Discrete Applied Mathematics 113 (2001) 255–260

DISCRETE APPLIED MATHEMATICS

Note

Recognition of quasi-Meyniel graphs

Celina M.H. de Figueiredo[a,*], Kristina Vušković[b]

[a] *Instituto de Matemática and COPPE, Universidade Federal do Rio de Janeiro, Caixa Postal 68530, 21945-970 Rio de Janeiro, RJ, Brazil*

[b] *Department of Mathematics, University of Kentucky, 715 Patterson Office Tower, Lexington, KY 40506, USA*

Received 18 November 1998; revised 3 December 1999; accepted 10 July 2000

Abstract

We present a polynomial-time algorithm for recognizing quasi-Meyniel graphs. A *hole* is a chordless cycle with at least four vertices. A *cap* is a cycle with at least five vertices, with a single chord that forms a triangle with two edges of the cycle. A graph G is *quasi-Meyniel* if it contains no odd hole and for some $x \in V(G)$, the chord of every cap in G has x as an endvertex. Our recognition algorithm is based on star cutset decompositions.

Keywords: Analysis of algorithms and problem complexity; Graph decomposition algorithms; Perfect graphs; Star cutsets

1. Quasi-Meyniel graphs

A cycle is *even* if it contains an even number of vertices, and it is *odd* otherwise. A *hole* is a chordless cycle with at least four vertices. A *cap* is a cycle with at least five vertices, with a single chord that forms a triangle with two edges of the cycle. A graph G *contains* a graph H if H is an induced subgraph of G. A graph is H*-free* if it does not contain H.

A graph is *Meyniel* if every odd cycle with at least five vertices has two or more chords [11]. Clearly, a graph is Meyniel if and only if it contains no odd hole and no cap. A graph G is *quasi-Meyniel* if it contains no odd hole, and if it contains a *tip*: a vertex x such that the chord of every cap in G has x as an endvertex. This class of graphs is introduced by Hertz [9], where he gives an efficient coloring algorithm for Meyniel graphs, by coloring quasi-Meyniel graphs with given tips.

* Corresponding author.

E-mail addresses: celina@cos.ufrj.br (C.M.H. de Figueiredo), kristina@ms.uky.edu (K. Vušković).

0166-218X/01/$ - see front matter
PII: S0166-218X(00)00290-0

Every Meyniel graph G is quasi-Meyniel, with every vertex of G being a tip. If G is quasi-Meyniel but not Meyniel, then G must contain a cap and G has at most two tips. If G is quasi-Meyniel with tip x, then every induced subgraph of G containing x is quasi-Meyniel with tip x and every induced subgraph of G not containing vertex x is Meyniel.

The goal of this paper is to address two algorithmic questions: how to find a tip of a quasi-Meyniel graph, and how to recognize quasi-Meyniel graphs.

Finding a tip of a quasi-Meyniel graph is important for coloring purposes. The search for polynomial-time algorithms for classes of graphs defined by chordality conditions has attracted much interest, especially in the context of Perfect Graph Theory [4–8,10]. A graph G is perfect if, for all induced subgraphs H of G, the size of the largest clique in H is equal to the chromatic number of H. A long-standing conjecture of Berge [1] states that G is perfect if and only if neither G nor its complement contain an odd hole. The existence of a polynomial-time algorithm to test whether G contains an odd hole implies a polynomial-time algorithm to test whether G is perfect, modulo the verification of Berge's conjecture, and it is possible that such an algorithm may itself prove the conjecture.

Finding a cap in a graph G or verifying that G does not contain one can be done as follows. For every edge xy in G and for every $z \in N(x) \cap N(y)$, check whether x and y both have a neighbor in the same component C of $G \backslash ((N(x) \cap N(y)) \cup N(z))$. If they do, then the node set $\{x, y, z\} \cup V(P)$, where P is a shortest path in C whose one endnode is adjacent to x and the other to y, induces a cap with chord xy. If the condition fails for all choices of nodes x, y and z, then G does not contain a cap.

Note that by finding chords of all of the caps of a graph G, one can easily find a tip candidate of G: a node that is an endnode of the chord of every cap. But this is not sufficient to recognize quasi-Meyniel graphs since one still has to check whether there is an odd hole that passes through the tip, and this problem is NP-complete in general [2].

