AUSTRALASIAN JOURNAL OF COMBINATORICS
Volume **52** (2012), Pages 19–31

Forbidden subgraph characterization of bipartite unit probe interval graphs

David E. Brown

Department of Mathematics and Statistics
Utah State University
Logan, UT 84341-3900
U.S.A.
david.e.brown@usu.edu

Larry J. Langley

Department of Mathematics
University of the Pacific
Stockton, CA 95211
U.S.A.
llangley@pacific.edu

Abstract

A graph is a *probe interval graph* (PIG) if its vertices can be partitioned into probes and nonprobes with an interval assigned to each vertex so that vertices are adjacent if and only if their corresponding intervals intersect and at least one of the vertices is a probe. When all intervals have the same length (or equivalently, no interval contains another properly) the graph is a unit probe interval graph. We characterize, via a list of minimal forbidden induced subgraphs, and no *a priori* partition of vertices into probes and nonprobes, the class of bipartite unit probe interval graphs and present a linear time recognition algorithm for them which is based on this characterization.

1 Introduction

We discuss undirected finite simple graphs G with vertex set $V(G)$ and edge set $E(G)$, or $G = (V, E)$ may be written meaning $V = V(G)$ and $E = E(G)$. A graph G is a *probe interval graph* if there is a partition of $V(G)$ into sets P and N and a collection $\{I_v : v \in V(G)\}$ of intervals of $\mathbb{R}$ such that, for $u, v \in V(G)$, $uv \in E(G)$ if and only if $I_u \cap I_v \neq \emptyset$ and at least one of u, v belongs to P. The sets P and N

are called the *probes* and *nonprobes*, respectively, and $\{I_v = [l(v), r(v)] : v \in V(G)\}$ together with the partition into probes and nonprobes will be referred to in this paper as a *representation.* If, for G a probe interval graph, the members of $\{I_v : v \in V(G)\}$ are closed intervals of identical length, then G is a *unit probe interval graph.* An *interval graph* is a probe interval graph with $N = \emptyset$ and this class of graphs has been studied extensively, see the texts by Fishburn [6], Golumbic [9], and Roberts [16] for introductions and other references. Here we study the unit probe interval graphs and characterize the bipartite graphs that are unit probe interval graphs by a list of minimal forbidden induced subgraphs and give a recognition algorithm for them which has $\mathcal{O}(|V(G)| + |E(G)|)$ complexity for input graph G. Note that our characterization requires only that the graph G under consideration be bipartite, but does not require the partition of $V(G)$ into probes and nonprobes be given.

The probe interval graph model was invented in order to aid with the task called *physical mapping* faced in connection with the human genome project, cf. work of Zhang and Zhang et al. [19, 20, 21]. In DNA sequencing projects, a *contig* is a set of overlapping DNA segments derived from a single genetic source. In order for DNA to be more easily studied, small fragments of it, called clones, are taken from multiple copies of the same genome. Physical mapping is the process of determining how DNA contained in a group of clones overlap without having to sequence all the DNA in the clones. Once the map is determined, the clones can be used as a resource to efficiently contain stretches of genome. If we are interested in overlap information between each pair of clones, we can use an interval graph to model this problem: vertices are clones and adjacency represents overlap. Using the probe interval graph model we can use any subset of clones, label them as probes, and test for overlap between a pair of clones if and only if at least one of them is a probe. This way there is flexibility, in contrast to the interval graph model, since all DNA fragments need not be known at time of construction of the probe interval graph model. Consequently, the size of the data set, which by nature can be quite large, is reduced.

We consider probe interval graphs as a combinatorial problem and focus on their structure. Here is a brief discussion of some of the recent results on the structure of probe interval graphs and where to find them. The paper by McMorris, Wang and Zhang [14] has results similar to those for interval graphs found in [7] by Fulkerson and Gross and [9] by Golumbic; e.g., probe interval graphs are weakly chordal, analogous to interval graphs being chordal, and, as maximal cliques are consecutively orderable in interval graphs, so-called quasi-maximal cliques are in probe interval graphs (see [14]). The classes of graphs related to probe interval graphs is discussed in [4] by Brown and Lundgren, [2] by Brown, Flink and Lundgren, and [10] by Golumbic and Lipshteyn. Relationships between bipartite probe interval graphs, interval bigraphs and the complements of circular arc graphs are presented in [4].

