete for comparability graphs, P_5-free graphs and (C_4, C_5)-free graphs.

Here we first show that the problem of finding a stable P_4-transversal remains NP-complete when restricted to chordal graphs:

Theorem 1.1 *It is NP-complete to decide whether a given chordal graph has a stable P_4-transversal.*

We apply this result to derive the following consequences:

Theorem 1.2 *It is NP-complete to decide whether a given chordal graph has a P_3-free P_4-transversal.*

Theorem 1.3 *It is NP-complete to decide whether a given chordal graph has a P_4-free P_4-transversal.*

Note that Theorem 1.3 also improves on the results from [9] mentioned above. We contrast Theorem 1.1 with the result of [4], which can be reformulated as follows:

Theorem 1.4 [4] *For chordal graphs, a stable P_3-transversal can be found in polynomial time.*

We note that the NP-completeness of these kinds of partition problems for general graphs has been proved in [6].

In the last section of the paper we discuss some extensions of these results.

2 Preliminaries

A graph G is called *H-free* if G contains no induced subgraph isomorphic to a graph H. In particular, a P_4-free graph is called a *cograph.* (Recall that P_4 is the path with four vertices and three edges.) It has been shown [3] that any P_4-free graph can be constructed from a single vertex using the operations of disjoint union and join. (The *join* of two graphs is constructed by taking their disjoint union and adding all possible edges between the two graphs.) The construction of a cograph G can be therefore represented as a rooted tree T in which the leaves are the vertices of the graph G, and the internal nodes are labeled either 0 or 1, denoting the operations of disjoint union and join respectively. T shall be referred to as a *tree representation* of G. It could be easily seen that two vertices of G are adjacent if and only if their least common ancestor in T is labeled 1. Note that T is not necessarily unique. We call T a *cotree* if the labels of the internal nodes of T strictly alternate on any path in T. It is known[3] that every cograph has a unique cotree (up to isomorphism). If a tree representation T of a cograph G is not a cotree, one can easily transform it into an equivalent cotree by identifying consecutive vertices of T having the same label. Hence for simplicity, we shall refer to any tree representation of a cograph as a cotree.

A graph is *chordal* if it does not contain an induced cycle of length 4 or more. It is known [7] that a graph is *chordal* if and only if there exists a linear ordering $\prec$ of its vertices such that if v, w are two neighbours of u with $u \prec v$, $u \prec w$, then v and w are adjacent. Such an ordering is called a *perfect elimination ordering.*

A *literal* is a variable v_i or its negation $\neg v_i$ (often written as $\bar{v}_i$). A *clause* is a disjunction of literals. A propositional formula is in *conjunctive normal form* if it is written as a conjunction of clauses. The set of all variables of the formula φ is denoted by $var(\varphi)$. The truth assignment τ for the set of variables $var(\varphi)$ is a mapping $\tau : var(\varphi) \to \{true, false\}$. The *3-satisfiability* problem $3SAT$ is the problem of finding a satisfying truth assignment for all variables of a given formula in conjunctive normal form in which every clause has exactly 3 literals. It is known to be NP-complete.

It should be noted that all the problems mentioned in section 1 are clearly in NP; this follows from the fact that testing whether a graph contains a P_4, a P_3, or is a stable set, can be done in polynomial time.

3 Stable P_4-transversals

To prove Theorem 1.1, we describe a polynomial time reduction from the problem $3SAT$. Let φ be a formula in conjunctive normal form with exactly three literals in any clause, i.e. $\varphi = \bigwedge_{j=1}^{m} C_j$ where $C_j = l_1^j \vee l_2^j \vee l_3^j$ where l_1^j, l_2^j, l_3^j are literals. Let $var(\varphi) = \{v_1, v_2, \ldots, v_n\}$ be all variables appearing in φ. Let $J_i^{(+)}$ be the indices of clauses which contain the literal v_i and $J_i^{(-)}$ be the indices of clauses which contain the literal $\neg v_i$.

Let $C(C_j)$ be the graph shown in Figure 1, and let $G(\varphi)$ for the formula φ be the graph G_n inductively defined as follows:

(1) Let G_0 be the disjoint union of the graphs $\{C(C_j)\}_{j=1}^{m}$ (see Figure 1).

