



**HAL**  
open science

## Recognizing Chordal-Bipartite Probe Graphs

Anne Berry, Elad Cohen, Martin C. Golumbic, Marina Lipshteyn, Nicolas Pinet,  
Alain Sigayret, Michal Stern

► **To cite this version:**

Anne Berry, Elad Cohen, Martin C. Golumbic, Marina Lipshteyn, Nicolas Pinet, et al.. Recognizing Chordal-Bipartite Probe Graphs. 2007. <hal-00678308>

**HAL Id: hal-00678308**

**<https://hal.science/hal-00678308v1>**

Submitted on 12 Mar 2012

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

# Recognizing Chordal-Bipartite Probe Graphs

A. Berry<sup>1</sup> E. Cohen<sup>2</sup> M.C. Golumbic<sup>2</sup>  
M. Lipshteyn<sup>2</sup> N. Pinet<sup>1</sup> A. Sigayret<sup>1</sup> M. Stern<sup>2,3</sup>

Research Report LIMOS/RR-07-09

16 avril 2007

<sup>1</sup>LIMOS, Ensemble scientifique des Cézéaux, 63177 Aubière cedex, France. Fax 00 33 473 40 76 39. berry@isima.fr, pinet@isima.fr, sigayret@isima.fr

<sup>2</sup>Caesarea Rothschild Institute and Department of Computer Science, University of Haifa, Mt. Carmel, Haifa, Israel. golumbic@cs.haifa.ac.il

<sup>3</sup>The Academic College of Tel-Aviv - Yaffo, Israel.

## Abstract

A graph  $G$  is chordal-bipartite probe if its vertices can be partitioned into two sets  $P$  (probes) and  $N$  (non-probes) where  $N$  is a stable set and such that  $G$  can be extended to a chordal-bipartite graph by adding edges between non-probes. A bipartite graph is called chordal-bipartite if it contains no chordless cycle of length strictly greater than 5. Such probe/non-probe completion problems have been studied previously on other families of graphs, such as interval graphs and chordal graphs.

In this paper, we give a characterization of chordal-bipartite probe graphs, in the case of a fixed given partition of the vertices into probes and non-probes. Our results are obtained by solving first the more general case without assuming that  $N$  is a stable set, and then this can be applied to the more specific case. Our characterization uses an edge elimination ordering which also implies a polynomial time recognition algorithm for the class.

This research was conducted in the context of a France-Israel Binational project, while the French team visited Haifa in March 2007.

**Keywords:** Chordal-bipartite graphs, probe graphs, elimination schemes.

## Résumé

Un graphe  $G$  est un graphe de sonde chordal-biparti quand ses sommets peuvent être partitionnés en deux ensembles  $P$  (probes) et  $N$  (non-probes) tels que  $N$  soit un stable et que  $G$  puisse être étendu à un graphe chordal-biparti par ajout d'arêtes entre non-probes. Un graphe biparti est dit chordal-biparti quand il ne contient aucun cycle sans corde de longueur strictement supérieure à 5. Ce type de problèmes de complétion probe/non-probe a été étudié précédemment sur d'autres familles de graphes, comme les graphes d'intervalles ou les graphes triangulés.

Nous donnons ici une caractérisation des graphes de sonde chordaux-bipartis dans le cas où la partition en probe/non-probe est donnée. Nos résultats sont obtenus en résolvant d'abord le cas plus général où  $N$  n'a pas besoin d'être un stable, ce qui peut être ensuite appliqué au cas spécifique. Notre caractérisation utilise un ordre d'élimination par arêtes, ce qui implique aussi un temps de reconnaissance polynomial pour la classe de graphes considérée.

**Mots clés :** graphe chordal-biparti, graphe de sonde, schéma d'élimination.

# 1 Introduction

Let  $\mathcal{C}$  be a graph class. We will say that a graph  $G = (V, E)$  is  $\mathcal{C}$  *probe* if  $V$  is partitioned into two sets  $P$  (probes) and  $N$  (non-probes) where  $N$  is a stable (i.e. independent) set and such that  $G$  can be extended to a graph  $G^* = (V, E + F)$  in  $\mathcal{C}$  by adding edges between some pairs of non-probes. The  $\mathcal{C}$ -Probe Graph Recognition problem is a special case of the  $\mathcal{C}$ -Graph Sandwich problem [7].

