CHICAGO JOURNAL OF THEORETICAL COMPUTER SCIENCE 2011, Article 2, pages 1–11
http://cjtcs.cs.uchicago.edu/

SPECIAL ISSUE FOR CATS 2010

k-Cographs are Kruskalian*

Ling-Ju Hung Ton Kloks†

Received: March 27, 2010; published: May 6, 2011.

Abstract: A class of graphs is Kruskalian if Kruskal's theorem on a well-quasi-ordering of finite trees provides a finite characterization in terms of forbidden induced subgraphs. Let k be a natural number. A graph is a k-cograph if its vertices can be colored with colors from the set $\{1,\dots,k\}$ such that for every nontrivial subset of vertices W there exists a partition $\{W_1,W_2\}$ of W into nonempty subsets such that either no vertex of W_1 is adjacent to a vertex of W_2 or, such that there exists a permutation $\pi \in S_k$ such that vertices with color i in W_1 are adjacent exactly to the vertices with color $\pi(i)$ in W_2, for all $i \in \{1,\dots,k\}$. We prove that k-cographs are Kruskalian. We show that k-cographs have bounded rankwidth and that they can be recognized in $O(n^3)$ time.

Key words and phrases: cographs, k-cographs, rankwidth

1 Introduction

Progress on the study of tree-decompositions of graphs, such as rankwidth and cliquewidth decompositions, makes it a point of interest to investigate classes of graphs for which the cutmatrices in the decomposition take on a certain shape. It would be nice if one could tell by the shape of the cutmatrices whether a class of graphs has a finite characterization in terms of forbidden induced subgraphs. We studied these graph classes to gain insight into the structures of classes of graphs of bounded rankwidth.

To make matters clear, we need a few definitions.

*This research was supported by the National Science Council of Taiwan under grants NSC 98–2218–E–194–004 and NSC 98–2811–E–194–006

†Ton Kloks is supported by the National Science Council of Taiwan, under grants NSC 99–2218–E–007–016 and NSC 99–2811–E–007–044.

DOI: 10.4086/cjtcs.2011.002

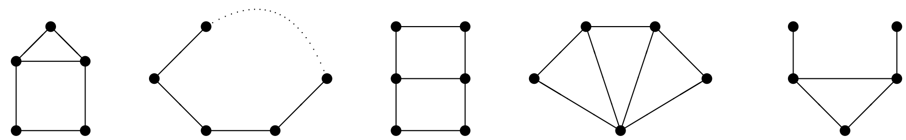

Figure 1: A house, a hole, a domino, a gem, and a bull. A graph is distance hereditary if it has no induced house, hole, domino, or gem. The co-gem is the complement of the gem; that is the union of a P_4 with a single vertex.

Definition 1.1. A *tree-decomposition* of a graph G is a pair (T, f) where T is a ternary tree, i.e., every internal node of T is of degree three and where f is a bijection from the leaves of T to the vertices of G.

We call the edges of a tree T in a tree-decomposition (T, f) the lines of T and the vertices of T the points of T or nodes of T.

Definition 1.2. Let (T, f) be a tree-decomposition of a graph $G = (V,E)$. Let e be a line in T. Consider the two sets A and B where A is the set of leaves in one subtree of $T - e$ and B is the set of leaves in the other subtree of $T - e$. The *cutmatrix* M_e is the submatrix of the adjacency matrix of G with rows indexed by the vertices of A and columns indexed by the vertices of B.

Definition 1.3 ([14])**.** A graph has *rankwidth* k if it has a tree-decomposition such that every cutmatrix has binary rank at most k. Here the binary rank is the rank of the matrix under the field $GF(2)$.

Definition 1.4. Consider a $0,1$-matrix M. Let M' be the maximal submatrix of M with no two rows equal and no two columns equal. The *shape of* M is the class of matrices equivalent to M' under permuting rows, permuting columns, and taking the transpose.

A graph is a cograph if it has no induced P_4, that is, a path with 4 vertices. It follows from a characterization by Corneil *et al.* that a graph is a cograph if and only if it has a tree-decomposition such every cutmatrix is shape-equivalent to a submatrix of $\begin{pmatrix} 1 & 0 \end{pmatrix}$ [5].

