4. Baumann, S., A linear algorithm for the homogeneous decomposition of graphs, Report No. M-9615, Zentrum für Mathematik, Technische Universität München, 1996.
5. Bodlaender, H., J. S. Deogun, K. Jansen, T. Kloks, D. Kratsch, H. Müller and Z. Tuza, Rankings of graphs, *SIAM Journal on Discrete Mathematics* **11**, (1998), pp. 168–181.
6. Bodlaender, H., J. R. Gilbert, H. Hafsteinsson and T. Kloks, Approximating treewidth, pathwidth, frontsize and shortest elimination tree, *Journal of Algorithms* **18**, (1995), pp. 238–255.
7. Deogun, J. S., T. Kloks, D. Kratsch and H. Müller, On vertex ranking for permutation and other graphs, *Proceedings* 11^{th} *Annual Symposium on Theoretical Aspects of Computer Science (STACS '94)*, Springer-Verlag, Lecture Notes in Computer Science, 775, pp. 747–758.
8. Gustedt, J., On the pathwidth of chordal graphs, *Discrete Applied Mathematics* **45**, (1993), pp. 233–248.
9. Hochstättler, W. and G. Tinhofer, Hamiltonicity in graphs with few P_4's, *Computing* **54**, (1995), pp. 213–225.
10. Jamison, B. and S. Olariu, p-components and the homogeneous decomposition of graphs, *SIAM J. Discrete Mathematics* **8**, (1995), pp. 448–463.
11. Jansen, K. and P. Scheffler, Generalized coloring for tree-like graphs, *Discrete Appl. Math.* **75**, (1997), pp. 135–155.
12. Kloks, T., *Treewidth–Computations and Approximations*, Springer–Verlag, LNCS 842, (1994).
13. Kloks, T., H. Müller and C. K. Wong, Vertex ranking of asteroidal triple-free graphs, *Proceedings of ISAAC'96*, Lecture Notes in Computer Science 1178, Springer-Verlag, 1996, pp. 174–182.
14. Möhring, R. H., Algorithmic aspects of comparability graphs and interval graphs, in: I. Rival, ed., *Graphs and Orders*, Dordrecht, Holland, 1985.
15. Schäffer, A. A., Optimal node ranking of trees in linear time, *Information Processing Letters* **33**, (1989/1990), pp. 91–96.
16. Scheffler, P, Node ranking and searching on graphs (abstract), 3^{rd} *Twente Workshop on Graphs and Combinatorial Optimization*, U. Faigle and C. Hoede, eds., Memorandum No. 1132, Faculty of Applied Mathematics, University of Twente, The Netherlands, 1993.

On vertex ranking, pathwidth and path cover number of $(q, q-4)$-graphs

Ton Kloks[1], Jan Kratochvíl[1*], Dieter Kratsch[2] and Haiko Müller[2]

[1] Department of Applied Mathematics and DIMATIA
Charles University
Malostranské nám. 25, 118 00 Praha 1
Czech Republic
{kloks,honza}@kam.ms.mff.cuni.cz

[2] Friedrich-Schiller-Universität Jena
Fakultät für Mathematik und Informatik
07740 Jena
Germany
{kratsch,hm}@minet.uni-jena.de

Abstract. We show that the vertex ranking problem, the pathwidth and the path cover number can all be solved in linear time for the class of $(q, q-4)$-graphs, for every fixed q. These are graphs for which no set of at most q vertices induces more than $q-4$ different P_4's.

1 Introduction

It was shown in [2] that the weighted versions of the clique and independent set problem and the chromatic number and clique cover problem can all be solved in linear time for so-called $(q, q-4)$-graphs, for fixed q. In a later paper [1], it was shown that also the treewidth and minimum fill-in problem (for definitions see, e.g., [12]) can be solved in linear time for these classes of graphs. These results are due to a unique tree-decomposition, the *primeval* decomposition, introduced by Jamison and Olariu [10]. It turns out that, in case of $(q, q-4)$-graphs for fixed q, the problems mentioned above can be solved efficiently for the subgraphs corresponding with the leaves of this tree (the p-components), and that the graph parameters for the subgraphs corresponding with internal nodes can be computed from the corresponding parameters of the children of the node.

