1992

A Tree Representation for P4-Sparse Graphs

B. Jamison
Old Dominion University

Stephan Olariu
Old Dominion University

Original Publication Citation

Jamison, B., & Olariu, S. (1992). A tree-representation for P_4-sparse graphs. *Discrete Applied Mathematics, 35*(2), 115-129.
doi:10.1016/0166-218x(92)90036-a

Discrete Applied Mathematics 35 (1992) 115-129
North-Holland

A tree representation for P_4-sparse graphs*

B. Jamison and S. Olariu

Department of Computer Science, Old Dominion University, Norfolk, VA 23529-0162, USA

Received 19 December 1988
Revised 12 December 1989

Abstract

Jamison, B. and S. Olariu, A tree representation for P_4-sparse graphs, Discrete Applied Mathematics 35 (1992) 115-129.

A graph G is P_4-sparse if no set of five vertices in G induces more than one chordless path of length three. P_4-sparse graphs generalize both the class of cographs and the class of P_4-reducible graphs. We give several characterizations for P_4-sparse graphs and show that they can be constructed from single-vertex graphs by a finite sequence of operations. Our characterization implies that the P_4-sparse graphs admit a tree representation unique up to isomorphism. Furthermore, this tree representation can be obtained in polynomial time.

1. Introduction

One of the most promising paradigms for the algorithmic study of a class Γ of graphs involves associating with every graph G in Γ a *unique* rooted tree $T(G)$ whose leaves are elements of G (e.g. vertices, edges, maximal cliques, maximal stable sets, cutsets) and whose internal nodes correspond to certain graph operations. If $T(G)$ can be obtained *efficiently* (i.e., in polynomial time in the size of the graph G) and if its leaves can be tested efficiently for isomorphism, then the graph isomorphism problem (which is still open for arbitrary graphs) can be solved efficiently for graphs in Γ, since it reduces to tree isomorphism.

An early example in this direction is the class of *cographs* discovered and investigated independently by various researchers [6-8,12,14,17-19]. As it turns out, the cographs are precisely the graphs containing no chordless path on four vertices (termed a P_4). In addition, Lerchs [14] showed that with every cograph G one can associate a unique rooted tree $T(G)$, called the *cotree* of G, whose leaves are precise-

* This work was supported by the National Science Foundation under grant CCR-8909996.

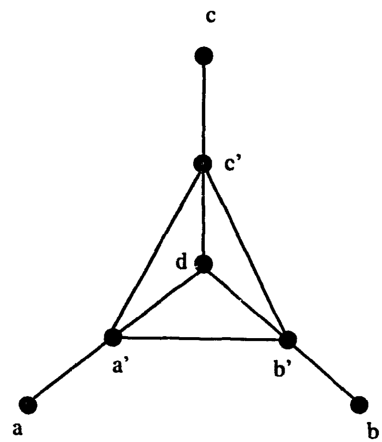

Fig. 1.

ly the vertices of G; the internal nodes are labeled by 0 or 1 in such a way that two vertices x, y are adjacent in G if and only if their lowest common ancestor in $T(G)$ is labeled 1.

Unique tree representations have been obtained for several other classes of graphs including the interval graphs [4], maximal outerplanar graphs [3], TSP digraphs [13], and P_4-reducible graphs [11].

In his doctoral dissertation, Hoàng [9] introduced the class of P_4-sparse graphs: these are graphs for which every set of five vertices induces at most one P_4. At the same time he gave a number characterizations of P_4-sparse graphs, and showed that P_4-sparse graphs are perfect in the sense of Berge [1] (a graph G is *perfect* if for every induced subgraph H of G, the chromatic number of H equals the largest number of pairwise adjacent vertices in H), and even *perfectly orderable* in the sense of Chvátal [5]. It is easy to see that the P_4-sparse graphs strictly contain all the cographs. Jamison and Olariu [11] defined the P_4-reducible graphs as graphs in which no vertex belongs to more than one P_4. The P_4-reducible graphs are a natural generalization of the cographs and find applications in areas such as scheduling and clustering. Trivially, every P_4-reducible graph is also P_4-sparse, but not conversely: the graph featured in Fig. 1 is P_4-sparse, but not P_4-reducible.

Our main result gives a constructive characterization of the P_4-sparse graphs. To anticipate, all the P_4-sparse graphs turn out to be constructible from single vertices by a finite sequence involving three graph operations. Our characterization implies that P_4-sparse graphs are uniquely tree representable. In turn, this tree representation can be used to provide efficient solutions to the four classical graph optimization problems: given a graph G, this involves finding the largest number $\omega(G)$ of pairwise adjacent vertices in G, the largest number $\alpha(G)$ of pairwise adjacent vertices in the complement $\bar{G}$ of G, the smallest number $\chi(G)$ of colours assigned to the vertices of G in such a way that adjacent vertices receive distinct colours, and the smallest number $\theta(G)$ of colours needed to colour the complement $\bar{G}$ of G.