Such probe/non-probe completion problems have been studied previously on other families of graphs. Specifically, interval probe graphs were introduced by Zhang in [17] and studied further in [14, 18, 19]. Polynomial time recognition algorithms (with respect to a fixed partition) were given in [12] using PQtrees, and in [13] using modular decomposition. In the case of trees which are interval probe graphs, Sheng [15] gave a characterization by a family of forbidden subgraphs (see also [10]).

Generalizing interval probe graphs, Golumbic and Lipshteyn [8, 9] introduced chordal probe graphs as a new class of perfect graphs, namely, partitioned graphs which can be completed into chordal graphs by adding edges between non-probes. They characterized the subfamily of chordal probe graphs which have no even holes (induced chordless cycles on at least 5 vertices and of even length). Berry, Golumbic and Lipshteyn [2, 3] then solved the general problem for chordal probe graphs, by giving polynomial time recognition algorithms for chordal probe graphs. In doing so, they introduce two new graph superclasses, the  $N$ -triangulatable graphs and the cycle-bicolorable graphs, proving interesting properties characterizing both of them.

In these various instances of  $\mathcal{C}$  probe graphs, several variations of the problem have thus been studied: either the partition into probes and non-probes is given as input (we call this the partitioned version of the problem), or the partition is not given, and the question is whether there exists such a partition (we call this the non-partitioned version of the problem). In both cases, the set of non-probes either is required to be a stable set, or it isn't. For chordal probe graphs, for example, the four problems were solved in [3].

In this paper, we address the issue of recognizing chordal-bipartite probe graphs in the partitioned version of the problem. Moreover, as in [3], we do not necessarily require the set of non-probes to define a stable set, thus also solving the recognition problem on a wider graph class.

The paper is organized as follows: in Section 2, we give the preliminaries needed for our results. In Section 3, we characterize chordal-bipartite probe graphs by an edge elimination scheme, and derive a polynomial time

recognition algorithm, providing a brief proof sketch. Section 4 gives some interesting properties of chordal-bipartite probe graphs, along with a conjectured characterization by cycles. Our conclusions and further open questions appear in Section 5.

## 2 Preliminaries

In this paper, all graphs  $G = (V, E)$  are undirected and finite, with no self loops nor multiple edges. We will denote  $|V| = n$  and  $|E| = m$ . The (open) neighborhood of a vertex  $x$  is denoted by  $N(x)$ , and we use the phrase ‘ $x$  sees  $y$ ’ for  $y \in N(x)$ . For simplicity, we will use informal notations such as  $xy$  to denote the edge joining  $x$  and  $y$ , and  $G + xy$  to denote the graph  $(V, E \cup \{xy\})$ . A stable set (or independent set) is a subset  $I$  of vertices such that no two members of  $I$  are connected by an edge of the graph.

A graph  $G$  is a *bipartite graph* if its vertex set can be partitioned into two stable sets, which we will refer to as the “black/white” bipartition. A subgraph of a bipartite graph is called a *complete bipartite subgraph* if all its black vertices are adjacent to all its white vertices. Finally, we define the neighborhood  $N(xy)$  of an edge  $xy$  as  $N(x) \cup N(y) - \{x, y\}$ . When no confusion arises, we may also use this notation to refer to the subgraph induced by  $N(xy)$ .

A bipartite graph is called *chordal-bipartite* [5, 6] if it contains no induced chordless cycle  $C_k$  of length  $k \geq 5$  (i.e., chordless 4-cycles are permitted). A graph  $G = (V, E)$  is *weakly chordal* [1, 11] if neither  $G$  nor its complement  $\bar{G}$  have an induced subgraph  $C_k$ ,  $k \geq 5$ . It can easily be seen that the chordal-bipartite graphs are precisely the graphs which are both bipartite and weakly chordal. (For more details see [4, 5]).

One of the known characterizations of chordal-bipartite graphs, based on a form of edge elimination, will now be described.

An edge  $e = xy$  of a bipartite graph  $G$  is called *bi-simplicial* if the subgraph induced by  $N(xy)$  is a complete bipartite subgraph. We define a bi-simplicial elimination ordering of the edges as follows:

Let  $G = (V, E)$  be a bipartite graph, and let  $\sigma = (e_1, \dots, e_m)$  be an ordering of the edges. Define  $G_i = (V, E_i)$  where  $E_i = \{e_i, \dots, e_m\}$ , that is,  $G_i$  is the graph obtained by erasing the edges  $e_1, \dots, e_{i-1}$ , but not their endpoints. We call  $\sigma$  a *bi-simplicial elimination ordering* if  $e_i$  is bi-simplicial in  $G_i$  for all  $i$ .