We show that the vertex ranking problem can be solved in linear time for the class of $(q, q-4)$-graphs, for every fixed q, using a similar technique.

In [9] is was shown that the path cover number can be computed in linear time for P_4-sparse and P_4-extendible graphs. We extend these results by showing that the path cover number can be computed for $(q, q-4)$-graphs for every fixed

* Research supported in part by the Czech Research Grants GAUK 194 and GAČR 201/1996/0194.

q, thereby settling an open problem mentioned in [9]. (The P_4-sparse graphs are exactly the $(5,1)$-graphs and the C_5-free P_4-extendible graphs are exactly the $(6,1)$-graphs.)

An open problem mentioned in [1] was the pathwidth of $(q, q-4)$-graphs. We show that the pathwidth problem is solvable in linear time for these graphs. The pathwidth problem is NP-complete when restricted to chordal graphs [8].

We mention some results on list colorings in the concluding remarks.

2 Preliminaries

Recently Jamison and Olariu introduced the *homogeneous decomposition* of a graph [10], which extends the well known *modular decomposition* [14]. We use the *primeval tree-decomposition* introduced in [10], and the characterization of the p-connected components of $(q, q-4)$-graphs determined in [2] to solve the vertex ranking problem of $(q, q-4)$-graphs.

As usual we denote by P_k a chordless path with k vertices.

Definition 1. A graph $G = (V, E)$ is *p-connected* if for every partition of V into non-empty subsets V_1 and V_2 there is a *crossing* P_4, that is, a P_4 with vertices both in V_1 and in V_2.

Definition 2. A maximal subset C of vertices such that $G[C]$ is p-connected is called a p-component of G.

It is easy to see (see, e.g., [10]) that each graph has a unique partition into p-components. The p-components are connected and closed under complementation, i.e., a p-component of G is also a p-component of $\overline{G}$.

Definition 3. A p-component C is called *separable* if there is a partition of C into non-empty subsets C_1 and C_2 such that every P_4 with vertices both in C_1 and in C_2 has both midpoints in C_1 and both endpoints in C_2.

Definition 4. A subset M of V with $1 < |M| < |V|$ is called a *homogeneous set* if each vertex outside is either adjacent to all vertices of H or to none of them. The graph obtained from G by shrinking every maximal homogenous set to one single vertex is called the *characteristic graph* of G.

If $1 \leq |M| \leq |V|$ and if every vertex outside M is adjacent to all or none of the vertices of M then we call M a *module*.

One of the results of [10] (see also [2]) is the following.

Lemma 5. *A p-connected graph G is separable if and only if its characteristic graph is a split graph.*

It is shown in [10] that if a p-component C is separable, then the partition (C_1, C_2) is *unique*. We call (C_1, C_2) the separation of C.

We need the main result of [10], called the *structure theorem*.

5.1 G_1 is a spider

We can refer one more time to [9]. Let the clique of G_1 be K.

Lemma 30. *If G_1 has thin legs then*

$$\pi(G) = \pi(G_2) + \max\left\{0, \left\lceil \tfrac{1}{2}|K| \right\rceil - \pi(G_2)\right\}$$

and if G_1 has thick legs with $|K| > 2$ then $\pi(G) = \pi(G_2)$.

5.2 $|V_1| < q$ and the characteristic of G_1 is a split graph

We assume we have a minimum path cover for G_2 and we consider all possible path covers for G_1. For every path cover of G_1, split the paths of G_2 until the path covers have equal size or until every path in G_2 has one vertex. Add edges from V_1^1 to V_2 between endvertices of paths as long as possible. The graph is Hamiltonian if there are only cycles left, and otherwise the number of paths in the obtained path cover for G is a lower bound for $\pi(G)$. Minimizing over all possible path covers for $\pi(G)$ decides whether G has a Hamiltonian cycle and determines the path cover number for G.