The recognition of Meyniel graphs in polynomial time was established by Burlet and Fonlupt [3] when they defined the amalgam decomposition and proved that the amalgam of two Meyniel graphs is a Meyniel graph and, conversely, that any Meyniel graph can be amalgam decomposed in polynomial time into basic Meyniel graphs, which in turn can be recognized in polynomial time. The polynomial-time algorithm for recognizing quasi-Meyniel graphs that we propose here is based on a decomposition through star cutsets which preserves in both senses, ascending and descending, the property of being quasi-Meyniel. This decomposition for quasi-Meyniel graphs yields a decomposition tree whose leaves are Meyniel graphs.

2. Decomposition

Given a graph G and $S \subseteq V(G)$, we denote by $G \backslash S$ the subgraph of G induced by the vertex set $V(G) \backslash S$. A node set $S \subseteq V(G)$ is a *cutset* of a connected graph G if the graph $G \backslash S$ is disconnected. A node set $S \subseteq V(G)$ is a *star cutset with center x* of G if

S is a cutset of G and some $x \in S$ is adjacent to all the vertices of $S\backslash\{x\}$. Let a node set S be a cutset of a graph G, and let $C_1, \ldots, C_n$ be the connected components of $G\backslash S$. The *blocks of decomposition* by S are graphs $G_1, \ldots, G_n$ where G_i is the subgraph of G induced by the vertex set $V(C_i) \cup S$.

Lemma 1. *Suppose that a graph G and a vertex $x \in V(G)$ are such that $G\backslash\{x\}$ is Meyniel. Let S be a star cutset of G with center x, and let $G_1, \ldots, G_n$ be the blocks of decomposition by S. Then G is quasi-Meyniel with tip x if and only if G_i is quasi-Meyniel with tip x, for every i.*

Proof. If G is quasi-Meyniel with tip x, so are all the G_i's, since they are induced subgraphs of G containing vertex x. To prove the converse, assume that G contains an odd hole H or a cap H in which x is not an endvertex of a chord. Since $G\backslash\{x\}$ is Meyniel, H contains x. But then $H\backslash S$ is contained in some connected component C_i of $G\backslash S$, and so H is contained in G_i. Hence, G_i is not quasi-Meyniel with tip x. □

A *wheel*, denoted by (H,x), is a graph induced by a hole H and a vertex $x \notin V(H)$ having at least three neighbors in H. Vertex x is the *center* of the wheel. A *sector* of the wheel is a subpath of H whose endvertices are neighbors of x and intermediate nodes are not. A *short sector* is a sector of length 1, and a *long sector* is a sector of length at least 2. A *twin wheel* is a wheel with three sectors, two of which are short.

Given a triangle $\{x_1,x_2,x_3\}$ and a vertex y adjacent to at most one vertex in $\{x_1,x_2,x_3\}$, a $3PC(x_1x_2x_3, y)$ is a graph induced by three chordless paths, $P_1 = x_1 \ldots y$, $P_2 = x_2 \ldots y$ and $P_3 = x_3 \ldots y$, having no common vertices other than y, and such that the only adjacencies between the vertices of $P_1\backslash\{y\}$, $P_2\backslash\{y\}$, and $P_3\backslash\{y\}$ are the edges of the triangle $\{x_1,x_2,x_3\}$. Note that $V(P_i) \cup V(P_j)$ induces a hole, when $i \neq j$.

The following simple facts will be used in the proof of Lemma 2 below:

Fact 1. Let (H, y) be a wheel in a quasi-Meyniel graph G with tip x, such that $y \neq x$. If (H, y) has both a short and a long sector, then (H, y) is a twin wheel.

Fact 2. Odd-hole-free graphs cannot contain a $3PC(x_1x_2x_3, y)$.

Lemma 2. *Let G be a quasi-Meyniel graph with tip x. Suppose G contains a cap induced by a hole $H = xx_1 \ldots x_kx$ and a vertex y adjacent to x and x_k. Then $S = (N(x) \cup \{x\})\backslash\{y\}$ is a cutset separating y from $H\backslash S$.*

Proof. We prove that $T = (N(x) \cup \{x\})\backslash\{y,x_1\}$ is a cutset separating y from $H\backslash T$, which clearly implies the lemma. Suppose not and let $P = p_1 \ldots p_n$ be a path in $G\backslash T$ such that p_1 is adjacent to y, p_n is adjacent to a node of $H\backslash T$, and no proper subset of $V(P)$ induces a path with these properties. Note that x does not have a neighbor in P, but x_k possibly does. Let x_i be the vertex of H with lowest index adjacent to p_n. Note that $i < k$. Let H' be the hole induced by the vertex set $V(P) \cup \{y,x,x_1,\ldots,x_i\}$. Let x_j be the neighbor of p_n in $H\backslash\{x_k\}$ with highest index. Note that possibly $i = j$. We now consider the following two cases.