According to [4], several researchers have observed the following:

**Characterization 2.1** A bipartite graph  $G$  is a chordal-bipartite graph if and only if  $G$  has a bi-simplicial elimination ordering (erasing edges but not vertices).

**Remark 2.2** We note that an edge may not remain bi-simplicial from iteration to iteration of the elimination process. For example, all 4 edges of the cycle  $C_4$  start out bi-simplicial, but after one edge is erased, the opposite edge will no longer be bi-simplicial. It regains its bi-simplicial status only in the third iteration.

Chordal-bipartite graphs are also characterized by their minimal separators:

**Characterization 2.3** [6] A bipartite graph  $G$  is a chordal-bipartite graph if and only if every minimal separator of  $G$  induces a complete bipartite subgraph.

A bipartite graph  $G$  is said to be a *chordal-bipartite probe graph* if one can add to  $G$  a set of edges between non-probes such that the resulting graph is chordal-bipartite. We will refer to such a chordal-bipartite graph as a *chordal-bipartite completion* of  $G$ , denoted  $G^*$ , and to the added edges as the *fill edges* of  $G^*$ . Note that the “black/white” bipartition of the vertices and the “probe/non-probe” bipartition of the vertices of a *chordal-bipartite probe graph* are different!

Figure 1 shows a chordal-bipartite probe graph and a corresponding chordal-bipartite completion. The non-probe vertices are represented by squares.

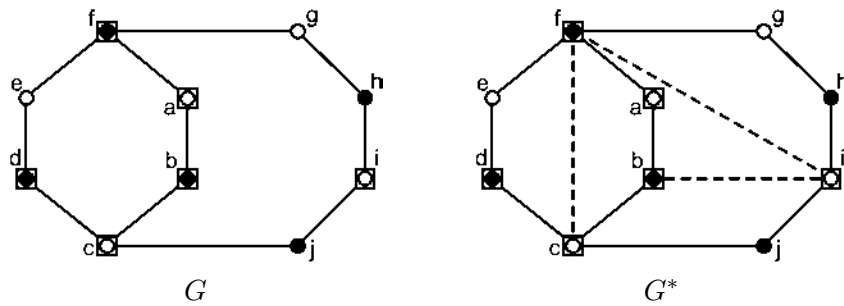


Figure 1: A chordal-bipartite probe graph  $G$  and a chordal-bipartite completion  $G^*$ . The non-probe vertices are represented by squares.

### 3 A characterizing elimination scheme

In this section, we will prove that the recognition of partitioned chordal-bipartite probe graphs is a polynomial problem. We do this by defining an elimination scheme on edges which yields a greedy recognition algorithm and provides a chordal-bipartite completion when the input graph is chordal-bipartite probe.

**Definition 3.1** *Let  $G = (V, E)$  be a bipartite graph. The deficiency of an edge  $ab \in E$  is the set of non-edges in  $N(ab)$ , denoted by  $D_G(ab)$ , i.e.,  $D_G(ab) = \{xy \notin E \mid ay, xb \in E\}$ .*

Clearly, the deficiency of a bi-simplicial edge is the empty set. Furthermore, for any edge  $ab$ , filling in all the non-edges of  $N(ab)$ , a process we will refer to as *bi-saturating  $N(ab)$* , will complete  $ab$  into a bi-simplicial edge. Whether or not  $N(ab)$  can be bi-saturated in a chordal-bipartite graph will depend upon several conditions to be given below.

We will now formulate an interactive procedure to be performed on a bipartite graph  $G$ , called a *bi-saturating elimination scheme by edges*:

- (0) initialize  $G_1 = H_1 = G$ ,
- (1) repeatedly choose (if one can) an edge  $e_i$  in  $G_i$  to make it bi-simplicial (or choose an edge which is already bi-simplicial in  $G_i$ ),
- (2) define the fill edges  $F_i = D_{G_i}(e_i)$  (if any) and create the next transitory graphs  $G_{i+1} = G_i + F_i - e_i$  and  $H_{i+1} = H_i + F_i$ ,
- (3) continue until no edge remains.