Since G_1 has constant size, the above procedure can be implemented to take constant time, and we obtain the following result.

Theorem 31. *For every integer $q \geq 4$ there exists a linear time algorithm to decide whether a $(q, q-4)$-graph is Hamiltonian and to determine its path cover number.*

6 Concluding remarks

One of the open problem we like to mention is the *bandwidth* problem. Results were obtained for the subclass of cographs, but, as far as we know, for $(q, q-4)$-graphs this problem is still open.

We make a remark on *list coloring*. The problem is NP-complete for cographs [11]. However, if we restrict the number of colors in the union of the lists by a constant then it can be seen that the problem is linear time solvable for $(q, q-4)$-graphs for fixed q. The restricted variant pre-color extension is also solvable in linear time, even if the total number of colors in each list is unbounded.

References

1. Babel, L., Triangulating graphs with few P_4's. Manuscript 1997.
2. Babel, L., On the P_4-structure of graphs, *Habilitationsschrift*, Zentrum für Mathematik, Technische Universität München, 1997.
3. Babel, L. and S. Olariu, On the isomorphism of graphs with few $P4$s, in: M. Nagl, ed., *Graph-Theoretic Concepts in Computer Science, 21^{th} International Workshop, WG'95*, Lecture Notes in Computer Science 1017, pp. 24–36, Springer, Berlin, 1995.

Theorem 25. *For every q there exists a linear time algorithm to compute the pathwidth for $(q, q-4)$-graphs.*

5 A path cover algorithm for $(q, q-4)$-graphs

Definition 26. A family $P_1, \ldots, P_k$ of paths in G is called a *path cover* of G if every vertex of G is contained in exactly one of these paths. The *path cover number* $\pi(G)$ of G is the minimum cardinality of a path cover of G.

In this section we show how to decide Hamiltonicity and how to compute the path cover number of a $(q, q-4)$-graph for fixed q in linear time. This extends the results of [9]. In this paper P_4-sparse and P_4-reducible graphs were considered. A graph is called *P_4-extendible* if and only if it has no p-component with more than five vertices. A graph is called *P_4-sparse* if it has no induced subgraph isomorphic to one of seven p-connected graphs on five vertices (see [9]). It turns out that the P_4-sparse graphs are exactly the $(5, 1)$-graphs and the P_4-extendible graphs without induced C_5 are exactly the $(6, 1)$-graphs (see [1]).

Our first observation concerning the disjoint union of two graphs is again obvious.

Lemma 27. *Let G be the disjoint union of G_1 and G_2. Then $\pi(G) = \pi(G_1) + \pi(G_2)$.*

Lemma 28. *Let G be the complete sum of G_1 and G_2. Let $\mathcal{M} = \max(\pi(G_1) - |V_2|, \pi(G_2) - |V_1|)$. G is Hamiltonian if and only if $\mathcal{M} \leq 0$. The path cover number satisfies $\pi(G) = \max(1, \mathcal{M})$.*

Proof. Assume G is Hamiltonian. Consider a Hamiltonian cycle. This induces a path cover $P_1, \ldots, P_\ell$ for G_1 and a path cover $Q_1, \ldots, Q_\ell$ for G_2. Then clearly $\pi(G_1) \leq \ell \leq |V_2|$ and $\pi(G_2) \leq \ell \leq |V_1|$. Hence $\mathcal{M} \leq 0$. For the remaining part of the proof we refer to Lemma 3 in [9]. □

We turn to the case where G is a spider. For a proof of the following lemma we can again refer to [9].