(2) Let G_i (see Figure 1) be the graph created from G_{i-1} as follows. Add two adjacent vertices v_i and $\overline{v}_i$ and make them completely adjacent to all

vertices of G_{i-1}. For every $j \in J_i^{(+)}$ add a vertex v_i^j adjacent to v_i and adjacent to the vertex l_k^j of $C(C_j)$ if v_i is the k-th literal of the clause C_j. Similarly, for every $j \in J_i^{(-)}$ add a vertex $\overline{v}_i^j$ adjacent to $\overline{v}_i$ and adjacent to the vertex l_k^j in $C(C_j)$ if $\neg v_i$ is the k-th literal of the clause C_j. (Note that we can assume that a literal occurs in a clause only once.)

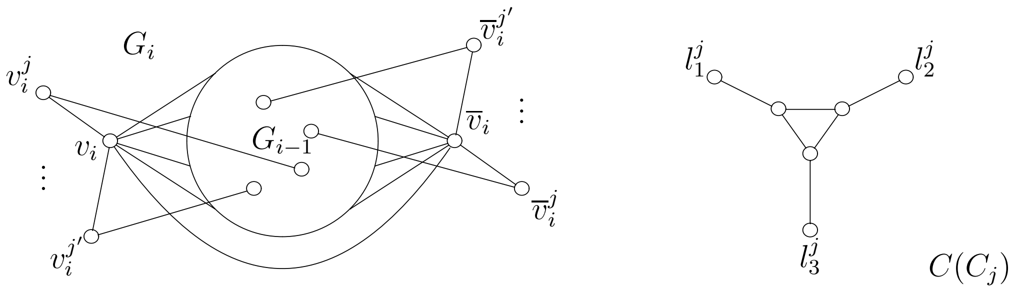

Fig. 1. The graphs G_i and $C(C_j)$. Note that the circle in the center is the graph G_{i-1} completely adjacent to v_i and $\overline{v}_i$; and each of the vertices v_i^j and $\overline{v}_i^j$ is adjacent to a single vertex in G_{i-1}. The graph $C(C_j)$ corresponds to the clause $C_j = l_1^j \vee l_2^j \vee l_3^j$.

We can also describe the graph $G(\varphi)$ in the following (non-inductive) way:

(1) $G(\varphi)$ contains the vertices $v_i, \overline{v}_i$ for every i, the vertices v_i^j (resp. $\overline{v}_i^j$) for every occurrence of the literal v_i (resp. $\neg v_i$) in the clause C_j, and the vertices of $C(C_j)$ for every clause C_j containing among others the vertices l_1^j, l_2^j and l_3^j.
(2) The vertex v_i (resp. $\overline{v}_i$) is adjacent to all vertices of $C(C_j)$ for every j, to the vertices $v_{i'}, \overline{v}_{i'}$ for all i', to all vertices v_i^j (resp. $\overline{v}_i^j$) that may exist, and to all vertices $v_{i'}^j, \overline{v}_{i'}^j$ that may exist, for each i' with $i' < i$.
(3) The vertex v_i^j (resp. $\overline{v}_i^j$) is adjacent to the vertex v_i (resp. $\overline{v}_i$), to the vertex l_k^j such that v_i (resp. $\neg v_i$) is the k-th literal of the clause C_j, and to all vertices $v_{i'}, \overline{v}_{i'}$ for all i' with $i' > i$.
(4) The vertex l_k^j is adjacent to its only neighbour in $C(C_j)$, to the vertex v_i^j (resp. $\overline{v}_i^j$) such that v_i (resp. $\neg v_i$) is the k-th literal of the clause C_j, and to the vertices $v_{i'}, \overline{v}_{i'}$ for all i'.
(5) The remaining vertices of $C(C_j)$ are only adjacent to their respective neighbours in $C(C_j)$ and to the vertices $v_i, \overline{v}_i$ for all i.