The bi-saturating elimination scheme by edges defines an ordering  $\sigma$  on  $E + F$ , where  $F$  denotes the set of fill edges we add to the graph in the bi-saturating process.

The graph  $(V, E + F)$  associated with  $\sigma$  will be denoted  $G_\sigma^+$ .

We now formalize what it means for an edge  $e_i$  to be made bi-simplicial.

**Definition 3.2** *We will say that  $e_i$  is bi-saturable in  $G_i$  if its deficiency  $D_{G_i}(e_i)$  consists of only non-edges whose endpoints are both non-probes **and** if no already processed edge  $e_l$  ( $l < i$ ) belongs to the deficiency of  $e_i$ . Note that  $e_i$  may be an edge of  $F$ , added at some previous step.*

Figure 2 shows a chordal-bipartite probe graph with an ordering  $(e_1, \dots, e_{15})$  and the computed fill edges. Note that the fill computed in this example is not minimal, as an unnecessary fill edge  $e_5 = di$  is added.

A bi-saturating elimination scheme by edges leads us to define a recognition algorithm, which we will give below and then go on to prove.

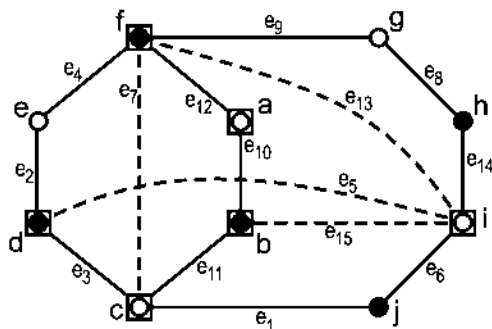


Figure 2: A chordal-bipartite probe graph with a bi-saturating elimination scheme on edges. Processing edge  $e_1$  creates fill edges  $e_5$  and  $e_{15}$ ; processing edge  $e_2$  creates fill edges  $e_7$  and  $e_{13}$ .

**Algorithm** EDGE-RECOGNITION

**Input:** A bipartite graph  $G = (V, E)$  with vertices labeled  $P$  (probes) or  $N$  (non-probes).

**Output:** An answer to the question: Is  $G$  a chordal-bipartite probe graph? And if yes, a certificate in the form of a chordal-bipartite completion of  $G$  associated with an ordering  $\sigma$  on its edges.

**Initialization:**  $G_1 \leftarrow G$ ;  $F \leftarrow \emptyset$ ;  $\sigma$  is an empty queue;  $i \leftarrow 1$ ;

**while**  $G_i$  has at least one edge **do**:

**if**  $G_i$  has no bi-saturable edge **then return**: NO.

**choose** a bi-saturable edge  $e_i$  of  $G_i$ ;

$D_{G_i}(e_i) \leftarrow$  set of edges necessary to add to bi-saturate  $e_i$  in  $G_i$ ;

    //  $D_{G_i}(e_i)$  may be empty

$G_{i+1} \leftarrow G_i + D_{G_i}(e_i) - e_i$ ;  $F \leftarrow F + D_{G_i}(e_i)$ ; add  $e_i$  to  $\sigma$ ;  $i \leftarrow i + 1$ ;

**return**: YES,  $G_\sigma^+ = (V, E + F)$  is a chordal-bipartite completion of  $G$ , and  $\sigma$  is a bi-saturating elimination scheme on edges of  $G$ .

The complexity is at most  $O(m^*n^2)$ , where  $m^*$  is the number of edges of  $G_\sigma^+$ , if to find the next bi-saturable edge you have to examine all the  $O(m^*)$  edges of the transitory graph and spend  $O(m^*)$  to test each for bi-simpliciality. However, an adequate data structure could avoid reexamining the same edge many times for bi-simpliciality.

To prove Algorithm EDGE-RECOGNITION, we will need the following theorem:

**Theorem 3.3** *A bipartite graph  $G$  with the vertices labeled  $P$  or  $N$  has a bi-saturating elimination scheme by edges if and only if  $G$  is chordal-bipartite probe.*

To prove this, we will need several results, and first the following invariant, which ensures that after an edge  $e_i$  is processed and eliminated as bi-simplicial, though a later processing step may add a fill edge  $f$  which will be incident to  $e_i$  in the resulting filled graph, this will not cause any problems; this is because along with  $f$ , fill edges are added which will ensure that  $e_i$  remains bi-simplicial in the elimination process of the filled graph.