Another example of a class of graphs that is characterized by shapes of cutmatrices is the class of distance-hereditary graphs, *i.e.*, the class of graphs with rankwidth at most one [10, 14]. Distance-hereditary graphs are those graphs that have a tree-decomposition such that every cutmatrix has a shape equivalent to a submatrix of $\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$ [4]. An obvious consequence is that the complements of the distance-hereditary graphs have a tree-decomposition with every cutmatrix equivalent to $\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$.

A third example is the class of graphs that have a tree-decomposition such that every cutmatrix is equivalent to a submatrix of $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. This turns out to be the class of graphs without C_5, bull, gem, or co-gem [3, 1, 12].

A basic difference between cographs and distance-hereditary graphs is that the set of forbidden induced subgraphs is finite for the class of cographs while it is infinite for the class of distance-hereditary graphs. Note that distance-hereditary graphs have a finite characterization in terms of *vertex-minors* [14].

Definition 1.5. Let k be a natural number. A graph $G = (V,E)$ is a k-cograph if there exists a coloring of the vertices with colors from $\{1, \dots, k\}$ such that for every subset of vertices $W \subseteq V$ with $|W| \geq 2$ there exists a partition $\{W_1, W_2\}$ of W into two nonempty subsets such that either

i. no vertex of W_1 is adjacent to a vertex of W_2, or

ii. there exists a permutation σ of the colors such that vertices of color i in W_1 are adjacent exactly to the vertices of color $\sigma(i)$ in W_2, for all $i \in \{1,\dots,k\}$. That is, for every vertex $x \in W_1$ of color i,
$$N(x) \cap W_2 = \{y \in W_2 \mid y \text{ is of color } \sigma(i)\}.$$

In this paper we look at a class of graphs for which Kruskal's theorem provides a proof that they can be characterized by a finite collection of forbidden induced subgraphs.

Note that, for $k = 1$, the 1-cographs are the ordinary cographs; this is Gallai's characterization [7]. Note that k-cographs are also $(k+1)$-cographs; we may simply leave some color class empty. If C is a nonempty color class, *i.e.*, a nonempty set of vertices with the same color, then the subgraph induced by C is a cograph.

One of our main results is that for each k there exists a finite characterization for the class of k-cographs in terms of forbidden induced subgraphs. We prove this in Section 3. We end this section with some of our notational customs and with one significant observation.

For two sets A and B we write $A+B$ and $A-B$ instead of $A \cup B$ and $A \setminus B$. We write $A \subseteq B$ if A is a subset of B with possible equality and we write $A \subset B$ if A is a subset of B and $A \neq B$. For a set A and an element x we write $A+x$ instead of $A+\{x\}$ and $A-x$ instead of $A-\{x\}$. It will be clear from the context when x is an element instead of a set.

A graph G is a pair $G = (V,E)$ where V is a *finite*, nonempty set, of which the elements are called the vertices of G, and where E is a set of two-element subsets of V, of which the elements are called the edges of G. A graph consisting of a single vertex is called *trivial*. We denote edges of a graph as (x,y) and we call x and y the endvertices of the edge. For a vertex x we write $N(x)$ for its set of neighbors and we write $N[x] = N(x)+x$ for the closed neighborhood of x. For a subset $W \subseteq V$ we write $N(W) = \bigcup_{x\in W} N(x) - W$ for its neighborhood and we write $N[W] = N(W)+W$ for its closed neighborhood. Usually we use $n = |V|$ to denote the number of vertices of G and we use $m = |E|$ to denote the number of edges of G.

For a graph $G = (V,E)$ and a subset $S \subseteq V$ of vertices we write $G[S]$ for the subgraph *induced* by S, that is, the graph with S as its set of vertices and with those edges of E that have both endvertices in S. For a subset $W \subseteq V$ we write $G-W$ for the graph $G[V-W]$. For a vertex x we write $G-x$ rather than $G-\{x\}$. We usually denote graph classes by calligraphic capitals.