First we need the following proposition and its corollary:

Proposition 3.1 *For all i, the graph G_i is chordal.*

Proof. We prove this by induction. For $i = 0$, observe that the graph $C(C_j)$ is chordal for every j, hence G_0 is chordal. For $i > 0$, suppose that G_{i-1} is chordal; let π be a perfect elimination ordering of its vertices. Now it is not difficult to see that $v_i^1, v_i^2, \ldots, \overline{v}_i^1, \overline{v}_i^2, \ldots, \pi, v_i, \overline{v}_i$ is a perfect elimination ordering of G_i. $\square$

Corollary 3.2 *The graph $G(\varphi)$ is chordal.*

We make the following observations about the graph $G(\varphi)$ and its subgraphs:

Observation 3.3 *Every stable P_4-transversal of the graph $C(C_j)$ contains at least one of the vertices l_1^j,l_2^j or l_3^j. Every maximal stable set of $C(C_j)$ is a P_4-transversal.*

Proof. The proof is by inspection. □

Proposition 3.4 *Let S be a stable P_4-transversal of $G(\varphi)$. Then the following holds:*

(1) For all i, $v_i \notin S$ and $\overline{v}_i \notin S$.
(2) For any j, $v_i^j \notin S$ implies for all j', $\overline{v}_i^{j'} \in S$

Proof.

(1) The vertex v_i is adjacent to all vertices of $C(C_j)$, and so if it belongs to the stable set S, then no vertex of $C(C_j)$ can be in S. Therefore $G(\varphi) \setminus S$ contains all vertices of $C(C_j)$ and hence contains a P_4, contrary to S being a P_4-transversal. The same holds for $\overline{v}_i$.
(2) Suppose that $v_i^j \notin S$ and also $\overline{v}_i^{j'} \notin S$ for some j, j'. Then by the previous argument also $v_i \notin S$ and $\overline{v}_i \notin S$, and hence S cannot be a P_4-transversal since the vertices $v_i^j, v_i, \overline{v}_i, \overline{v}_i^{j'}$ form a P_4 in $G(\varphi) \setminus S$.

□

Lemma 3.5 *The formula φ is satisfiable if and only if the graph $G(\varphi)$ has a stable P_4-transversal.*

Proof. First suppose that τ is a satisfying truth assignment of φ. We use τ to construct a stable P_4-transversal of $G(\varphi) = G_n$.

Let S_0^j be any maximal stable set in $C(C_j)$ with the following property. For all k, the vertex $l_k^j \in S_0^j$ if and only if v_i is the k-th literal of the clause C_j and $\tau(v_i) = true$, or $\neg v_i$ is the k-th literal of the clause C_j and $\tau(v_i) = false$. Clearly since τ satisfies φ and therefore also satisfies the clause C_j, at least one of the vertices l_1^j, l_2^j, l_3^j is in S_0^j. By Observation 3.3, S_0^j is a P_4-transversal in $C(C_j)$. Now let $S_0 = \bigcup_{j=1}^m S_0^j$. Since the graphs $C(C_j)$ in G_0 are vertex disjoint, it follows that S_0 is a stable P_4-transversal of G_0.

Now let $S = S_0 \cup \bigcup_{i=1}^n S_i^+$ where

$$S_i^+ = \begin{cases} \{\overline{v}_i^j\}_{j \in J_i^{(-)}} & \text{if } \tau(v_i) = true \\ \{v_i^j\}_{j \in J_i^{(+)}} & \text{if } \tau(v_i) = false \end{cases}$$

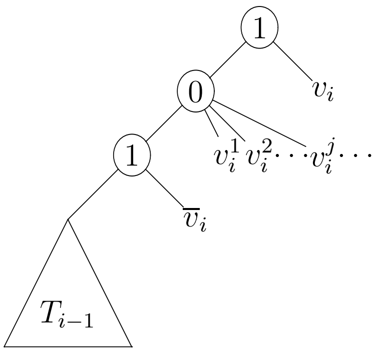

Fig. 2. The cotree for the graph $G_i \setminus S_i$

We show that S is a stable P_4-transversal of $G(\varphi)$. First let $S_i = S_0 \cup \bigcup_{j=1}^{i} S_j^+$. Clearly $S_i = S_{i-1} \cup S_i^+$ and $S_{i-1} \subseteq S_i$. We show by induction that S_i is a stable P_4-transversal of G_i.