**Invariant 3.4** *Let  $G = (V, E)$  be a chordal-bipartite probe graph, let  $\sigma$  be the edge ordering corresponding to a bi-saturating elimination scheme by edges, let  $G_\sigma^+ = (V, E + F)$  be the filled graph obtained, let  $e$  be an edge of  $E + F$ . Then the deficiency of  $e_i$  cannot increase in the course of the elimination process at any step  $j$  after iteration  $i$  in graph  $H_j - \{e_1, \dots, e_{i-1}\} = G_i + (F_i + \dots + F_{j-1})$ .*

**Proof:** Suppose the deficiency of  $e_i = xy$  increases after iteration  $i$ , and that there is a new non-edge  $\{u, v\}$  created in  $N(e_i)$ : there must be at least one fill edge added incident to  $e_i$  during the elimination process at some step  $j > i$ . Let us consider the first such fill edge incident to  $e_i$  which was added at step  $j > i$ , and call it  $xu$ . Let  $e_j = zt$  be the edge whose bi-saturation created edge  $f = xu$  in  $G_j$ . Clearly,  $e_j$  must see  $x$  in the transitory graph  $G_j$ , by edge  $zx$ , which is in  $G_i$  as it cannot be a fill edge added after step  $i$ . Let  $v$  be a neighbor of  $y$  in  $G_i$ ; when  $e_i = xy$  was bi-saturated, fill edge  $zv$  was created if it was not already in the graph. Therefore,  $e_j$  must see  $v$  in  $G_j$ , but then  $e_j$  sees both  $u$  and  $v$  in  $G_j$ , and thus the bi-saturation of  $e_j$  adds fill edge  $uv$ , which contradicts the assumption that  $\{u, v\}$  is in the deficiency of  $e_i$  in  $H_j - \{e_1 \dots e_{i-1}\}$ .

The same arguments apply to the next fill edges defined incident to  $e_i$ .

□

**Property 3.5** *Let  $G = (V, E)$  be a bipartite graph, let  $\sigma$  be the edge ordering defined by a bi-saturating elimination scheme by edges on  $G$ . Then  $\sigma$  is a bi-simplicial elimination scheme on edges of  $G_\sigma^+$ , and  $G_\sigma^+$  is a chordal-bipartite completion of  $G$ .*

**Proof:**  $\sigma$  is a bi-simplicial elimination scheme on edges of  $G_\sigma^+$  by Invariant 3.4. Furthermore, by Characterization 2.1 it is chordal-bipartite, and it is a supergraph of  $G$  obtained by adding only edges between two non-probes, so it is a chordal-bipartite completion of  $G$ .  $\square$

**Lemma 3.6** *Let  $G = (V, E)$  be a chordal-bipartite probe graph, let  $G^*$  be a chordal-bipartite completion of  $G$ , let  $\sigma$  be a bi-simplicial elimination scheme of  $G^*$ . Then  $\sigma$  can be used to define a bi-saturating elimination scheme by edges  $\sigma'$  on  $G$ .*

**Proof:** (sketch)

At Step  $i$  of the construction of  $G_i$ , processing edge  $e_i$ , all the edges of  $G_i$  are edges of  $G_i^*$ , with a number which is greater than  $i$  by  $\sigma$ : that is to say that no edge which is created as fill edge in some  $G_i$  will have been already processed and eliminated in a corresponding elimination process of  $G^*$ . This is because when  $e_i$  is processed in  $G_i^*$ , it must be bi-simplicial.  $G^*$  is not necessarily equal to  $G_\sigma^+$ , (we can have  $G_\sigma^+ \subset G^*$ ), so  $\sigma'$  is a sub-ordering of  $\sigma$ . Obtaining  $\sigma'$  from  $\sigma$  can be done by repeatedly processing the next edge in  $\sigma$  which is in the transitory graph  $G_i$ .  $\square$

**Proof:** (of Theorem 3.3)

By Property 3.5, if a bipartite graph  $G$  has a bi-saturating elimination scheme by edges then it is chordal-bipartite probe. Conversely, by Lemma 3.6, any chordal-bipartite probe graph has a bi-saturating elimination scheme by edges.  $\square$

As a result of this,  $G$  has a chordal-bipartite completion  $G_\sigma^+$ , and the first edge  $e_1$  in  $\sigma'$  is clearly a bi-saturable edge in  $G$ , so if  $G$  is chordal bipartite probe, it has at least one bi-saturable edge. We will now show that, given a chordal-bipartite probe graph  $G$  and a chordal-bipartite completion  $G^*$  of  $G$ , if we choose a saturable edge  $e$  in  $G$  which is not a bi-simplicial edge of the chordal-bipartite completion  $G^*$ , then we can extend  $G^*$  to another chordal-bipartite probe graph which is also a completion of  $G$ , thus ensuring that the greedy approach of Algorithm EDGE-RECOGNITION will work.