Finally, the problem of finding a largest induced P_4-free graph of an arbitrary graph G is known to be intractable (see Corneil et al. [8]). For P_4-sparse graphs, it turns out that the solution is provided by a surprisingly simple greedy algorithm in polynomial time. In fact, Theorem 2.19 shows that the P_4-sparse graphs coincide with a certain class of graphs for which this greedy algorithm is guaranteed to produce a graph unique up to isomorphism.

2. The results

All the graphs in this work are finite, with no loops or multiple edges. In addition to standard graph-theoretical terminology compatible with Berge [2], we use some new terms that we are about to define.

Let $G=(V,E)$ be an arbitrary graph. For a vertex x of G, we let $N_G(x)$ denote the set of all the vertices of G which are adjacent to x: we assume adjacency to be nonreflexive, and so $x \notin N_G(x)$; we let $d_G(w)$ stand for $|N_G(w)|$. If S is a subset of the vertex set of G, we let G_S stand for the subgraph of G induced by S. If a vertex x is nonadjacent to a vertex y, we shall say that x *misses* y. (Similarly, y *misses* x.)

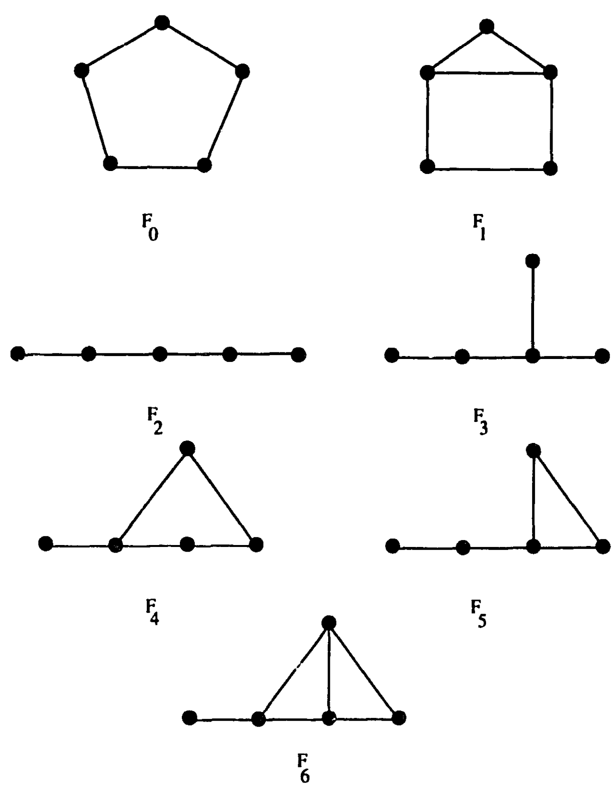

Fig. 2.

A vertex z is said to *distinguish* between vertices u and v, whenever z misses precisely one of u, v. A *clique* is a set of pairwise adjacent vertices. A *stable set* is a set of pairwise nonadjacent vertices. We let P_k (C_k) stand for the chordless path (cycle) on k vertices.

To simplify our notation, a P_4 with vertices a, b, c, d and edges ab, bc, cd will be denoted by $abcd$. In this context, the vertices a and d are referred to as *endpoints* while b and c are termed *midpoints* of the P_4. For a set $A \subseteq V$ inducing a P_4 in G, we define $T_G(A)$ as the set of all the vertices in $V-A$ which miss no vertex in A, $I_G(A)$ as the set of all the vertices which miss every vertex in A, and $P_G(A)$ as the set of all vertices which are adjacent to the midpoints of A and miss the endpoints of A.

Theorem 2.1. *A graph $G=(V,E)$ is P_4-sparse if and only if for every set A inducing a P_4 in G, we have $V=A\cup T_G(A)\cup P_G(A)\cup I_G(A)$.*

Proof. Follows trivially from the observation that for a vertex x outside A, $A\cup\{x\}$ induces two distinct P_4's in G if and only if $x\in V-(A\cup T_G(A)\cup P_G(A)\cup I_G(A))$. □

Theorem 2.1 implies the following characterization of P_4-sparse graphs by forbidden subgraphs (the justification is immediate and left to the reader).

Corollary 2.2. *A graph G is P_4-sparse if and only if G contains no induced subgraph isomorphic to one of the graphs F_i, $0\le i\le 6$, in Fig. 2.*

An underlying P_4-sparse graph $G=(V,E)$ together with a set A inducing a P_4 $abcd$ in G is assumed throughout. We let $T(A)$, $I(A)$, and $P(A)$ stand for $T_G(A)$, $P_G(A)$, and $I_G(A)$, respectively, since no confusion is possible. By Theorem 2.1 we can write

$$V=A\cup T(A)\cup P(A)\cup I(A).$$

For further reference, we make note of the following simple results which follow directly from the definition and whose justification is immediate.

Observation 2.3. *No vertex in $T(A)$ distinguishes between adjacent vertices in $P(A)\cup I(A)$.*

[Else, if a vertex t in $T(A)$ distinguishes between adjacent vertices u, v in $P(A)\cup I(A)$, then $\{t,u,v,a,d\}$ induces two distinct P_4's, a contradiction.]