Case 1: *x_k is adjacent to a vertex of P*. Then (H',x_k) is a wheel that contains at least one short sector (since x_k is adjacent to x and y) and at least one long sector (since x_k is not adjacent to x_1). Hence, by Fact 1, (H',x_k) is a twin wheel. So the only neighbor of x_k in P is p_1. If $n \neq 1$ or $j \neq k-1$, then the vertex set $V(P) \cup \{y,x_j,\ldots,x_k\}$ induces a cap, contradicting the assumption that x is a tip of G. Hence, $n=1$ and $j=k-1$. If $i=j$, then (H',x_k) is a wheel that contradicts Fact 1. Hence, $i \neq j$, and so (H,p_1) is a wheel. By Fact 1, (H,p_1) is a twin wheel. So the neighbors of p_1 in H are x_k,x_{k-1} and x_{k-2}. But then the vertex set $(V(H) \cup \{y,p_1\}) \backslash \{x_k\}$ induces a cap, contradicting the assumption that x is a tip of G.

Case 2: *x_k is not adjacent to a vertex of P*. If $i=j$, then the vertex set $V(H) \cup V(P) \cup \{y\}$ induces a $3PC(xyx_k,x_i)$, contradicting Fact 2. Hence $i \neq j$. If x_ix_j is an edge, then the vertex set $V(P) \cup \{y,x_i,\ldots,x_k\}$ induces a cap, contradicting the assumption that x is a tip of G. Otherwise, the vertex set $V(P) \cup \{x,x_1,\ldots,x_i,x_j,\ldots,x_k\} \cup \{y\}$ induces a $3PC(xyx_k,p_n)$, contradicting Fact 2. □

A *good star cutset with center x* is a cutset of the form $S=(N(x) \cup \{x\}) \backslash \{y\}$, where $y \in N(x)$ (the kind of a cutset used in Lemma 2).

3. Recognition algorithm

To test whether a graph G is quasi-Meyniel, we first test whether G contains a cap. If it does not, then it is sufficient to test whether G is Meyniel. If a cap with chord xy is detected, then we apply the recognition algorithm below twice, to check whether G is quasi-Meyniel with tip x and to check whether G is quasi-Meyniel with tip y.

Algorithm 1

Input: A graph G and a vertex $x \in V(G)$
Output: YES, if G is quasi-Meyniel with tip x, and NO otherwise

```
if G\{x} is not Meyniel then return NO
L1 ← G, L2 ← ∅;
while L1 ≠ ∅ do
  remove a graph F from L1
  if there is a good star cutset S with center x in F then
    decompose F by S and add the blocks of decomposition to L1
  else
    add F to L2
if all the graphs in L2 are Meyniel then
  return YES
else
  return NO
```

Lemma 3. *Algorithm 1 correctly identifies whether G is quasi-Meyniel with tip x.*

Proof. If $G\backslash\{x\}$ is not Meyniel, then clearly G is not quasi-Meyniel with tip x, and Algorithm 1 correctly terminates. So, assume that $G\backslash\{x\}$ is Meyniel. Correctness of Algorithm 1 will follow from showing that G is quasi-Meyniel with tip x if and only if all the graphs in $\mathscr{L}_2$ are Meyniel. Algorithm 1 builds a decomposition tree whose internal nodes correspond to graphs having a good star cutset with center x, whereas the leaves correspond to graphs having no such cutset. Applying Lemma 1 to every graph corresponding to an internal node, we have that G is quasi-Meyniel with tip x if and only if all the graphs in $\mathscr{L}_2$ are quasi-Meyniel with tip x. Let $F \in \mathscr{L}_2$. We now show that F is quasi-Meyniel with tip x if and only if F is Meyniel. If F is Meyniel, then trivially F is quasi-Meyniel with tip x. To show the converse, assume that F is quasi-Meyniel with tip x, but that F is not Meyniel. Then F contains a cap with x being an endvertex of its chord. But then, by Lemma 2, F has a good star cutset with center x, contradicting the assumption that $F \in \mathscr{L}_2$. □

3.1. Complexity analysis

Let G be a graph with n vertices and m edges. The above-described test for the existence of a cap takes time $\mathcal{O}(n^3m)$. The algorithm in [3] for checking whether a graph is Meyniel is $\mathcal{O}(n^7)$. More recently, Roussel and Rusu [12] showed how to do this recognition in time $\mathcal{O}(m^2+mn)$. The complexity of the proposed quasi-Meyniel graph recognition is $\mathcal{O}(n^3m)$, being dominated by the complexity of Algorithm 1 which checks whether G is quasi-Meyniel with tip x, whose complexity we now establish to be $\mathcal{O}(mn^3+m^2n+mn^2)$.