The basis for this is the following property, which holds for any bipartite graph:

**Property 3.7** *Let  $G$  be a chordal-bipartite graph, let  $ab$  be an edge of  $G$ ; let  $G'$  be the graph obtained from  $G$  by adding edges to make  $N(ab)$  a complete bipartite subgraph; then  $G'$  is also chordal-bipartite.*

**Proof:** Suppose that adding edges to make  $N(ab)$  a complete bipartite subgraph has created a chordless cycle  $C = (x, y, u, z_1, z_2, z_3 \dots)$  of length  $\geq 6$ , with  $x, y \in N(ab)$ . Clearly, neither  $a$  nor  $b$  can belong to  $C$ .  $C$  can contain at most 3 consecutive vertices which belong to  $N(ab)$ . If  $C$  contains 3 consecutive vertices  $x, y, u$  of  $N(ab)$ , with  $x$  and  $u$  seeing  $a$  and  $y$  seeing  $b$ , then in  $G$  there was a chordless cycle  $(x, a, u, z_1, z_2, z_3)$ . But if only  $x$  and  $y$  are in  $C \cap N(ab)$ , then  $x$  and  $y$  belong to a common minimal separator, separating  $a, b$  from  $u, z_1, z_2, z_3$ . But by Characterization 2.3,  $x$  and  $y$  were already adjacent in  $G$ , which is chordal-bipartite.  $\square$

**Property 3.8** *Let  $G$  be a chordal-bipartite probe graph, let  $e$  be a bi-saturable edge of  $G$ . Then  $G + D_G(e)$  and  $G + D_G(e) - e$  remain chordal-bipartite probe.*

**Proof:** Let  $G^*$  be a chordal-bipartite completion of  $G$ , let  $\sigma'$  be the ordering corresponding to a bi-saturating elimination scheme by edges on  $G$  as in Property 3.6, let  $i$  be the number of  $e$  in  $\sigma'$ . By Property 3.5,  $G_{\sigma'}^+$  is a chordal-bipartite completion of  $G$ . We claim that  $e$  is bi-saturable in  $G_{\sigma'}^+$ . Suppose it is not. Then, as in the proof of Invariant 3.4, there is a non-edge  $\{u, v\}$  in  $N(e)$  in  $G_{\sigma'}^+$ , with  $v$  a probe. A fill edge  $xu$  must have been added incident to  $e$  in the course of the process computing  $G_{\sigma'}^+$ .

Case 1:  $xu$  is added after Step  $i$ : by Invariant 3.4, no non-edge  $\{u, v\}$  can appear.

Case 2:  $xu$  is added before Step  $i$ : let us consider the first edge  $xu$  added incident to  $e$  before Step  $i$ . Let  $e_k = tz, k < i$ , be the edge whose bi-saturation created edge  $xu$  during the construction of  $G_{\sigma'}^+$ :  $e_k$  must see  $e$ , by edge  $zx$ . But since  $zx$  cannot be a fill edge, because  $xu$  was the first fill edge added incident to  $e$  and  $zx$  is present when  $e_k$  is processed. So  $zx$  is in  $G$ , and since  $e$  is bi-saturable in  $G$ , and since  $e$  sees both  $z$  and  $v$ , with  $v$  a probe, then  $vz$  must also be an edge of  $G$ . But then at Step  $k$ ,  $e_k$ , sees both  $v$  and  $u$  and is bi-saturable in the transitory graph, so edge  $uv$  is also in  $G$ , which contradicts the assumption that  $\{u, v\}$  is a non-edge.