Observation 2.4. *No vertex in $P(A)\cup I(A)$ distinguishes between nonadjacent vertices in $T(A)$.*

[If a vertex u in $P(A) \cup I(A)$ distinguishes between nonadjacent vertices t, t' in $T(A)$, then the set $\{u, t, t', a, d\}$ induces two distinct P_4's, contrary to our assumption.]

Observation 2.5. *If $\bar{G}$ is connected, then every vertex in $T(A)$ misses some vertex in $P(A) \cup I(A)$.*

[We may assume $T(A)$ nonempty, for otherwise the statement is vacuously true. Since $\bar{G}$ is connected, it must be the case that $P(A) \cup I(A)$ is nonempty. If the statement is false, then some vertex in $T(A)$ is adjacent to all the vertices in $P(A) \cup I(A)$. Let F stand for the component of the subgraph of $\bar{G}$ induced by $T(A)$ containing t. Now the definition of $T(A)$, the definition of F, and Observation 2.4 combined guarantee that every vertex in F is adjacent to all the vertices in $G-F$, contradicting the connectedness of $\bar{G}$.]

Observation 2.6. *No vertex in $P(A)$ misses more than one vertex in $T(A)$.*

[Else, if a vertex p in $P(A)$ misses distinct vertices t, t' in $T(A)$, then the set $\{a, t, t', c, p\}$ induces two distinct P_4's, a contradiction.]

Observation 2.7. *No vertex in $T(A)$ misses more than one vertex in $P(A)$.*

[Else, if $t \in T(A)$ misses distinct vertices p, p' in $P(A)$, then the set $\{a, t, c, p, p'\}$ induces two distinct P_4's, a contradiction.]

In our arguments, we shall often find it convenient to rely on the properties of a special graph that we are about to define.

A graph G is termed a *spider* if the vertex set V of G admits a partition into sets S, K, R such that:

(s1). $|S| = |K| \geq 2$, S is stable, K is a clique.

(s2). Every vertex in R is adjacent to all the vertices in K and misses all the vertices in S.

(s3). There exists a bijection $f: S \to K$ such that either

$$N_G(s) \cap K = \{f(s)\} \quad \text{for all vertices } s \text{ in } S,$$

or else,

$$N_G(s) \cap K = K - \{f(s)\} \quad \text{for all vertices } s \text{ in } S.$$

Note that the graph featured in Fig. 1 is a spider with $S = \{a, b, c\}$, $K = \{a', b', c'\}$, $R = \{d\}$, and $f = \{(a, a'), (b, b'), (c, c')\}$.

It is easy to see that the complement of a spider is also a spider.

Observation 2.8. *If a graph $G = (V, E)$ is a spider, then either*

(1) *$d_G(s) = 1$ and $d_G(k) = |V| - |S|$ for every $s \in S$ and $k \in K$, or*

(2) *$d_G(s) = |K| - 1$ and $d_G(k) = |V| - 2$ for every $s \in S$ and $k \in K$.*

[To begin, if $N_G(s) \cap K = \{f(s)\}$ for all vertices in S, then, clearly, $d_G(s) = 1$ and, consequently, $d_G(k) = 1 + |K| - 1 + |R| = |K| + |R| = |V| - |S|$. The case $N_G(v) = K - \{f(s)\}$ is similar.]

Observation 2.9. *If $G = (S \cup K \cup R, E)$ is a spider and R is nonempty, then for every choice of vertices s, k, r in S, K, and R respectively, $d_G(s) < d_G(r) < d_G(k)$.*

[Follows easily from Observation 2.8 and the definition of the spider.]

Observation 2.10. *If G is a spider, then the sets S, K, R are unique.*

[Follows by Observations 2.8 and 2.9 combined.]

Observation 2.11. *Let G be a spider. Every P_4 in G has vertices in $K \cup S$ or in R only. Furthermore, if a P_4 has vertices in $K \cup S$, then it is induced by a set of the form $\{x, y, f(x), f(y)\}$ with distinct x, y in S.*

[Let $uvwz$ be a P_4 in G with vertices from both $K \cup S$ and R. We note that this P_4 cannot contain more than one vertex in R, since there is no set of two or three vertices of a P_4 which have exactly the same neighbors in the remaining part of the P_4. Since every vertex in R is adjacent to all the vertices in K, it follows that v, w are not in R. Symmetry allows us to assume that $u \in R$. Now $v \in K$ and $w \in S$. Since S is stable, and since no vertex in S is adjacent to vertices in R it follows that $z \in K$. But since v, z are nonadjacent we contradict that K is a clique. The second part of the claim follows directly from (s3).]

Observation 2.12. *Let G be a spider, G is P_4-sparse if and only if the graph G_R induced by R is P_4-sparse.*

[Trivially, the definition of the spider together with Theorem 2.1 implies that the subgraph of G induced by $K \cup S$ is P_4-sparse. By Observation 2.11, no P_4 in G contains vertices from both $K \cup S$ and R. Hence, G is P_4-sparse if and only if G_R is P_4-sparse, as claimed.]