Consider an internal node of the decomposition tree and suppose it corresponds to decomposing the graph H with a good star cutset $S=(N(x)\cup\{x\})\backslash\{y\}$. Let z be a node of a component of $H\backslash S$ that does not contain y. Label the corresponding internal node of the decomposition tree with pair (y,z). Clearly no two internal nodes are labeled with the same pair, so the number of internal nodes in the decomposition tree is $\mathcal{O}(n^2)$. Decomposition through a good star cutset with center x can be performed in time $\mathcal{O}(mn)$: for every neighbor y of x, test whether $(N(x)\cup\{x\})\backslash\{y\}$ is a cutset in time $\mathcal{O}(m)$. Thus the total cost of building the decomposition tree is $\mathcal{O}(mn^3)$.

The leaves in the decomposition tree that do not contain any non-neighbors of x are clearly Meyniel. The number of leaves that contain a non-neighbor of x is $\mathcal{O}(n)$, since no two distinct leaves can contain the same non-neighbor of x. So verifying whether the graphs in $\mathscr{L}_2$ are Meyniel can be implemented to run in time $\mathcal{O}(m^2n+mn^2)$, assuming an $\mathcal{O}(m^2+mn)$ algorithm for testing whether a graph is Meyniel. Hence, Algorithm 1 can be implemented to run in time $\mathcal{O}(mn^3+m^2n+mn^2)$.

Acknowledgements

This research was done while the second author was visiting the Universidade Federal do Rio de Janeiro supported by Conselho Nacional de Desenvolvimento Científico e

Tecnológico, CNPq. The first author is partially supported by CNPq and PRONEX/ FINEP. We thank referees for their careful reading and valuable suggestions, which helped improve the content and presentation of this paper.

References

[1] C. Berge, Les problèmes de coloration en théorie des graphes, Publ. Inst. Statist. Univ. Paris 9 (1960) 123–160.
[2] D. Bienstock, On the complexity of testing for odd holes and induced odd paths, Discrete Math. 90 (1991) 85–92. Corrigendum in Discrete Math. 102 (1992) 109.
[3] M. Burlet, J. Fonlupt, Polynomial algorithm to recognize a Meyniel graph, Ann. Discrete Math. 21 (1984) 225–252.
[4] M. Conforti, G. Cornuéjols, A. Kapoor, K. Vušković, Universally signable graphs, Combinatorica 17 (1997) 67–77.
[5] M. Conforti, G. Cornuéjols, A. Kapoor, K. Vušković, Even and odd holes in cap-free graphs, J. Graph Theory 30 (1999) 289–308.
[6] M. Conforti, G. Cornuéjols, A. Kapoor and K. Vušković, Finding an even hole in a graph, in: 38th Annual Symp. on Foundations of Computer Science, Miami Beach, FL, October 1997. IEEE, pp. 480–485.
[7] M. Conforti, G. Cornuéjols, A. Kapoor, K. Vušković, Even-hole free graphs, Part I: decomposition, Part II: recognition algorithm, preprint, 1997, J. Graph Theory, submitted for publication.
[8] M. Conforti, G. Cornuéjols, A. Kapoor, K. Vušković, Triangle-free graphs that are signable without even holes, preprint, 1996, submitted for publication.
[9] A. Hertz, A fast algorithm for colouring Meyniel graphs, J. Combin. Theory Ser. B 50 (1990) 231–240.
[10] A. Lubiw, Short-chorded and perfect graphs, J. Combin. Theory Ser. B 51 (1991) 24–33.
[11] H. Meyniel, The graphs whose odd cycles have at least two chords, Ann. Discrete Math. 21 (1984) 115–120.
[12] F. Roussel, I. Rusu, Holes and dominos in Meyniel graphs, Internat. J. Found. Comput. Sci. 10 (1999) 127–146.