Some results regarding recognition of whether a given graph is a probe interval graph have been obtained, however nearly all require that the partition of vertices into probes and nonprobes be given in advance. A graph in which the probe/nonprobe partition has been specified is called a *partitioned probe interval graph.* The problem of recognizing whether a given n-vertex graph $G = (V, E)$ with m edges and partition

$V = (P, N)$ is a partitioned probe interval graph (with probes being P) is solvable in time $\mathcal{O}(n^2)$ via a method involving modified PQ-trees, see [11], by Johnson and Spinrad. Another method given in [13] by McConnell and Spinrad uses modular decomposition and has complexity $\mathcal{O}(n+m \log n)$. Finally, the problem of recognizing whether a given graph with no partition specified is a probe interval graph has been treated in [5] and the algorithm is of polynomial order, but specific complexity is not addressed.

One way to characterize and describe the structure of a class of graphs is via a complete list of minimal forbidden induced subgraphs; indeed, such a characterization is often tantamount to an efficient recognition algorithm, as will be the case here. A requisite for such a characterization for a class of graphs is that any induced subgraph of a graph from the class still belongs to the class. It is easy to see that this is the case for probe interval graphs and unit probe interval graphs. The problem of characterizing generic probe interval graphs for now appears to be difficult, whence this paper's restriction to the unit and bipartite case. We explain the current situation by mentioning some results regarding a forbidden induced subgraph characterization for generic probe interval graphs. The paper [15] by Pržulj and Corneil suggests that the problem of determining a complete list of forbidden induced subgraphs for generic probe interval graphs may be difficult; specifically, in [15] it is shown that the probe interval graphs that are 2-trees have at least 62 distinct minimal forbidden induced subgraphs. In [4] various characterizations are given for the probe interval graphs that are bipartite, but no forbidden induced subgraph characterization is given. However, a somewhat large list of forbidden induced subgraphs is conjectured for the bipartite probe interval graphs in which the probe/nonprobe partition corresponds to a bipartition in [4].

On a more positive note, in [1] a linear time recognition algorithm and structural characterization is given for bipartite probe interval graphs which is based on their close relationship to bipartite tolerance graphs. Also, the trees that are probe interval graphs (with no specified probe/nonprobe partition) have been characterized via a list of two forbidden induced subgraphs in [17]. The cycle-free unit probe interval graphs have been characterized via the following theorem which we will use here:

Theorem 1.1 (Brown, Lundgren, Sheng, [3]) *Let G be a cycle-free graph. G is a unit probe interval graph if and only if it has no induced subgraph isomorphic to any of Figure 1.*

In summary, the cycle-free probe interval graphs and cycle-free unit probe interval graphs are the only classes of probe interval graphs or unit probe interval graphs which enjoy a characterization by forbidden induced subgraphs. The purpose of this paper is to prove the following theorem and give a forbidden induced subgraph characterization for bipartite unit probe interval graphs.

Theorem 1.2 *A bipartite graph G is a unit probe interval graph if and only if it has no chordless cycle on more than 4 vertices, none of the graphs of Figure 2, and no graph of Figure 1 as an induced subgraph of G.*

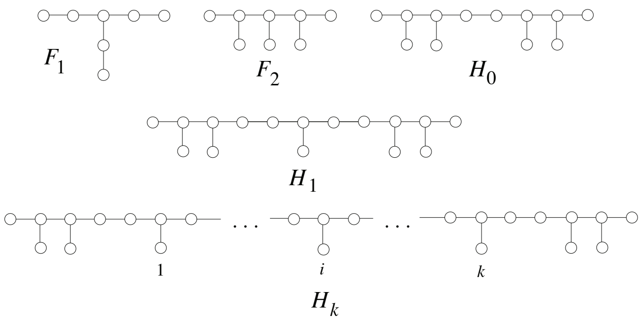

Figure 1: The forbidden induced subgraphs that characterize cycle-free unit probe interval graphs.

A *proper probe interval graph* is a probe interval graph that has a representation in which no interval contains another properly. Clearly a unit probe interval graph is a proper probe interval graph and the following theorem shows that the converse relationship holds. Thus, we may use either perspective: unit or proper representation.

Theorem 1.3 (Lipshteyn, [12]) *A graph is a unit probe interval graph if and only if it is a proper probe interval graph.*

As this paper is written, the characterization for non-bipartite unit probe interval graphs remains elusive, let alone the problem of characterizing the generic probe interval graphs.

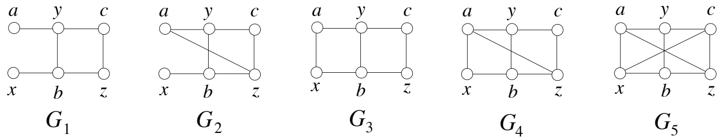

Figure 2: Forbidden induced subgraphs for bipartite unit probe interval graphs

2 Proof of Theorem 1.2

We will henceforth use UPIG to stand for 'unit probe interval graph'. For G a graph and $S \subseteq V(G)$, we denote by $G(S)$ the subgraph of G induced by S. We use $\mathcal{I}(S)$ to denote the interval representation for the graph induced on S. If G is given a partition, regardless of whether it is a probe interval graph, we may write $G = (P, N, E)$, and mean that P is the set of probes and N the nonprobes; in this