We are now in a position to state a characterization of P_4-sparse graphs which is the key ingredient for most of our subsequent results.

Theorem 2.13. *For a graph G, the following conditions are equivalent:*

(i) *G is a P_4-sparse graph;*

(ii) *for every induced subgraph H of G with at least two vertices, exactly one of the following statements is satisfied:*

(ii.1) *H is disconnected;*

(ii.2) *$\bar{H}$ is disconnected;*

(ii.3) *H is isomorphic to a spider.*

Proof. The proof of the implication (ii) $\rightarrow$ (i) is easy: we only need observe that for all the graphs $F_0, \ldots, F_6$ in Fig. 2, (ii) fails.

To prove the implication (i) $\rightarrow$ (ii), assume that G is a P_4-sparse graph, and let H be an arbitrary induced subgraph of G. Since the conditions (ii.1), (ii.2), and (ii.3) cannot hold simultaneously, we only need prove that if (ii.1) and (ii.2) fail, then (ii.3) holds true. Since, by assumption, (ii.1) and (ii.2) fail, a result of Seinsche [17] guarantees that H contains a P_4. We choose a set A inducing a P_4 *abcd* in H such that $|P_H(A)|$ is as large as possible. (We shall write, simply, $T(A)$, $P(A)$, and $I(A)$ instead of $T_H(A)$, $P_H(A)$, and $I_H(A)$.)

Since H is P_4-sparse, Theorem 2.1 guarantees that every vertex in H belongs to exactly one of the sets A, $T(A)$, $P(A)$, $I(A)$. We may assume that

$$T(A) \cup I(A) \neq \emptyset \tag{1}$$

for otherwise there is nothing to prove: setting $S \leftarrow \{a, d\}$, $K \leftarrow \{b, c\}$, $R \leftarrow P(A)$, the statement (ii.3) follows instantly.

By replacing H by $\bar{H}$ if necessary, (1) guarantees that

$$T(A) \neq \emptyset. \tag{2}$$

Our proof of Theorem 2.13 relies on the following intermediate results which we present next.

Observation 2.14. *If $I(A)$ is nonempty, then every vertex in $T(A)$ misses a vertex in $I(A)$.*

[We may assume that $P(A) \neq \emptyset$ for otherwise the conclusion follows trivially from Observation 2.5. Consider a vertex t in $T(A)$ adjacent to all the vertices in $I(A)$. By Observation 2.5, t misses some vertex p in $P(A)$. Note that, by Observation 2.3, p is adjacent to no vertices in $I(A)$. However, for an arbitrary vertex x in $I(A)$, the set $\{x, t, p, b, c\}$ induces two distinct P_4's, a contradiction.]

Fact 2.15. $I(A) = \emptyset$.

Proof. Suppose not; let t be a vertex in $T(A)$ such that

$$|N(t) \cap (P(A) \cup I(A))| \text{ is as large as possible.}$$

By Observation 2.14, there exists a vertex in $I(A)$ nonadjacent to t. Let C stand for the component of the subgraph of H induced by $P(A) \cup I(A)$, containing this vertex. By Observation 2.3, t misses all the vertices in C. We claim that

$$C \subseteq I(A). \tag{3}$$

[To prove (3), note that by Observation 2.7, C and $P(A)$ have at most one vertex p in common. By the connectedness of C, there exists a vertex i in $C \cap I(A)$ with $pi \in E$. But now, $\{t, b, c, p, i\}$ induces two distinct P_4's, a contradiction.]

Since H is connected, some vertex z in C must have a neighbour t' in $T(A)$. By Observation 2.4, $tt' \in E$. By our choice of t, there exists a vertex z' in $P(A) \cup I(A)$ adjacent to t but not to t'. By Observation 2.3, $zz' \notin E$.

Trivially, $z' \in I(A)$, for otherwise $\{t, t', b, z, z'\}$ induces two distinct P_4's. Notice that the set $X = \{t, t', z, z'\}$ induces a P_4 in H with edges zt', $t't$, tz'.

We claim that

$$P(A) \subset P(X). \tag{4}$$

[To see that this is the case, let p stand for an arbitrary vertex in $P(A) - P(X)$. By Observation 2.3, $p \notin T(X)$ (see vertices t, z, and p). By Theorem 2.1, p belongs to $I(X)$, contradicting Observation 2.6 (see vertices p, t, t'). To show that the containment in (4) is strict, note that $A \subset P(X)$ and $A \not\subset P(A)$.]

To complete the proof of Fact 2.15, we only need observe that (4) contradicts our choice of A. □

By Observation 2.6, we write

$$P(A) = P_0 \cup P_1$$

in such a way that $p \in P_1$ if and only if p misses some vertex in $T(A)$. Note that Fact 2.15, Observations 2.6 and 2.7 combined guarantee that

$$|T(A)| = |P_1|. \tag{5}$$

We claim that

$$T(A) \text{ is a clique and } P_1 \text{ is a stable set.} \tag{6}$$

[To prove (6), let t, t' be arbitrary vertices in $T(A)$. By Observation 2.5 and Fact 2.15 combined, we find a vertex p in P_1 such that $tp \notin E$. By Observation 2.6, $pt' \in E$. Now Observation 2.4 guarantees that $tt' \in E$. The proof that P_1 is stable is similar.]