A *module* is a set M of vertices such that
$$x,y \in M \quad \Rightarrow \quad N(x)-M = N(y)-M.$$
A *twin* is a module with two vertices. The twin is *false* if the two vertices are not adjacent and it is *true* if the two vertices are adjacent.

Cographs can be characterized as those graphs for which every nontrivial induced subgraph has a twin [2].

Definition 1.6. 'Creating a twin' of a vertex x in a graph G is the operation of adding a new vertex x' and adding edges incident with x' such that x' and x become false twins or true twins.

We show that the class of k-cographs is closed under creating twins.

Lemma 1.7. *Assume that G is a k-cograph and let G' be obtained from G by creating a twin. Then G' is also a k-cograph.*

Proof. Let G' be obtained by creating a twin x' of a vertex x in G. Consider a k-coloring as in Definition 1.5 and give x' the same color as x. Let W be a subset of vertices of G' with at least two elements. If $W = \{x, x'\}$ then let $W_1 = \{x\}$ and let $W_2 = \{x'\}$. If x and x' are false twins, then no vertex of W_1 is adjacent to a vertex of W_2. If vertices x and x' are true twins in G', then consider any permutation σ that maps the color of x onto itself, for example the identity map. Then a vertex with color i from W_1 is adjacent to exactly the vertices with color $\sigma(i)$ of W_2 for all $i = 1, \dots, k$ because x and x' are adjacent. Thus the condition of Definition 1.5 is satisfied.
Now assume that W contains the vertex x' but not the vertex x. Then consider the same subset of vertices in the graph G but with x' replaced by x. By definition there exists a partition $\{W_1, W_2\}$ satisfying the conditions of Definition 1.5. Assume $x \in W_1$. Then consider the same partition of W in G' but with the vertex x in W_1 replaced by x'. Since x and x' have the same color and they are twins, the condition of Definition 1.5 is satisfied.
If W contains x but not x' then there is nothing to prove; we can take the same partition of W in G' as in G. Finally, if W contains both x and x' and at least one other vertex, then remove the vertex x' from the set. In the implied partition $\{W_1, W_2\}$ assume that $x \in W_1$. Add the vertex x' also to W_1. The claim follows, and this proves the lemma. □

2 k-Cotrees

In this section we show that for any natural number k, k-cographs can be recognized in $O(n^3)$ time.

A k-cotree for a graph $G = (V, E)$ is a pair (T, f) where T is a rooted binary tree and f is a 1-1 mapping from the leaves of T to V; the set of vertices of G. Each leaf is labeled with a color from $\{1, \dots, k\}$. Each internal node of T, including the root, is labeled either as a union node, or by some permutation σ of $\{1, \dots, k\}$. If an internal node of T is labeled as a union node then in G the vertices that are mapped to its left subtree are not adjacent to the vertices that are mapped to its right subtree. If an internal node is labeled with a permutation σ then in G the vertices of color i that are mapped to leaves in its left subtree are adjacent to the vertices of color $\sigma(i)$ that are mapped to its right subtree for all $i \in \{1, ..., k\}$. This implies that two vertices x and y in G are adjacent if and only if the lowest common ancestor is labeled with a permutation σ that matches the colors of x and y.

The following lemma echoes the classic [5].

Lemma 2.1. *A graph is a k-cograph if and only if it has a k-cotree representation.*

Proof. First assume that G is a k-cograph. Consider a coloring of the vertices as prescribed in Definition 1.5. We construct a k-cotree as follows.
Let $\{V_1, V_2\}$ be a partition of V such that either no vertex in V_1 is adjacent to a vertex of V_2 or there exists a permutation σ of the colors such that vertex of color i is V_1 are adjacent exactly to the vertices of color $\sigma(i)$ in V_2 for $i \in \{1, \dots, k\}$. Construct a root node r. If no vertex of V_1 is adjacent to a vertex of V_2, then label r as a union node. Otherwise, label r with the permutation that matches the color classes of V_1 with the color classes of V_2. If V_1, or if V_2 consists of a single node then create a leaf and label it with the color

of that vertex. Otherwise, recursively construct k-cotrees for $G[V_1]$ and for $G[V_2]$ and make the roots of those two trees adjacent to r. By construction, two vertices x and y are adjacent in G if and only if their lowest common ancestor is labeled with a permutation that matches their colors.