Lemma 29. *Let G be a spider with clique K. If G has thin legs then $\pi(G) = \left\lceil \frac{|K|}{2} \right\rceil$. If G has thick legs with $|K| > 2$ then G is Hamiltonian.*

We can again concentrate on the type 2-operation. We assume that the graph G is obtained from a separable p-connected graph $G_1 = (V_1, E_1)$ with separation (V_1^1, V_1^2) and a graph $G_2 = (V_2, E_2)$ by making every vertex of G_2 adjacent to every vertex of V_1^1. Since G_1 is a p-component, G_1 is either a spider or a graph with less than q vertices of which the characteristic is a splitgraph (Theorem 9 and Lemma 5).

Theorem 6. *For an arbitrary graph G exactly one of the following conditions is satisfied:*

- *G is disconnected*
- *$\overline{G}$ is disconnected*
- *There is a unique proper separable p-connected component H of G with separation (H_1, H_2) such that every vertex outside H is adjacent to all vertices in H_1 and to no vertex in H_2*
- *G is p-connected.*

In order to define the primeval decomposition tree we introduce three graph operations each acting on two graphs G_1 and G_2, corresponding with the first three cases of the structure theorem.

- For operation 0, G_1 and G_2 are arbitrary graphs. Operation 0 takes the disjoint union of G_1 and G_2.
- For operation 1, G_1 and G_2 are arbitrary. Operation 1 takes the complete sum of G_1 and G_2, i.e., every vertex of G_1 is made adjacent to every vertex of G_2.
- For operation 2, G_1 is *not* arbitrary: G_1 is a separable p-connected graph with separation (V_1^1, V_1^2). G_2 is an arbitrary graph. Operation 2 makes every vertex of G_2 adjacent to every vertex of V_1^1 and to no vertex of V_1^2.

These operations suggest a tree representation for arbitrary graphs which is unique up to isomorphism. The leaves of this rooted binary tree are exactly the p-components of the graph. The root corresponds with the input graph G. Internal nodes are labeled with integers $i \in \{0, 1, 2\}$ where an i-node means that the subgraph at this node is obtained by an i-operation applied to the two subgraphs corresponding with the two sons of the node.

2.1 Preliminaries on $(q, q-4)$-graphs

Babel and Olariu introduced the following classes of graphs [3].

Definition 7. A graph is a *(q, t)-graph* if no set of at most q vertices induces more than t distinct P_4's.

The aim of this subsection is to recall the charactization of the p-components of $(q, q-4)$-graphs presented in [2]. Notice that the $(4, 0)$-graphs are exactly the cographs.

Definition 8. A *spider* is a splitgraph consisting of a clique and an independent set of equal size at least two such that each vertex of the independent set has precisely one neighbor in the clique and each vertex of the clique has precisely one neighbor in the independent set, or it is the complement of such a graph.

Notice that the smallest spider is a P_4.

The main reason for the linear time solvability of the ranking problem for $(q, q-4)$-graphs for fixed q is that the p-components are of a very specific type. This is reflected by the following characterization found by Babel [1, 2].

Theorem 9. *Let G be p-connected.*

1. *If G is a $(5,1)$-graph then G is a spider.*
2. *If G is a $(7,3)$-graph then $|V| < 7$ or G is a spider.*
3. *If G is a $(q, q-4)$-graph, $q = 6$ or $q \geq 8$, then $|V| < q$.*

2.2 Preliminaries on vertex ranking

Definition 10. Let $G = (V, E)$ be a graph and let t be some integer. A *(vertex) t-ranking* is a numbering $c : V \to \{1, \ldots, t\}$ such that for every pair of vertices x and y with $c(x) = c(y)$ and for every path between x and y there is a vertex z on the path with $c(z) > c(x)$. The *(vertex) ranking number* of G denoted by $\chi_{\mathrm{r}}(G)$ is the smallest value t for which the graph G admits a t-ranking.