For $i = 0$ the claim follows from the above. Therefore suppose that $i > 0$ and S_{i-1} is a stable P_4-transversal of the graph G_{i-1}. Without loss of generality, we may assume that $\tau(v_i) = true$. Then $S_i = S_{i-1} \cup \{\overline{v}_i^j\}_{j \in J_i^{(-)}}$, where S_{i-1} is a stable set. Using the fact that $\tau(v_i) = true$ and the definition of S_0^j, we have that $l_k^j \notin S_0^j$ whenever $\neg v_i$ is the k-th literal of the clause C_j. Since in that case l_k^j and $\overline{v}_i$ are the only neighbours of the vertex $\overline{v}_i^j$ in G_i, it easily follows that S_i is a stable set. Now we only need to show that S_i is a P_4-transversal of G_i, that is that $G_i \setminus S_i$ is a P_4-free graph. From the induction hypothesis $G_{i-1} \setminus S_{i-1}$ is already a P_4-free graph. Therefore there exists a cotree T_{i-1} for this graph. To show the claim we construct a cotree for $G_i \setminus S_i$. As in the previous argument, it follows that $l_k^j \in S_0^j$, whenever v_i is the k-th literal of the clause C_j. Since in that case the vertex v_i^j is only adjacent to the vertex v_i in $G_i \setminus S_i$, we obtain the cotree for $G_i \setminus S_i$ as shown in Figure 2.

It follows that $S = S_n$ is a stable P_4-transversal of the graph $G(\varphi) = G_n$.

Now suppose that $G(\varphi)$ has a stable P_4-transversal, say S. We construct the truth assignment τ for the formula φ in the following way: for every variable v_i we set $\tau(v_i) = true$ just if for some j the vertex $v_i^j \notin S$. We show that τ satisfies φ.

Consider the clause C_j of φ. Since S is a stable P_4-transversal of $G(\varphi)$, the set $S \cap C(C_j)$ is a stable P_4-transversal of $C(C_j)$. It follows from Observation 3.3 that the vertex $l_k^j \in S \cap C(C_j)$ for some k, and hence $l_k^j \in S$. If v_i is the k-th literal of the clause C_j, it follows that v_i^j is not in the stable set S. (Recall that the vertex v_i^j is a neighbour of l_k^j.) Therefore by the definition of τ, we have that $\tau(v_i) = true$, and therefore τ satisfies C_j. If $\neg v_i$ is the k-th literal of the clause C_j, we deduce that $\overline{v}_i^j \notin S$. By Proposition 3.4, we must have for all j', $v_i^{j'} \in S$. Therefore it follows from the definition of τ that $\tau(v_i) = false$, and we again conclude that τ satisfies C_j.

Clearly, since τ satisfies all clauses C_j, it satisfies the formula φ; this concludes the proof. □

Proof. [Theorem 1.1] One can easily see that the graph $G(\varphi)$ can be constructed in polynomial time. Hence the claim follows from Lemma 3.5. □

4 P_3-free and P_4-free P_4-transversals

We now proceed to prove Theorem 1.2 and 1.3.

Proposition 4.1 *Let Y be the graph shown in Figure 3.*

(1) Every P_3-free P_4-transversal S of the graph Y has the property that $u \in S$ and $v \notin S$ (see Figure 3a).
(2) Every P_4-free P_4-transversal S of the graph Y has the property that either $u \in S$ and $v \notin S$, or $u \notin S$ and $v \in S$ (see Figure 3a,3b).

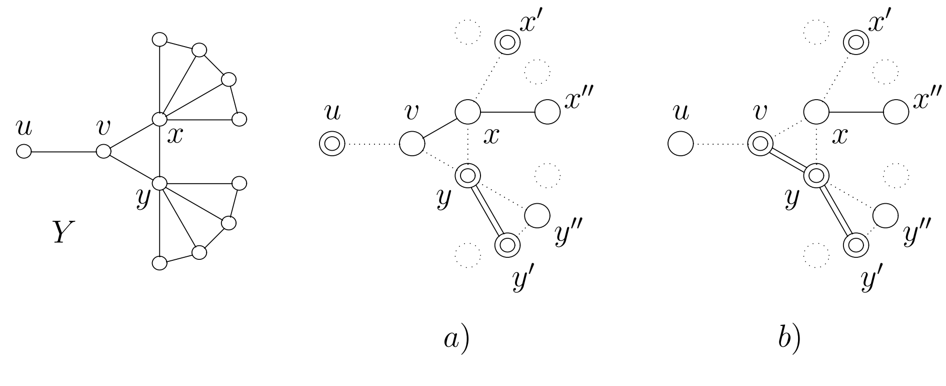

Fig. 3. The graph Y and a sketch of some possible P_4-transversals (the doubly circled vertices)