Consider a graph $G = (V,E)$ and let (T, f) be a binary k-cotree that represents G. The labels at the leaves provide a coloring of the vertices. Let $W \subseteq V$ be a nontrivial subset of vertices and consider the lowest common ancestor p in T of the leaves that are mapped to the vertices of W. The two subtrees at p partition W into two nonempty subsets W_1 and W_2 and the permutation σ at p makes vertices with color i in the left subtree adjacent to the vertices with color $\sigma(i)$ in the right subtree. This shows that the graph is a k-cograph. □

Lemma 2.2. *Let $G = (V,E)$ be a k-cotree and let (T, f) be a k-cotree representation of G. Consider the tree-decomposition (T', f) obtained from (T, f) by removing the root and by making the two children of the root adjacent in T'. Then every cutmatrix of (T', f) is shape-equivalent to a submatrix of $\begin{pmatrix} I_k & 0 \end{pmatrix}$, where I_k is the $k \times k$ identity matrix.*

Proof. Let p be a child of the root r of T. Let e be the line in T that connects p to r. If r is labeled as a union node, then the cutmatrix of e is a zero matrix. Otherwise, assume that r is labeled with a permutation σ. Then the cutmatrix at e is equivalent to a submatrix of the permutation matrix that matches vertices of color i in the subtree at p to vertices with color $\sigma(i)$ in the other subtree. Note that some color classes may be empty.

Now consider an arbitrary node q, not equal to r, and let q_1 and q_2 be two children of q in T. Let Q be the set of vertices of G mapped to leaves in the subtree of q. Let $Q_1 \subset Q$ be the set of leaves in the subtree at q_1. We consider the cutmatrix of the line e' that connects q_1 with q, with rows indexed by the vertices of Q_1.

Let e be the line that connects q with its parent. By induction, we may assume that vertices of Q that have the same color, have the same neighbors in the rest of the graph and that vertices of Q that have different colors have no common neighbors in the rest of the graph. Consider a subset of vertices of Q_1 with the same color. They have the same neighbors in $V - Q$. By definition of a k-cotree they have the same neighbors in Q_2 and thus they have the same neighbors in $V - Q_1$. Furthermore, vertices in Q_1 with different colors have no common neighbor in $V - Q$ and they have also no common neighbor in Q_2. Summarizing, vertices of Q_1 with the same color have the same neighbors in $V - Q_1$ and vertices of Q_1 with different colors have no common neighbors in $V - Q_1$. Since there are only k colors, this shows that the shape of the cutmatrix at e' is equivalent to a submatrix of $\begin{pmatrix} I_k & 0 \end{pmatrix}$. This proves the lemma. □

Lemma 2.3. *k-Cographs have rankwidth at most k.*

Proof. By definition, a graph has rankwidth k if it has a tree-decomposition such that every cutmatrix has binary rank at most k. By Lemma 2.2, k-cographs have such a tree-decomposition and it can be found in $O(n^3)$ time [9]. □

Theorem 2.4. *For every natural number k, there exists an $O(n^3)$ algorithm that recognizes k-cographs.*

Proof. The definition of a k-cograph can be expressed in monadic second-order logic. It is proved in [14] that every graph problem that is specified by a monadic second-order logic sentence without edge-set quantification can be solved in $O(n^3)$ time on graphs of bounded rankwidth. Since k-cographs have bounded rankwidth, this proves the claim. $\square$

In a recent paper we introduced the concept of simple-width [12]. A graph has simple-width k if it has a tree-decomposition such that every cutmatrix is shape-equivalent to some matrix with at most k rows and at most k columns. It is easy to see that

$$r(G) \leq s(G) \leq 2^{r(G)},$$

where $r(G)$ is the rankwidth of G and $s(G)$ is the simple-width of G. The benefit of simple-width lies in the fact that characterizations of graph classes, defined in terms of permitted shapes of cutmatrices, are easier to obtain. In the mentioned paper we characterize graphs with simple-width 2. Note that no characterization is known of graphs with rankwidth at most two.