Next, we claim that

$$\text{no edge in } H \text{ has one endpoint in } P_0 \text{ and the other in } P_1. \tag{7}$$

[Let pp' be an edge in H with $p \in P_1$ and $p' \in P_0$. By the definition of P_1, there exists a unique vertex t in $T(A)$ that misses p. Since $p' \in P_0$, we have $p't \in E$. But now t, p, p' contradict Observation 2.3.]

Proof of Theorem 2.13 (continued). Finally, to complete the proof of Theorem 2.13, we claim that with the assignment

$$K \leftarrow T(A) \cup \{b, c\}; \; S \leftarrow P_1 \cup \{a, d\}; \; R \leftarrow P_0$$

H is a spider.

Trivially, by (6) K is a clique and S is a stable set; by (5) we have $|K| = |S|$; (7), Observations 2.6 and 2.7 combined, guarantee that every vertex in S misses exactly one vertex in K, and every vertex in R is adjacent to all the vertices in K and to none in S. □

Our constructive characterization of the P_4-sparse graphs relies, in part, on two graph operations devised by Lerchs [14] for the purpose of characterizing the class of cographs. More precisely, let $G_1=(V_1,E_1)$ and $G_2=(V_2,E_2)$ be disjoint graphs. Define

- $G_1 ⓪ G_2=(V_1\cup V_2, E_1\cup E_2)$;
- $G_1 ① G_2=(V_1\cup V_2, E_1\cup E_2\cup\{xy \mid x\in V_1,\ y\in V_2\})$;

It is easy to see the operations ⓪ and ① reflect (ii.1) and (ii.2), respectively, in Theorem 2.13. For the purpose of constructing the P_4-sparse graphs, we need to introduce a third graph operation to reflect (ii.3).

Consider disjoint graphs $G_1=(V_1,\emptyset)$ and $G_2=(V_2,E_2)$ with $V_2=\{v\}\cup K\cup R$ such that

(a) $|K|=|V_1|+1\geq 2$;
(b) K is a clique;
(c) every vertex in R is adjacent to all the vertices in K and nonadjacent to v;
(d) there exists a vertex v' in K such that $N_{G_2}(v)=\{v'\}$ or $N_{G_2}(v)=K-\{v'\}$.

Choose a bijection $f\colon V_1\to K-\{v'\}$ and define

$$G_1 ② G_2=(V_1\cup V_2, E_2\cup E') \tag{8}$$

with

$$E'=\begin{cases}\{xf(x)\mid x\in V_1\}, & \text{whenever } N_{G_2}(v)=\{v'\},\\ \{xz\mid x\in V_1,\ z\in K-\{f(x)\}\}, & \text{whenever } N_{G_2}(v)=K-\{v'\}.\end{cases} \tag{9}$$

The relationship between the ② operation and (ii.3) of Theorem 2.13 will be made more precise by the following facts.

Fact 2.16. *A graph G is a spider if and only if it arises from two of its proper induced subgraphs by a ② operation.*

Proof. Write $G=(V,E)$; if G is a spider then V partitions into sets S, K, R satisfying the conditions (s1)–(s3). Let v be an arbitrary vertex in S. Now it is a routine task to check that G arises from the graphs $G_1=(S-\{v\},\emptyset)$ and $G_2=(\{v\}\cup K\cup R, E_2)$ with $E_2=E-\{xy\mid x\in S-\{v\},\ y\in K\}$, by a ② operation.

Conversely, assume that G arises from two of its proper induced subgraphs G_1 and G_2 by a ② operation. We only need verify that the conditions (s1)–(s3) hold true. For this purpose, write $G_1=(V_1,\emptyset)$, $G_2=(\{v\}\cup K\cup R, E_2)$, $S\leftarrow V_1\cup\{v\}$ such that (a)–(d) and (8), (9) are satisfied.

To see that (s1) is true, note that S is stable; by (b), K is a clique; by (a), $|S|=|K|\geq 2$. Next, (s2) follows trivially from (c) and (9) combined. Finally, (s3) follows from (8) and (9) combined. □

Fact 2.17. *If G is a spider, then $G=G_1 ② G_2$ with G_1, G_2 unique up to isomorphism.*

Proof. By Observation 2.10, the vertex set V of G admits a unique decomposition into disjoint sets S, K, R satisfying (s1)–(s3). If the statement is false, then we have $G=G_1 ② G_2$ and $G=G_1' ② G_2'$ such that either $G_1 \neq G_1'$ or $G_2 \neq G_2'$.