Proof. Observe that among the neighbours of x and the neighbours of y there must always be a vertex $x' \in S$ and a vertex $x'' \notin S$, and similarly a vertex $y' \in S$ and a vertex $y'' \notin S$. (Clearly both neighbourhoods contain a P_4, so they cannot be entirely in S, nor entirely not in S.) It follows that x and y cannot both be in S, and cannot both be not in S. (In the former case the vertices y', y, x, x' form a P_4 in S, and in the latter case, the vertices y'', y, x, x'' form a P_4 in $Y \setminus S$.) Without loss of generality we may assume that $y \in S$ and $x \notin S$.

First suppose that S is P_3-free. Then we have $v \notin S$, since otherwise the vertices v, y, y' form a P_3 in S. Moreover u must be in S, since otherwise the vertices u, v, x, x'' form a P_4 in $Y \setminus S$. This proves the first part of the claim.

Now suppose that S is P_4-free. Then either $v \notin S$ and we similarly find that $u \in S$, or $v \in S$ and then we have that $u \notin S$, since otherwise the vertices u, v, y, y' form a P_4 in S. This concludes the proof. □

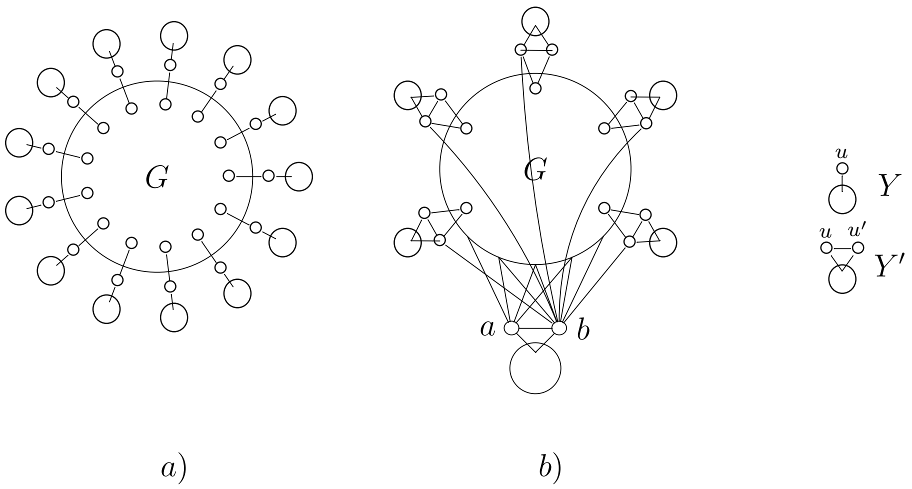

Fig. 4. The construction of the graph G' used in the proof of Theorem 1.2 (a) and Theorem 1.3 (b)

Proof. [Theorem 1.2] In order to prove the NP-completeness of the problem of recognizing the existence of a P_3-free P_4-transversal, we construct a polynomial time reduction from the stable P_4-transversal problem for chordal graphs.

Let G be a chordal graph. Let G' be the graph constructed from G in the following way (see Figure 4a). For every vertex $w \in V(G)$ we add a copy Y_w of the graph Y (see Figure 3) in which we change the labels of the vertices u and v to u_w and v_w respectively, and we make the vertices w and u_w adjacent. Observe that G' is chordal, since both G and Y are chordal. We now prove that G has a stable P_4-transversal if and only if G' has a P_3-free P_4-transversal.

Suppose that G has a stable P_4-transversal S. Let S_w be any P_3-free transversal of Y_w. Let $S' = S \cup \bigcup_{w \in V(G)} S_w$. By Proposition 4.1, we have $u_w \in S'$ for all $w \in V(G)$. Since for every w the vertex u_w is a cut-vertex in G', it easily follows that S' is a P_4-transversal of G'. Moreover, since S is stable, and for every w the vertex $v_w \notin S'$, it follows that S' is P_3-free.