Thus  $e$  is bi-saturable in  $G_{\sigma'}^+$ . By Property 3.7, the graph  $G''$  obtained from  $G_{\sigma'}^+$  by bi-saturating  $e$  is chordal-bipartite, and is clearly a chordal-bipartite completion of  $G + D_G(e)$ . In  $G' = G + D_G(e)$  and in  $G''$ ,  $e$  is bi-simplicial, and so when  $e$  is removed from  $G'$ ,  $G'' - e$  is a chordal bipartite completion of  $G'$ , so  $G' - e$  remains chordal-bipartite probe.  $\square$

Thus any bi-saturable edge can be chosen at each step of Algorithm EDGE-RECOGNITION, and the algorithm answers yes if and

only if the input graph was indeed chordal-bipartite probe.

## 4 Properties of Chordal-Bipartite Probe Graphs

In view of the strong structural properties exhibited by the cycles of chordal probe graphs [3], we now investigate the cycles of chordal-bipartite probe graphs.

We will need the following property:

**Property 4.1** *Let  $G = (N + P, E)$  be a partitioned bipartite graph ( $N$  is not necessarily a stable set). If  $G$  is a chordal-bipartite probe graph then the following rules both hold:*

**Rule 1:** *On each chordless cycle of length at least 6 of  $G$ , a probe sees at most one other probe.*

**Rule 2:** *On each chordless cycle of length at least 6 of  $G$ , there is an edge with both endpoints which are probes, or there is an edge with both endpoints which are non-probes.*

**Proof:** By contraposition: let us show that if Rule 1 or Rule 2 is not respected for a partitioned bipartite graph  $G = (N + P, E)$  then  $G$  is not chordal-bipartite probe.

**1.** Suppose that Rule 1 does not hold: there is a chordless cycle of length at least 6 with 3 consecutive probes; it is easy to see that whatever is done to add edges between non-probes, there will remain a chordless cycle of length at least 6 containing the 3 probes. Then  $G$  is not chordal-bipartite probe.

**2.** Suppose that Rule 2 does not hold: there is a chordless cycle of length at least 6 with all edges having as endpoints a probe and a non-probe. In this case, all the non-probes have the same color (black or white), so no fill edge can chord the cycle. Then  $G$  is not chordal-bipartite probe.  $\square$

**Remark 4.2** *If  $N$  is a stable set, Rule 2 can be replaced by the following:*

**Rule 3:** *On each chordless cycle of length at least 6 of  $G$ , there is an edge with both endpoints which are probes.*

Figure 3 shows a graph which is not a chordal-bipartite probe graph.

**Remark 4.3** *Unless the input graph is already chordal-bipartite, the partition into black/white vertices must be different from the partition into probes and non-probes, because of Rule 2.*

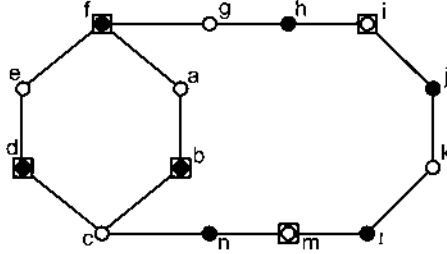


Figure 3: A graph which is not a chordal-bipartite probe graph: cycle  $(f, g, h, i, j, k, l, m, n, c, b, a, f)$  violates Rule 1 as  $j, k, l \in P$ , and cycle  $(a, b, c, d, e, f, a)$  violates Rule 2 as  $b, d, f \in N$ .

We conjecture that Property 4.1 actually characterizes chordal-bipartite probe graphs. If this is true, we have an alternate polynomial-time recognition algorithm for chordal-bipartite probe graphs, which we will discuss below.

To find a cycle violating Rule 1, for each  $PP$  edge  $e$ , compute the minimal separators (called the substars of  $e$ ) in the neighborhood of  $e$  (which costs  $O(m)$  per  $PP$  edge, as explained in [1]);  $e$  is on a chordless cycle of length at least 6 with each pair of non-adjacent vertices belonging to a common substar of  $e$ , so if one endpoint  $x$  of such a non-edge  $x, y$  is a probe, we know that the graph fails to be chordal-probe. For a certificate cycle violating Rule 2, simply find a path  $\mu$  from  $x$  to  $y$  in  $G(V - (e \cup N(e)))$ : the cycle will be constituted of  $x, e, y$  and  $\mu$ . This phase costs  $O(mp)$ , where  $p$  is the number of  $PP$  edges in the input graph  $G$ . If the input graph  $G$  fails to be chordal-bipartite probe, a certificate can be given easily with any cycle containing the incriminated  $PP$  edge and its pair of non-adjacent neighbors, one of which is a probe.