Write $G_2=(V_2,E_2)$; $G_2'=(V_2',E_2')$. Clearly, $K \cup R \subseteq V_2$ and $K \cup R \subseteq V_2'$. In fact, we can write $V_2=\{v_2\} \cup K \cup R$ and $V_2'=\{v_2'\} \cup K \cup R$. It is easy to see that the only way $G_2 \neq G_2'$ is that $|N_{G_2}(v_2) \cap K| \neq |N_{G_2'}(v_2') \cap K|$. Since v_2, $v_2' \in S$, this contradicts (s3). Similarly, it is easy to see that $G_1 \neq G_1'$ leads to a contradiction. □

As it turns out, all P_4-sparse graphs are constructible by means of the operations ⓪, ①, and ②. More precisely, we state the following result.

Theorem 2.18. *For a graph G the following statements are equivalent*:

(i) *G is a P_4-sparse graph*;

(ii) *G is obtained from single-vertex graphs by a finite sequence of operations* ⓪, ①, ②.

Proof. Let $G=(V,E)$ be obtained from single-vertex graphs by a finite sequence σ of zero or more operations ⓪, ①, ②. We prove the implication (ii) → (i) by induction on the length of σ. Write $\sigma=s_0s_1\ldots s_n$ ($n \geq 0$); assume the statement true for graphs obtained by sequences involving fewer operations than σ. If s_n involves the nonempty graphs $G_1=(V_1,E_1)$ and $G_2=(V_2,E_2)$ then, by the induction hypothesis both G_1 and G_2 are P_4-sparse graphs.

Furthermore, if s_n is one of the operations ⓪ or ①, then we are done by the induction hypothesis: no P_4 in G has vertices from both V_1 and V_2, implying that G is P_4-sparse.

If s_n is a ② operation then, by Fact 2.16, G is a spider; now the induction hypothesis, together with Observations 2.11 and 2.12 combined guarantee that G is P_4-sparse.

To prove the implication (i) → (ii), we proceed by induction on the size of G. If G contains a single vertex, then there is nothing to prove. Assuming the implication true for all the P_4-sparse graphs with fewer vertices than G, we propose to show that G itself satisfies the implication.

For this purpose, note that if G is disconnected, then G arises from two of its proper induced subgraphs by a ⓪ operation; if $\bar{G}$ is disconnected, then G arises from two of its proper induced subgraphs by a ① operation. Finally, by Theorem 2.13, if both G and $\bar{G}$ are connected, then G is a spider. Now Fact 2.16 guarantees that G arises by a ② operation from two of its proper induced subgraphs, and the proof of Theorem 2.18 is complete. □

Theorems 2.13 and 2.18 suggest a natural way of associating with every P_4-sparse graph G a tree $T(G)$ (called the *ps-tree* of G). We describe the formal construction of the ps-tree of a P_4-sparse graph G by the following recursive procedure.

Procedure Build_tree(G);
{*Input*: a P_4-sparse graph $G=(V,E)$;
Output: the ps-tree $T(G)$ corresponding to G}

begin
  if $|V|=1$ **then**
    return the tree $T(G)$ consisting of the unique vertex of G;
  if G ($\bar{G}$) is disconnected **then**
  begin
    let $G_1, G_2, \ldots, G_p$ ($p \geq 2$) be the components of G ($\bar{G}$);
    let $T_1, T_2, \ldots, T_p$ be the corresponding ps-trees rooted at $r_1, r_2, \ldots, r_p$;
    return the tree $T(G)$ obtained by adding $r_1, r_2, \ldots, r_p$ as children of a node labelled 0 (1);
  end
  else
  begin
    {now both G and $\bar{G}$ are connected}
    write $G = G_1 ② G_2$ as in (8) and (9);
    let T_1, T_2 be the corresponding ps-trees rooted at r_1 and r_2;
    return the tree $T(G)$ obtained by adding r_1, r_2 as children of a node labelled 2
  end
end; {Build_tree}

Clearly, Procedure Build_tree runs in polynomial time. To see this, note that the connected components of G (or $\bar{G}$) can be found efficiently by performing a depth-first search on G or $\bar{G}$, respectively. In case both G and $\bar{G}$ are connected, then the unique set S featured in Theorem 2.13 can be found in linear time by selecting all vertices of lowest degree, as guaranteed by Observation 2.8 and 2.9.

Furthermore, by Theorem 2.13, Fact 2.17, and Theorem 2.18 combined, the ps-tree of a P_4-sparse graph G is unique up to isomorphism. In addition, it is easy to see that

- the leaves of $T(G)$ are precisely the vertices of G;
- an internal node w of $T(G)$ is labelled by 0, 1, or 2 according to the following rule:

$$\text{label}(w) = \begin{cases} 0, & \text{iff } G_{L(w)} \text{ is disconnected,} \\ 1, & \text{iff } \bar{G}_{L(w)} \text{ is disconnected,} \\ 2, & \text{otherwise.} \end{cases}$$

[Here, $L(w)$ is the set of all the leaf descendants of w.]

An interesting computational problem, given a graph G, asks for the largest induced subgraph of G which contains no P_4. As mentioned in the introduction, the general problem is known to be intractable (see Corneil et al. [8]) but, as it turns out, it can be solved efficiently for P_4-sparse graphs.