Now suppose that G' has a P_3-free P_4-transversal S'. Let $S = S' \cap V(G)$. Clearly S is a P_4-transversal of G. We show that S is also stable, thus proving the claim. Suppose that there are two adjacent vertices $w, w' \in S$. By Proposition 4.1, we have $u_w \in S'$ and $u_{w'} \in S'$. Therefore the vertices $u_w, w, w', u_{w'}$ form a P_4 in S' contrary to S' being P_3-free. □

Proof. [Theorem 1.3] As in the previous proof, we construct a polynomial time reduction from the stable P_4-transversal problem for chordal graphs. Let G be a chordal graph. Let Y' be the graph obtained from Y by adding an additional vertex u' adjacent to both u and v. Let G' be the graph obtained from G as follows (see Figure 4b). For every vertex $w \in V(G)$ we add a copy Y'_w of the graph Y' in which we change the labels of the vertices u, u' and v to u_w, u'_w and v_w respectively, and we make w adjacent to u_w and u'_w. Moreover, we add a copy Y'' of the graph Y', in which we change the labels of the vertices u, u' and v to a, b and f respectively. We make a and b adjacent to all vertices of G, and make b adjacent to u_w for every $w \in V(G)$.

Observe that G' is chordal. Indeed, since G is chordal, it has a perfect elimination ordering π. Similarly, since Y'_w is chordal, it also has a perfect elimination ordering π_w. It is easy to see, by inspection, that we may choose π_w to end with the vertices u'_w and u_w, in that order. Lastly, since Y'' is chordal, let π'' be any perfect elimination ordering of Y''. Now one can easily verify that $\pi_{w_1}, \pi_{w_2}, \ldots, \pi, \pi''$ is a perfect elimination ordering of G', where $w_1, w_2, \ldots$ is an enumeration of the vertices of G.

We now prove that G has a stable P_4-transversal if and only if G' has a P_4-free P_4-transversal. Suppose that G has a stable P_4-transversal S. Let S_w be a P_4-free P_4-transversal of Y'_w satisfying $u_w, u'_w \in S_w$. Let S'' be a P_4-free P_4-transversal of Y'' satisfying $a, b \notin S''$; it also follows that $f \in S''$. Now let $S' = S \cup S'' \cup \bigcup_{w \in V(G)} S_w$. We show that S' is a P_4-free P_4-transversal of the graph G'.

Since S_w is P_4-free, let T_w be the cotree representing $S_w \setminus \{u_w, u'_w\}$. Similarly, let T'' be the cotree for S'', $\overline{T}$ the cotree for $G \setminus S$, $\overline{T}_w$ the cotree for $Y_w \setminus S_w$, and $\overline{T}''$ the cotree for $Y'' \setminus (S'' \cup \{a, b\})$. Let T' and $\overline{T}'$ be the cotrees depicted on Figure 5.

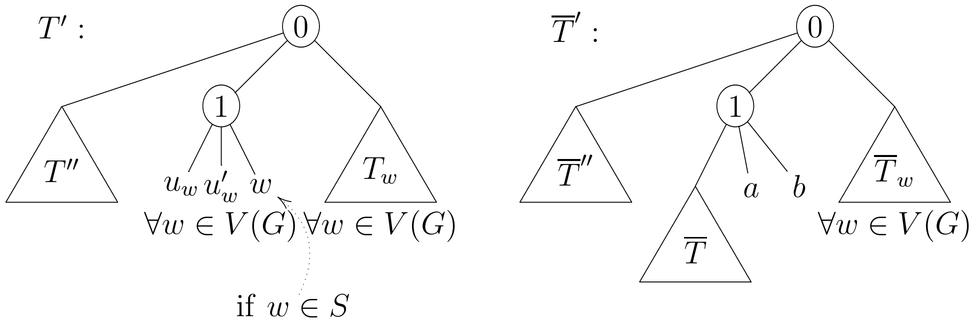

Fig. 5. The cotrees corresponding to the subgraphs of G' induced on S' and $G' \setminus S'$. (The subtrees marked with $\forall w \in V(G)$ indicate that for every vertex in G, such subtree is added.)

One can easily verify that T' and $\overline{T}'$ are exactly the cotrees of the subgraphs of G' induced on S' and $G' \setminus S'$ respectively. That shows that S' and $G' \setminus S'$ are both P_4-free, and hence S' is a P_4-free P_4-transversal of the graph G'.