To find a cycle violating Rule 2, first remark that if there is a chorded cycle where  $N$  and  $P$  alternate, each chord is an  $NP$  edge, so no chordless cycle of length more than 5 where  $N$  and  $P$  alternate can be created by removing all  $PP$  edges and all  $NN$  edges. Therefore, we can remove all such  $PP$  edges and all  $NN$  edges, obtaining graph  $G^o$ , then test in  $O(n^2)$  whether  $G^o$  is chordal-bipartite ([16]). The input graph  $G$  violates Rule 2 if and only if  $G^o$  fails to be chordal-bipartite, and it is easy to produce a certificate cycle with alternating probes and non-probes.

The algorithm thus runs in  $O(mp)$  time, with  $p$  the number of  $PP$  edges in the input graph.



defined for chordal probe graphs [3], using both elimination schemes and labeling rules on the vertices of chordless cycles of length at least 5.

In the non-partitioned case (where the partition into probes and non-probes is not given), if  $N$  does not have to be a stable set, then we only need to say that  $V = N$  and embed the graph into a complete bipartite graph to solve the problem. In the case where  $N$  is required to be a stable set, a bipartite graph is not always chordal-bipartite probe, as shown in Figure 4.

We leave open the question of recognizing chordal-bipartite probe graphs in the non-partitioned case.

## References

- [1] A. Berry, J-P. Bordat, and P. Heggernes. Recognizing weakly triangulated graphs by edge separability. *Nordic Journal Computing*, 7:3:164-177, 2000.
- [2] A. Berry, M.C. Golumbic, and M. Lipshteyn. Two tricks to triangulate chordal probe graphs in polynomial time. *Proc. SODA '04*, 962-969, 2004.
- [3] A. Berry, M.C. Golumbic and M. Lipshteyn. Cycle-bicolorable graphs and triangulating chordal probe graphs. *SIAM Journal on Discrete Mathematics*, to appear.
- [4] A. Brändstadt, V. Le, J.P. Spinrad. *Graph classes: A Survey*. SIAM Monographs on Discrete Math. and Applications, Philadelphia, 1999.
- [5] M.C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs*. Second edition: *Annals of Discrete Math.* 57, Elsevier, Amsterdam, 2004.
- [6] M.C. Golumbic and C.F. Goss. Perfect elimination and chordal-bipartite graphs. *Journal of Graph Theory*, 2:155-163, 1978.
- [7] M.C. Golumbic, H. Kaplan and R. Shamir. Graph sandwich problems. *Journal of Algorithms*, 19:449-473, 1995.
- [8] M.C. Golumbic and M. Lipshteyn. Chordal probe graphs (extended abstract). *Proceedings of WG'03, LNCS*, 2880:249-260, 2003.
- [9] M.C. Golumbic and M. Lipshteyn. Chordal probe graphs. *Discrete Applied Mathematics*, 143:221-237, 2004.
- [10] M.C. Golumbic and A.N. Trenk. *Tolerance Graphs*. Cambridge University Press, 2004.

- [11] R. Hayward. Weakly triangulated graphs. *J. Combin. Theory B*, 39:200-208, 1985.
- [12] J.L. Johnson and J.P. Spinrad. A polynomial time recognition algorithm for probe interval graphs. *Proc. SODA '01*, 477-486, 2001.
- [13] R.M. McConnell and J.P. Spinrad. Construction of probe interval models. *Proceedings of SODA '02*, 866-875, 2002.
- [14] F.R. McMorris, C. Wang and P. Zhang. On probe interval graphs. *Discrete Applied Mathematics*, 88:315-324, 1998.
- [15] L. Sheng. Cycle free probe interval graphs. *Congressus Numerantium*, 140:33-42, 1999.
- [16] J.P. Spinrad. Doubly lexical ordering of dense 0-1 matrices. *Inform. Process. Lett.*, 45:229-235, 1993.
- [17] P. Zhang. Probe interval graphs and their application to physical mapping of DNA. *Manuscript*, 1994.
- [18] P. Zhang, E.A. Schon, S.G. Fischer, E. Cayanis, J. Weiss, S. Kistler, and P.E. Bourne. An algorithm based on graph theory for the assembly of contigs in physical mapping of DNA. *CABIOS*, 10:309-317, 1999.
- [19] P. Zhang, X. YE, L. Liao, J. Russo, and S.G. Fischer. Integrated mapping package - a physical mapping software tool kit. *Genomics*, 55:78-87, 1999.