Notice that a ranking is a proper coloring of the graph. There are polynomial time algorithms for trees [15], cographs [16], AT-free graphs with a polynomial number of minimal separators [13], (including interval graphs, circular-arc graphs, permutation graphs and d-trapezoid graphs).

The vertex ranking problem is NP-complete in general [5], but it is an interesting observation that for each t the class of graphs with ranking number at most t is minor closed, thereby making the class recognizable in linear time.

A minimal a, b-separator S for non adjacent vertices a and b is a minimal set (with respect to inclusion) of vertices such that a and b are contained in different components of $G - S$. The following lemma appeared first in [7].

Lemma 11.

$$\chi_{\mathrm{r}}(G) = \min_{S} \left(|S| + \max_{C} \chi_{\mathrm{r}}(C) \right)$$

where the minimum is taken over all minimal separators S and the maximum is taken over all components C of $G - S$.

Closely related to the ranking problem is the minimum height elimination tree [6].

Definition 12. Let G be a graph. An *elimination tree* for G is a rooted tree T with vertex set V defined recursively as follows. If $V = \{x\}$ then T is the rooted tree containing only one vertex x. Otherwise choose any vertex $r \in V$ as the root of T. Let $C_1, \ldots, C_p$ be the connected components of $G - r$. For each component C_i let T_i be an elimination tree for $G[C_i]$. T is defined by making the roots r_i of T_i adjacent to r.

The *height* of a rooted tree is the maximal length of a path from the root to a leaf. The following result appeared in [5].

Lemma 13. *Let G be a connected graph. Let $h(G)$ be the minimum height of an elimination tree of G. Then $\chi_{\mathrm{r}}(G) = h(G) + 1$.*

4.2 $|V_1| < q$ and the characteristic of G_1 is a split graph

Let (K, S) be the separation of G_1. We show that in this case there is a linear time algorithm to compute the pathwidth of G. We consider two cases. First we consider triangulations into an interval graph where K is a clique.

Triangulations with K a clique. Let G^* be the graph obtained from G by making a clique of K. An optimal path-decomposition of $G^*[V_2 \cup K]$ can be made by adding K to every subset of an optimal path-decomposition for G_2.

The algorithm to compute the pathwidth of G^* works as follows. Compute all possible path-decompositions for $G^*[V_1]$ (i.e., G_1 with K turned into a clique). We can of course restrict to those path-decompositions with subsequent subsets not equal, hence the set of all possible path-decompositions is finite.

Let $\mathcal{P} = [X_1, \ldots, X_\ell]$ be such a path-decomposition. Add $X_0 = X_{\ell+1} = \emptyset$, Consider the subsequence $X_i, \ldots, X_j$ of subsets that contain the clique K and determine the minimum of $|(X_s \cap X_{s+1}) \setminus K|$ for $s = i-1, \ldots, j$. Let the minimum cardinality be attained for $(X_s \cap X_{s+1}) \setminus K$. Then a pathdecomposition for G^* can be obtained by making an optimal path-decomposition for G_2, adding $K \cup (X_s \cap X_{s+1})$ to every subset and putting this sequence of subsets between X_s and X_{s+1}. The width of this path-decomposition becomes $\max(width(\mathcal{P}), pw(G_2) + |K \cup (X_s \cap X_{s+1})|)$.

It is easy to see that there exists an optimal path-decomposition of this type for G^*, and it follows that we can compute the pathwidth of G^* in constant time.

Triangulations with K not a clique. Now consider the case where K is not a clique in some optimal interval graph embedding H of G. Let X be a clique module with non adjacent vertices x and y in H. The common neighbors of x and y must form a clique in H. Hence in H, $(K - X) \cup S_X \cup V_2$ is a clique, where S_X is the union of independent set modules which are neighbors of X. Hence an optimal path-decomposition for $G[K \cup S_X \cup V_2]$ with clique $(K - X) \cup S_X \cup V_2$ consists of an optimal path-decomposition of X with $S_X \cup V_2 \cup (K - X)$ added to every subset. Since X has bounded size, an optimal path-decomposition for X can be found in constant time.