3 Forbidden induced subgraphs

In this section we show that for every natural number k, k-cographs can be characterized by a finite set of forbidden induced subgraphs.

When a coloring that satisfies Definition 1.5 is a part of the input, then we call the k-cograph *partitioned*.

Theorem 3.1. *Let k be a natural number. Partitioned k-probe cographs are well-quasi-ordered by the induced subgraph relation.*

Proof. A cotree is a binary tree with a bijection from the leaves to the vertices of the graph and internal nodes labeled as join- or union-operators [5]. Two vertices are adjacent in the graph if and only if their lowest common ancestor is a join-node. Kruskal's theorem [13, 8] states that trees, with points labeled by a well-quasi-ordering, are well-quasi-ordered with respect to their lowest common ancestor embedding. Pouzet observed that this implies that cographs are well-quasi-ordered by the induced subgraph relation [15] (see also [6, 16]). For partitioned k-cographs we equip each leaf with a label that is a color from $\{1,\ldots,k\}$. Each internal node receives a label which is either a union label or a permutation of the colors. Two vertices are adjacent if their lowest common ancestor is a permutation-node and if their colors are matched by the permutation at that node. Kruskal's theorem implies the claim. $\square$

It is interesting to notice that Theorem 3.1 implies that k-cographs do not have long induced paths. This implication can be deduced as follows. Let $[P_1, P_2, \ldots]$ be an infinite sequence of k-colored paths of increasing length and assume that they are all partitioned k-cographs. Construct graphs P_i' by creating a false twin with the same color of each of the two endpoints of P_i. Then each P_i' is also a partitioned k-colored cograph since this class of graphs is closed under creating false twins. The sequence $[P_1', P_2', \ldots]$ is an infinite sequence and P_i' is not an induced subgraph of P_j' as long as $i \neq j$. This contradicts Theorem 3.1.

Theorem 3.2. *Let k be a natural number. The class of partitioned k-cographs can be characterized by a finite set of forbidden induced, colored subgraphs.*

Proof. Consider a sequence $[G_1, G_2, \ldots]$ of k-colored graphs which are not partitioned k-cographs. Assume that for each vertex x in $G_i = (V_i, E_i)$ the subgraph induced by $V_i - x$ is an induced k-cograph $(i = 1, 2, \ldots)$. Assume also that each G_i is equipped with a 'root' r_i which is a vertex of G_i and assume that all roots $r_1, r_2, \ldots$ have the same color.
For $i = 1, 2, \ldots$, consider k-cotrees of $G_i - r_i$ as in Theorem 3.1. Extend the labels at the leaves with an additional label-entry 0 or 1 that indicates whether the vertex, that is mapped to the leaf, is adjacent to r_i or not. Consider the well-quasi-ordering of these labeled trees by the lowest-common-ancestor ordering. Kruskal's theorem implies that there must exist $i < j$ such that G_i is an induced subgraph of G_j. This proves the theorem. □

We close this section with a similar proof for the unpartitioned case.

Theorem 3.3. *Let k be a natural number. The class of k-cographs is characterized by a finite set of forbidden induced subgraphs.*

Proof. Let $[G_1, G_2, \ldots]$ be an infinite sequence of graphs that are not in the class of k-cographs. Assume furthermore that each G_i satisfies the property that $G_i - x$ is a k-cograph for any vertex x of G_i. Single out one vertex r_i in each G_i and label each vertex of $G_i - r_i$ with a 0 or a 1, depending whether it is adjacent to r_i or not. Consider furthermore a (good) k-coloring for each $G_i - r_i$. Thus we obtain a sequence of k-colored graphs $[G_1 - r_1, G_2 - r_2, \ldots]$ with an additional $0/1$-label at each vertex. By Kruskal's theorem there must exist $i < j$ such that the labeled graph $G_i - r_i$ is an induced subgraph of $G_j - r_j$. Since we may assume that no two graphs G_i and G_j are equal, this proves the theorem. □

Note that Theorem 3.3 gives an alternative proof for the fact that k-cographs can be recognized in $O(n^3)$ time. Let $\mathcal{O}_k$ be the (finite) set of forbidden induced subgraphs for k-cographs. Thus a graph is a k-cograph if and only if it has no element of $\mathcal{O}_k$ as an induced subgraph. Because $\mathcal{O}_k$ is finite, this is a monadic second-order characterization. Note however, that in this case the proof is nonconstructive; Kruskal's theorem does not provide the set $\mathcal{O}_k$.