Let $G=(V,E)$ be a P_4-sparse graph. The *canonical cograph* $C(G)$ associated with G is the induced subgraph of G obtained by the following procedure.

```
Procedure Greedy(G);
{Input: a P4-sparse graph G;
 Output: the canonical cograph C(G)}
begin
  C(G) ← G;
  while there exist P4's in C(G) do
  begin
    pick a P4 uvxy in C(G);
    pick z at random in {u, y};
    C(G) ← C(G) − {z}
  end;
  return(C(G))
end;
```

An easy inductive argument shows that for every induced subgraph H of G which satisfies (ii.3) in Theorem 2.13, Procedure Greedy removes all the vertices in S, except for an arbitrary one. Clearly, the graph $C(G)$ returned by Greedy is a cograph; the fact that $C(G)$ is as large as possible follows from Observation 2.11, Theorem 2.13 and an easy inductive argument. The uniqueness implied by the definition is justified by the following stronger result.

Theorem 2.19. *For a graph G with no induced C_5, the following statements are equivalent*:

(i) *G is P_4-sparse*;

(ii) *for every induced subgraph H of G, $C(H)$ is unique up to isomorphism.*

Proof. To prove the implication (ii)$\rightarrow$(i), we shall use the characterization of P_4-sparse graphs by forbidden subgraphs given in Corollary 2.2. We need only observe that for each of the graphs F_i, $1\le i\le 6$, $C(F_i)$ is not unique.

To prove the implication (i)$\rightarrow$(ii), we proceed by induction on the size of G. If G contains a single vertex, then we are done. Assume the statement true for all P_4-sparse graphs with fewer vertices than G.

If G or $\bar{G}$ is disconnected, then we are done by the induction hypothesis since no P_4 in G has vertices in distinct components of G or $\bar{G}$. We may assume, therefore, that both G and $\bar{G}$ are connected. Now Theorem 2.13 guarantees that G is a spider.

By virtue of Observation 2.10, the vertex set of G partitions uniquely into sets S, K, and R satisfying (s1)–(s3). By the induction hypothesis, the subgraph G' of G induced by $K \cup R$ has a canonical cograph $C(G')$ unique up to isomorphism. To complete the proof of Theorem 2.19, we note that a canonical cograph $C(G)$ is obtained from $C(G')$ by adding a single vertex from S. The conclusion follows by (s3) and Observation 2.8. □

Our next result, Theorem 2.21, shows that the canonical cograph associated with a P_4-sparse graph can be used to solve the four optimization problems (mentioned in the introduction) for P_4-sparse graphs by reducing them to the corresponding optimization problems on cographs [7,8]. Our proof of Theorem 2.21 relies on the following result.

Fact 2.20 (Meyniel [15]). *Let G be an arbitrary graph and let u, v be nonadjacent vertices in G such that u and v are not the endpoints of the same P_4 in G. The graph G', obtained from G by deleting v and by joining u by an edge to all the vertices in $N_G(v)$, satisfies $\omega(G')=\omega(G)$.*

Since the P_4-sparse graphs are closed under complementation, we let $C(\bar{G})$ stand for the canonical cograph of the complement $\bar{G}$ of G.

Theorem 2.21. *Let G be a P_4-sparse graph and let $C(G)$ be the canonical subgraph of G. The following statements are satisfied*

(4.1) $\omega(G)=\omega(C(G))$,

(4.2) $\chi(G)=\chi(C(G))$,

(4.3) $\alpha(G)=\omega(C(\bar{G}))$,

(4.4) $\theta(G)=\chi(C(\bar{G}))$.

Proof. If G is a cograph, then there is nothing to prove: G and $C(G)$ coincide.

We shall, therefore, assume that G contains a P_4. Let $A=\{a,b,c,d\}$ induce a P_4 in G with edges ab, bc, cd. We claim that

$$N_G(a) \subset N_G(c). \tag{10}$$

[Let a' be an arbitrary vertex in $N_G(a)$. If a' misses c, then we contradict Theorem 2.1. The inclusion is strict since c is adjacent to d, while a is not.]

Consider the graph G' obtained from G be removing a and making all the neighbours of a adjacent to c. By (10),

G' is an induced subgraph of G.

By Fact 2.20, and the fact that a and c are not the endpoints of the same P_4 in G, we have

$$\omega(G)=\omega(G').$$

Now Theorem 2.19 together with an easy inductive argument, shows that (4.1) must be true.

To settle (4.2), note that the cographs are perfect (see [8], for example). It follows that

$$\chi(C(G)) = \omega(C(G)).$$

Observe that every colouring of $C(G)$ using $\chi(C(G))$ colours extends trivially into a colouring of G with the same number of colours. But now, obviously,

$$\chi(G) \leq \chi(C(G)) = \omega(C(G)) = \omega(G) \leq \chi(G)$$

and so equality must hold throughout.