We consider two cases. First the case where $|V_2| < q$. In that case the graph H has at most $2q-2$ vertices, and the pathwidth can be determined with exhaustive search.

Now assume $|V_2| \geq q$. We know that an optimal path-decomposition for $G[K \cup S_X \cup V_2]$ with clique $(K-X) \cup S_X \cup V_2$ has width at least $|V_2| - 1 \geq q - 1$. Hence we can simply add a subset containing $(K_X) \cup (S - S_X)$ without increasing the width to an optimal path-decomposition of $G[K \cup S_X \cup V_2]$ with clique $(K - X) \cup S_X \cup V_2$.

Varying over all possible clique modules X gives the pathwidth of G in this case.

Lemma 21. *Let G be the complete sum of $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$. Then*

$$pw(G) = \min(pw(G_1) + |V_2|, pw(G_2) + |V_1|)$$

Proof. Since we look for an interval supergraph H of G, H is chordal, and hence either $H[V_1]$ or $H[V_2]$ is a clique. If V_2 is a clique in H then the maximum clique number is equal to the clique number of $H[V_1]$ plus $|V_2|$. Since $H[V_1]$ is an interval supergraph of G_1, the clique number is at least $pw(G_1) + |V_2|$. □

We start again with the primeval tree. First we compute the pathwidth of the leaves. These leaves are either spiders or graphs of bounded size. If the graph is of bounded size, we can simply try all interval supergraphs to find the pathwidth. The following lemma leads to formulas for the pathwidth of spiders. For a proof of this lemma see [12].

Lemma 22. *Let G be a splitgraph with maximum size clique K. Then the pathwidth is either $|K|$ or $|K| - 1$. The pathwidth is $|K| - 1$ if and only if there are vertices x and y in K with $N(x) \cap N(y) \subseteq K$.*

We call a spider *thin* if every vertex of the independent set has one neighbor in the clique. We call the spider *thick* if every vertex of the independent set is non adjacent to one vertex of the clique. Notice that a P_4 is both thick and thin.

The following corollary takes care of those leaves that are spiders.

Corollary 23. *If G is a thin spider with $|K| > 1$ then $pw(G) = |K| - 1$. If G is a thick spider with $|K| > 2$ then $pw(G) = |K|$.*

We can now concentrate on the type 2-operation. We assume that the graph G is obtained from a separable p-connected graph $G_1 = (V_1, E_1)$ with separation (V_1^1, V_1^2) and a graph $G_2 = (V_2, E_2)$ by making every vertex of G_2 adjacent to every vertex of V_1^1. Since G_1 is a p-component, G_1 is either a spider or a graph with less than q vertices of which the characteristic is a splitgraph (Theorem 9 and Lemma 5).

4.1 G_1 is a spider

First we consider the case where G_1 is a spider. Let K be the clique and S be the independent set of G_1.

Lemma 24. *If $V_2 \neq \emptyset$ then $pw(G) = pw(G_2) + |K|$.*

Proof. Notice that $pw(G_2) + |K|$ is clearly a lower bound for the pathwidth of G since every interval graph embedding has at least a clique of this size plus one. We show that there is a path-decomposition of this width.

Make a path-decomposition of G_2 and add all vertices of K to every subset. Make a path-decomposition for G_1 by making a subset for each vertex $x \in S$ containing $K + x$. Put the path-decomposition for G_1 behind the one for G_2. □

For most of our proofs we will make use of the following result, which appeared first in [7]. A *triangulation* of a graph G is a chordal graph H with the same vertex set as G such that G is a subgraph of H (see, e.g., [12]).