Now suppose that S' is a P_4-free P_4-transversal of G'. We may assume that the vertex f of Y'' is in S'. (Otherwise we consider $G' \setminus S'$ in place of S' since both S' and $G' \setminus S'$ are P_4-free.) Now it follows from Proposition 4.1 that $a, b \notin S'$. Similarly, it follows that for every w either $u_w, u'_w \in S'$ and $v_w \notin S'$, or $u_w, u'_w \notin S'$ and $v_w \in S'$. Since the latter would create a P_4 in $G' \setminus S'$ (i.e., the vertices a, b, u_w, u'_w form a P_4 in G'), it follows that $u_w, u'_w \in S'$ and $v_w \notin S'$ for every $w \in V(G)$.

Now let $S = S' \cap V(G)$. We show that S is a stable P_4-transversal of the graph G. Clearly S is a P_4-transversal of G. We only need to show that S is also stable. Suppose otherwise, i.e., let $w, w' \in S$ be adjacent. Then the vertices $u_w, w, w', u_{w'}$ clearly form a P_4 in S' (recall that $u_w \in S'$ for all $w \in V(G)$), which leads to a contradiction since S' is P_4-free. Hence S must be stable. $\square$

5 Further Results

A graph G is *strongly chordal* if it is chordal and there exists a perfect elimination ordering $\prec$ of the vertices of G such that if $u \prec v \prec w \prec z$ and (u, z), (u, w) and (v, w) are edges of G then also (v, z) is an edge (such ordering is called *strong elimination ordering*). Strongly chordal graphs form an interesting subclass of chordal graph as there are several difficult combinatorial graph problems that are polynomially solvable in strongly chordal graphs, but are NP-complete for chordal graphs [5].

In the previous sections we proved that it is NP-complete to decide whether a chordal graph has a stable P_4-transversal, a P_3-free P_4-transversal or a P_4-free transversal. It is easy to check that the perfect elimination orderings of $G(\varphi)$ and G' used in the proofs of these results are in fact strong elimination orderings (provided G is strongly chordal in the case of G'). Note that it suffices to show this for $G(\varphi)$ and G' from Theorem 1.3 since G' from Theorem 1.2 is an induced subgraph of G' from Theorem 1.3. Hence we obtain the following stronger result.

Theorem 5.1 *(1) It is NP-complete to decide whether a strongly chordal graph has a stable P_4-transversal.*

(2) It is NP-complete to decide whether a strongly chordal graph has a P_3-free P_4-transversal.

(3) It is NP-complete to decide whether a strongly chordal graph has a P_4-free P_4-transversal.

Acknowledgements

The author would like to thank his advisor Pavol Hell for directing this research and for his help with the preparation of this article. The author would also like to thank anonymous referees for their useful comments that greatly improved the presentation of this work.

References

[1] C. Berge: *Färbung von Graphen deren sämtliche bzw. deren ungerade Kreise starr sind*, Wiss. Zeitschrift, Martin-Luther-Univ. Halle-Wittenberg, Math.-Natur. Reihe 10 (1961), 114–115.

[2] M. Chudnovsky, N. Robertson, P. Seymour, R. Thomas: *The Strong Perfect Graph Theorem*, Annals of Mathematics 164 (2006), 51–229.

[3] D.G. Corneil, H. Lerchs, L. Stewart: *Complement Reducible Graphs*, Discrete Applied Mathematics 3 (1981), 163–174.

[4] T. Ekim, P. Hell, J. Stacho, D. de Werra: *Polar chordal graphs*, manuscript.

[5] M. Farber: *Domination, independent domination, and duality in strongly chordal graphs*, Discrete Applied Mathematics 7 (1984), 115–130.

[6] A. Farrugia: *Vertex-Partitioning into Fixed Additive Induced-Hereditary Properties is NP-hard*, Electron. J. Combin. 11 (2004).

[7] M.C. Golumbic: **Algorithmic Graph Theory and Perfect Graphs**, Academic Press, New York, 1980.

[8] C. T. Hoàng, V. B. Le: *On P_4-transversals of perfect graphs*, Discrete Mathematics 216 (2000), 195–210.

[9] C. T. Hoàng, V. B. Le: *P_4-free Colorings and P_4-Bipartite Graphs*, Discrete Mathematics and Theoretical Computer Science 4 (2001), 109–122.

[10] J. Stacho: Ph.D. Thesis, Simon Fraser University, 2008, to appear.