To settle (4.3), we note that, trivially, $\alpha(G) = \omega(\bar{G})$. Now (4.1) guarantees that $\alpha(G) = \omega(C(\bar{G}))$.

Finally, to settle (4.4), we note that $\theta(G) = \chi(\bar{G})$ and the result follows by (4.2). □

3. Discussion

In this work we have investigated the class of P_4-sparse graphs for which a tree representation unique up to isomorphism has been developed. The conversion between a P_4-sparse graph and the corresponding tree representation can be carried out in polynomial time and, consequently, the graph isomorphism problem can be solved in polynomial time for P_4-sparse graphs. It would be of interest to further investigate this tree structure for the purpose of solving efficiently other computational problems important in applications such as: clustering, minimum fill-in, minimum weight dominating set, hamiltonicity and others.

It is interesting to note that Theorem 2.13 leads, quite naturally, to a different decomposition of P_4-sparse graphs as follows:

- if the graph G is disconnected, then decompose each component separately;
- if the complement is disconnected, then decompose each connected component of the complement separately;
- otherwise, by Theorem 2.13, G is a spider with the vertex set partitioned into S, K, R; if R is not empty, decompose G into $G_{S \cup K}$ and G_R.

At the end of such a decomposition, we obtain isolated vertices and spiders with an empty set R. The obvious disadvantage of this decomposition is that the leaves of the obtained tree are no longer single vertices. However, in such a decomposition the P_4's are restricted to leaves. This makes it possible to see at a glance that P_4-sparse graphs are superbrittle [16] and that a perfect order for P_4-sparse graphs is easy to obtain.

Finally, note that our characterization of P_4-sparse graphs by a finite number of forbidden configurations immediately suggests a naive (but polynomial) recognition algorithm: form all $O(n^5)$ distinct subsets of five vertices of G and for each of them check in constant time whether they are isomorphic to one of the forbidden graphs. In [8,10] linear-time recognition algorithms are given for cographs and P_4-reducible graphs. We conjecture that the P_4-sparse graphs can be recognized in linear time by using similar techniques.

Acknowledgement

The authors are indebted to an anonymous referee for many constructive comments and suggestions that contributed to a much improved presentation of this work. In particular, the referee pointed out the decomposition mentioned in Section 3.

References

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

[2] C. Berge, Graphs and Hypergraphs (North-Holland, Amsterdam, 1973).

[3] T. Beyer, W. Jones and S. Mitchell, A linear algorithm for isomorphism of maximal outerplanar graphs, CS-TR-78-1, Department of Computer Science, University of Oregon, Eugene, OR (1978).

[4] K.S. Booth and G.S. Lueker, A linear time algorithm for deciding interval graph isomorphism, J. ACM 26 (1979) 183–195.

[5] V. Chvátal, Perfectly ordered graphs, in: C. Berge and V. Chvátal, eds., Topics on Perfect Graphs (North-Holland, Amsterdam, 1984) 63–65.

[6] D.G. Corneil and D.G. Kirkpatrick, Families of recursively defined perfect graphs, Congr. Numer. 39 (1983) 237–246.

[7] D.G. Corneil, H. Lerchs and L. Stewart Burlingham, Complement reducible graphs, Discrete Appl. Math. 3 (1981) 163–174.

[8] D.G. Corneil, Y. Perl and L.K. Stewart, A linear recognition algorithm for cographs, SIAM J. Comput. 14 (1985) 926–934.

[9] C. Hoàng, Doctoral Dissertation, McGill University, Montreal, Que. (1985).

[10] B. Jamison and S. Olariu, A linear-time recognition algorithm for P_4-reducible graphs, in: Proceedings 9th Conference on Foundations of Software Technology and Theoretical Computer Science, Lecture Notes in Computer Science 405 (Springer, Berlin, 1989) 1–19.

[11] B. Jamison and S. Olariu, P_4-reducible graphs, a class of uniquely tree representable graphs, Stud. Appl. Math. 81 (1989) 79–87.

[12] H.A. Jung, On a class of posets and the corresponding comparability graphs, J. Combin. Theory Ser. B 24 (1978) 125–133.

[13] E.L. Lawler, Graphical algorithms and their complexity, Math. Centre Tracts 81 (1976) 3–32.

[14] H. Lerchs, On the clique-kernel structure of graphs, Department of Computer Science, University of Toronto, Toronto, Ont. (1972).

[15] H. Meyniel, A new property of critical imperfect graphs and some consequences, European J. Combin. 8 (1987) 313–316.

[16] M. Preissmann, D. de Werra and N.V.R. Mahadev, A note on superbrittle graphs, Discrete Math. 61 (1986) 259–267.

[17] D. Seinsche, On a property of the class of n-colorable graphs, J. Combin. Theory Ser. B 16 (1974) 191–193.

[18] L. Stewart, Cographs, a class of tree representable graphs, M. Sc. Thesis, TR 126/78, Department of Computer Science, University of Toronto, Toronto, Ont. (1978).

[19] D.P. Sumner, Dacey graphs, J. Austral. Math. Soc. 18 (1974) 492–502.