Theorem 14. *For any graph G:*

$$\chi_r(G) = \{\omega(H) \mid H \text{ is a } P_4\text{-free triangulation of } G\}$$

Proof. First let G be connected with $\chi_r(G) = k$. There is an elimination tree T of G with height $h(G) = k - 1$. Define a graph H by making vertices x and y adjacent whenever they appear on a common path from the root of T to a leaf. The graph H is the intersection graph of subgraphs of a tree; for each vertex x take the subtree of T rooted at x as a subtree. Hence H is chordal.

Notice that G is a subgraph of H.

Finally we show that H is P_4-free. We prove that for all $W \subseteq V$ either $H[W]$ or $\overline{H}[W]$ is disconnected. This is obvious when W is a clique. If for every non adjacent pair $x, y \in W$ there is a common ancestor $z \in W$ in T, then there is a vertex $Z \in W$ which is an ancestor of all other vertices in W. Then Z is adjacent in H to all other vertices in W. Hence $\overline{H}[W]$ is disconnected. Otherwise there exist non adjacent vertices $x, y \in W$ without common ancestor in T. Then $H[W]$ is disconnected.

Notice that $\omega(H) = h(G) + 1 = \chi_r(G)$.

If G is disconnected, we construct P_4-free triangulations for all components as described above. □

Lemma 15. *Let G be the disjoint union of G_1 and G_2. Then*

$$\chi_r(G) = \max(\chi_r(G_1), \chi_r(G_2))$$

Lemma 16. *Let G be the complete sum of $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$.*

$$\chi_r(G) = \min(\chi_r(G_1) + |V_2|, \chi_r(G_2) + |V_1|)$$

Proof. Consider a P_4-free triangulation H of G. If two vertices of V_1 are not adjacent in H and two vertices of V_2 are not adjacent in H we obtain a C_4, which is a contradiction since H is chordal. Hence either V_1 or V_2 induces a complete subgraph in H. This proves the lemma. □

A proof of the following Lemma appeared also in [16].

Lemma 17. *Let G be a splitgraph with clique K and independent set S such that every vertex of K has at least one neighbor in S. Then $\chi_r(G) = |K| + 1$.*

Proof. Consider a P_4-free triangulation H of G. If two vertices x and y of S have private neighbors in K (i.e., if the neighborhoods of x and y are incomparable) in the graph H then either we obtain a C_4 or a P_4, contradiction that H is a chordal cograph. Hence the neighborhoods in K of the vertices of S in H are linearly ordered. In other words, there is an ordering $x_1, \ldots, x_s$ of the vertices in S such that $\bigcup_{\ell=1}^{i} N_G(x_i) \subseteq N_H(x_i) \cap K$. Since every vertex of K has a neighbor in S, this shows that the maximum clique of H is at least $|K| + 1$. □

3 A ranking algorithm for $(q, q-4)$-graphs

In this section we describe a linear time algorithm for the ranking problem on $(q, q-4)$-graphs for fixed q. This generalizes the early results of [16] for the ranking problem of cographs.

The first step of our algorithm is to build the primeval tree. This can be done in linear time (in fact Baumann [4] developed a linear time algorithm which computes a *homogeneous decomposition tree*, which is a primeval tree with an extra decomposition rule replacing the modules by single vertices).

We start computing the ranking numbers of the leaves of the tree. By Theorem 9 these leaves are either spiders or graphs of bounded size (with less than q vertices). In case the graph is a spider, the ranking number can be determined using Lemma 17. In case the leaf corresponds with a graph of bounded size, we can simply list all P_4-triangulations in constant time, and look for the one that minimizes the clique number.

Next consider an internal vertex of the primeval tree. If the label of this vertex is a 0-operation, the ranking number of the subgraph can be determined from the ranking number of the two sons using Lemma 15. If the label is a 1-operation, we can use Lemma 16.