4 2-Cographs

Let's satisfy some of our cravings for practicalities; some discouraging news to start with. Forbidden induced subgraphs for 2-cographs include the gem, C_5, C_7, C_8, C_9, C_{10}, and P_{10}, and the graphs depicted in Figure 2. We think that the list is complete, but we do not have a proof to show that.

We find it surprising that the list is so long, already for the case of 2-cographs. One good, and very useful thing that comes out of it is this:

Corollary 4.1. *2-Cographs are perfect.*

Although we proved the existence of an $O(n^3)$ recognition algorithm for k-cographs, such an algorithm is hard to obtain in practice. We end this section with the description of an easy $O(n^4)$ recognition algorithm for 2-cographs.

Theorem 4.2. *There exists an $O(n^4)$ algorithm for the recognition of 2-cographs.*

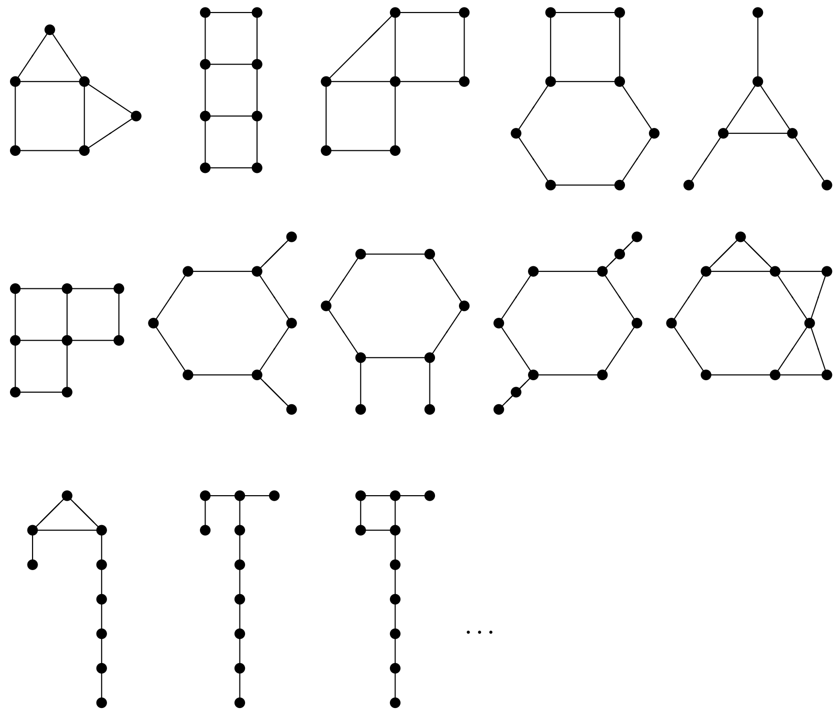

Figure 2: Some forbidden induced subgraphs for 2-cographs. The gem, C_5, C_7, C_8, C_9, C_{10}, and P_{10} are also forbidden but are not depicted.