In the rest of this section we concentrate on the type 2-operation. Hence we assume that the graph G is obtained from a separable p-connected graph $G_1 = (V_1, E_1)$ with separation (V_1^1, V_1^2) and a graph $G_2 = (V_2, E_2)$ by making every vertex of G_2 adjacent to every vertex of V_1^1. Since G_1 is a p-component, G_1 is either a spider or a graph with less than q vertices of which the characteristic is a splitgraph (Theorem 9 and Lemma 5).

3.1 G_1 is a spider

First consider the case where G_1 is a spider. Let K be the clique and S be the independent set of G_1.

Lemma 18. *If $V_2 \neq \emptyset$ then $\chi_r(G) = \chi_r(G_2) + |K|$.*

Proof. Give vertices of G_2 an optimal ranking with numbers $1, \ldots, \chi_r(G_2)$. Give vertices of K numbers $\chi_r(G_2)+1, \ldots, \chi_r(G_2)+|K|$. Finally, give all vertices of S number 1. This shows that $\chi_r(G) \leq \chi_r(G_2) + |K|$.

We now show that $\chi_r(G) \geq \chi_r(G_2) + |K|$. Consider a P_4-free triangulation H of G. Then the induced subgraph H_2 of H induced by V_2 is P_4-free. The clique number of H_2 is at least $\chi_r(G_2)$. Hence the clique number of H is at least $\chi_r(G_2) + |K|$. □

3.2 $|V_1| < q$ and the characteristic of G_1 is a split graph

Let (K, S) be the separation of G_1.

We consider two types of triangulations of G. First consider the graph G^* obtained from G by making a clique of K. I.e., consider P_4-free triangulations

H_1 of G_1 where K is made a clique. Let H_2 be a P_4-free triangulation of G_2. Notice that the operation 2 of H_1 and H_2 in this case is a chordal cograph also. Hence,

$$\chi_r(G^*) = \min_{H_1} \max(\omega(H_1), \chi_r(G_2) + |K|)$$

where the minimum is taken over all P_4-free triangulations of G_1 where K is made a clique.

Now consider the case of a P_4-free triangulation H where K is not a clique. Let x and y be two non adjacent vertices of some $X \subseteq K$ and let $S_X = S \cap N(X)$. Notice that every minimal x,y-separator contains at least the common neighborhood. Since in a chordal graph every minimal separator must be a clique we see that $(K-X) \cup S_X \cup V_2$ must be a clique in H. If we make $(K-X) \cup S_X \cup V_2$ into clique, and X into a chordal cograph with minimum clique number, it follows that the only P_4's left in the graph must have vertices in $S - S_x$ and $K - X$, where $K - X$ is a clique. Hence to obtain the optimal clique number of this type we have to determine the ranking number of every graph $G_1[(K-X) \cup (S - S_X)]$ with $K - X$ turned into a clique, for every clique module X. Since G_1 has at most q vertices this can be done in constant time.

Hence we obtain the following result.

Theorem 19. *For every value q there is a linear time algorithm to determine the vertex ranking number of a $(q, q-4)$-graph.*

4 A pathwidth algorithm for $(q, q-4)$-graphs

We show that the pathwidth problem can be solved in polynomial time for $(q, q-4)$-graphs for every fixed q. (For starlike graphs, and hence for those graphs of which the characteristic is a splitgraph, the problem is NP-complete [8].)

A *path-decomposition* of G is a sequence $[X_1, X_2, \ldots, X_\ell]$ of subsets of vertices, such that

- every vertex appears in some subset,
- the endvertices of every edge appear in some common subset, and
- For every vertex x, the subsets containing x appear consecutively in the sequence.

The pathwidth of a graph equals the minimum width of a path-decomposition, where the width of a path-decomposition is the maximum size of a subset minus one.

First we consider the two easy cases disjoint union and complete sum. The disjoint union is the most trivial case.

Lemma 20. *Let G be the disjoint union of G_1 and G_2. Then*

$$pw(G) = \max(pw(G_1), pw(G_2))$$