Proof. A 2-cograph G can be seen as a 'P_4-free structure,' which we define next.
Consider a 2-cotree (T, f) with leaves colored black and white. If G is nontrivial, then there exist two leaves with a common ancestor in T. Either the two leaves are twins, or they are pairs of vertices x and y with different colors; that is, x and y have no common neighbors.
Consider a subtree with three or four leaves. If G has enough vertices than such a subtree exists. In the case of a subtree with 3 leaves, one pair of leaves have a common ancestor q and there is a point p which has two children, namely q and the third leaf. In the second case there are two pairs of leaves, each with a common ancestor. Assume that G has no twins. Consider the first case. Let x and y be the two leaves with a common ancestor and let z be the third leaf. Then x and y have different colors, thus they have no common neighbors outside $\{x, y\}$. Either the vertex z has the same neighbors as x, or it has the same neighbors as y outside $\{x, y, z\}$. A similar situation occurs when the subtree has 4 leaves.
Define a structure with two kinds of 'points.' The first type of points are the pairs of vertices with different colors that have a common ancestor in T. The second type of points are the single vertices that do not occur in points of the first type.
The structure is defined by the property that every sub-structure, induced by a subset of the points, has a 'structure-twin.' Such a twin in the structure is either

(i) a pair of points that are vertices of G that are twins in G; or

(ii) two points; one a pair $\{x, y\}$ and the other a vertex z such that either z and $\{x, y\}$ are disjoint or such that z is adjacent to exactly one of x and y. Furthermore, either x and z or y and z have the same neighbors outside $\{x, y, z\}$; or

(iii) two points; both pairs, say $\{x, y\}$ and $\{p, q\}$. Either they are disjoint, or they are pairwise connected. Furthermore, either $\{x, p\}$ and $\{y, q\}$, or $\{x, q\}$ and $\{y, p\}$ have the same neighbors outside $\{x, y, p, q\}$, and these two neighborhoods are disjoint.

Obviously, a graph is a 2-cograph if and only if there is a corresponding 'P_4-free structure,' that is a structure with no induced P_4 of points. Two points are adjacent in the structure if there is an induced subset of points for which the points are adjacent structure-twins, that is, they are structure-twins and there is at least one edge in G between the corresponding subsets of vertices. The recognition problem for 2-cographs boils down to the identification of the points of a corresponding structure.
Our algorithm builds a P_4-free structure, if it exists. It mimics the building of a 1-cotree by grouping twins; in that algorithm one repeatedly searches for a twin, and then removes one of the two vertices from the graph.
We start with a collection of feasible subtrees consisting of all single vertices and of all pairs of colored vertices that have no common neighbors. Each subtree is represented, either by a point which is a pair of vertices, or by a point which is a single vertex. For each subtree the algorithm maintains a list of leaves. Two subtrees T_1 and T_2 merge if the points are structure-twins. Consider two subtrees represented by structure-twins. Consider the graph G induced by the vertices in the two points and the vertices that do not appear in the leaves of T_1 and T_2 nor in the lists of vertices that are 'covered' (see below).

a. If T_1 and T_2 are twins, and if T_1 has two colors and T_2 only one, then we say that T_1 'covers' T_2. For each subtree the algorithm maintains a list of vertices that are covered by it. In this case the set of leaves of T_2 is added to the list of vertices covered by T_1. In this case no subtree is deleted from the list.

b. If both points that represent T_1 and T_2 are single vertices that are twins in G then delete one of the two subtrees from the list. Make a union of the list of leaves.

c. If two subtrees are structure-twins such that both have pairs of vertices with different colors, then delete one of them from the list. Make a union of the list of leaves and of the list of vertices that are covered.

Note that a 2-cotree is a subtree T_i such that every vertex of G appears as a leaf of T_i, or is covered by T_i.
To prove the correctness one may observe that, in the first case there exists a 2-cotree, with T_1 and T_2 as subtrees, if and only if there exists a 2-cotree that contains T_1 as a subtree. In the second and third case, one easily observes that there exists a 2-cotree that contains T_1 and T_2, if and only if there exists a 2-cotree that contains one of them as a subtree.
There are $O(n^2)$ basic building blocks, and since a 2-cotree has $O(n)$ nodes, the algorithm builds a 2-cotree in $O(n^3)$ steps or decides that no such tree can exist because there are no more possible mergers. The update of the list of vertices that are covered by a subtree takes $O(n)$ time in each step. This proves the theorem. $\square$

5 Concluding remark

Chang *et al.* show that distance-hereditary graphs can be regarded as tree-decompositions where subtrees are equipped with a 'twinset.' The adjacencies between subtrees are joins or unions of the twinsets [4]. The twinsets are cographs. As mentioned at the beginning of this paper, cographs have tree-decompositions with cutmatrices equivalent to submatrices of $\begin{pmatrix} 1 & 0 \end{pmatrix}$ and distance-hereditary graphs have tree-decompositions with cutmatrices equivalent to submatrices of $\left(\begin{smallmatrix} 1 & 0 \\ 0 & 0 \end{smallmatrix}\right)$.

Consider the class of graphs that have a tree-decomposition such that every cutmatrix has a shape equivalent with some submatrix of

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

This class of graphs could be defined as graphs built from subtrees that are equipped with a '2-twinset.' These 2-twinsets are graphs with a structure similar to the structure of the 2-cographs [11].

Finally, to characterize the graphs with rankwidth at most 2, one needs to consider tree-decompositions where cutmatrices have shapes equivalent to a submatrix of

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

This class contains the 'probe distance-hereditary graphs.' A graph is probe-distance hereditary if it is a distance-hereditary graph minus the edge-set of an arbitrary induced subgraph [1].

References

[1] CHANDLER D. B., M.-S. CHANG, T. KLOKS, AND S.-L. PENG: Probe graphs. Manuscript, 2009. 2, 10

[2] A. BRANDSTÄDT, V. B. LE, AND J. P. SPINRAD: *Graph classes: A survey*. Volume 1999 of *SIAM Monographs on Discrete Mathematics and Applications*. Philadelphia. 3

[3] P. J. CAMERON: Two-graphs and trees. *Discrete Mathematics*, 127:63–74, 1994. 2

[4] M.-S. CHANG, S.-Y. HSIEH, AND G.-H. CHEN: Dynamic programming on distance-hereditary graphs. In *Proceedings of the* 8^{th} *International Symposium on Algorithms and Computations*, volume 1350 of *LNCS*, pp. 344–353, 1997. 2, 10

[5] D. G. CORNEIL, H. LERCHS, AND L. S. BURLINGHAM: Complement reducible graphs. *Discrete Applied Mathematics*, 3:163–174, 1981. 2, 4, 6

[6] P. DAMASCHKE: Induced subgraphs and well-quasi-ordering. *Journal of Graph Theory*, 14:426–435, 1990. 6

[7] T. GALLAI: Transitiv orientierbare graphen. *Acta Math. Acad. Sci. Hungar.*, 18:25–66, 1967. 3

[8] G. HIGMAN: Ordering by divisibility in abstract algebras. *Proc. London Math. Soc.*, 2:326–226, 1952. 6

[9] P. HLINĚNÝ AND S. OUM: Finding branch-decompositions and rank-decompositions. *SIAM J. Comput.*, 38:1012–1032, 2008. 5

[10] E. HOWORKA: A characterization of distance-hereditary graphs. *Quarterly Journal of Mathematics*, 28:417–420, 1977. 2

[11] L.-J. HUNG AND T. KLOKS: Classifying rankwidth; **k**-dh-graphs. 6072:195–203, 2010. 10

[12] L.-J. HUNG AND T. KLOKS: On some simple widths. In *Proceedings of WALCOM 2010*, volume 5942 of *LNCS*, pp. 204–215, 2010. 2, 6

[13] J. KRUSKAL: Well-quasi-ordering, the tree theorem, and vazsonyi's conjecture. *Transactions of the American Mathematical Society*, 95:210–225, 1960. 6

[14] S. OUM: *Graphs of bounded rank-width*. PhD thesis, Princeton University, 2005. 2, 5

[15] M. POUZET: Applications of well-quasi-ordering and better-quasi-ordering. In I. RIVAL, editor, *Graphs and Orders*, pp. 503–519. D. Reidel Publishing Company, 1985. 6

[16] S. THOMASSÉ: On better-quasi-ordering countable series-parallel orders. *Transactions of the American Mathematical Society*, 352:2491–2505, 2000. 6

AUTHORS

Ling-Ju Hung
Department of Computer Science and Information Engineering
National Chung Cheng University
Min-Hsiung 621, Chia-Yi, Taiwan
hunglc@cs.ccu.edu.tw

Ton Kloks