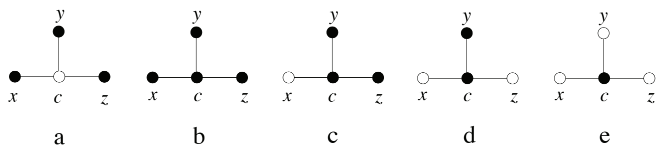

Figure 3: Darkened vertices represent probes. The first three are not possible induced subgraphs for unit probe interval graphs; the last two are.

case we say G *is partitioned*. Note that if $G = (P, N, E)$ is a probe interval graph then $G(P)$ is an interval graph, and $G(N)$ is an independent set.

For the interval corresponding to vertex v, we will use $I_v = [l(v), r(v)]$. Since every graph we consider is simple and finite, we will always initially assume that all intervals are distinct and that no two intervals share an endpoint, except in some constructions where using copies of an interval will be helpful.

The *neighborhood* of a vertex v in graph G is the set $\{u : uv \in E(G)\}$ and is denoted $N(v)$. The *closed neighborhood* of v is $N[v] = N(v) \cup \{v\}$. An *asteroidal triple* in a graph G is a set of three vertices with a path between any two that does not intersect the neighborhood of the third.

We now develop some results that will help us prove Theorem 1.2. The next result shows that, like interval graphs, unit probe interval graphs have no asteroidal triples in any induced subgraph.

Lemma 2.1 *If G is a unit probe interval graph, then G has no asteroidal triple in any induced subgraph.*

Proof. Suppose $G = (P, N, E)$ is a unit probe interval graph with an asteroidal triple on the vertices x, y and z, and that P_{xy} is a path from x to y with $N(z) \cap P_{xy} = \emptyset$. Since I_z is neither contained in nor contains another interval corresponding to a vertex on P_{xy}, I_z intersects at least two overlapping intervals from P_{xy}. Both of these intervals cannot correspond to nonprobes, so regardless of z's membership to P or N, z is adjacent to a vertex on P_{xy}, contradicting the assumption that $N(z) \cap P_{xy} = \emptyset$.■

Lemma 2.2 *If G is a unit probe interval graph, then in any induced $K_{1,3}$, the center vertex must be a probe, and at most one of the pendant vertices is a probe. (See Figure 3).*

Proof. Let G be a unit probe interval graph with induced subgraph H isomorphic to $K_{1,3}$ with center c and pendants x, y, z. Since $K_{1,3}$ is not a unit interval graph, $\{c, x, y, z\} \cap N \neq \emptyset$. If $c \in N$, then $x, y, z \in P$, but then I_x, I_y, I_z must be disjoint and each must have a nonempty intersection with I_c. Since this is impossible, $c \in P$.

Suppose two of x, y, z are probes, say, $x, y \in P$. Then we cannot have $l(x) < l(c)$ and $l(y) < l(c)$, otherwise $I_x \cap I_y \neq \emptyset$ and $xy \in E(G)$. So let us assume $l(x) < l(c) <$

$l(y)$, and of course $l(y) < r(c)$ and $l(c) < r(x)$. But then $l(z) \in I_x$ or $r(z) \in I_y$, giving $xz \in E(G)$ or $yz \in E(G)$, a contradiction either way.

It is easy to verify that Figure 3(d) and Figure 3(e) have unit probe interval representations. ■

Corollary 2.1 *If G is a bipartite unit probe interval graph, then any vertices of degree greater than or equal to 3 must be probes.*

Proof. Since G is bipartite then the closed neighborhood of any vertex of degree 3 or higher includes an induced $K_{1,3}$. ■

Now we present a few lemmas that will identify the cycle structure of a unit probe interval graph. The first result we mention is due to McMorris, Wang, and Zhang, and says that the largest induced cycle in a probe interval graph is a 4-cycle; in other words (unit) probe interval graphs are *weakly chordal*. Lemma 2.2 will extend this result for us by showing that there are no cycles other than 4-cycles in a bipartite unit probe interval graph.

Theorem 2.1 (McMorris, Wang, Zhang, [14]) *Probe interval graphs are weakly chordal.*

Lemma 2.3 *In any induced 4-cycle of a probe interval graph, two nonadjacent vertices must be nonprobes. In other words, if G is a probe interval graph, then any induced 4-cycle of G must be as in Figure 4(c).*

Proof. Let $\langle a, b, c, d, a\rangle$ be an induced 4-cycle in probe interval graph G. Since a 4-cycle is not an interval graph, at least one of a, b, c, d must be a nonprobe, while no three can be all nonprobes, since nonprobes induce an independent set in G. Relabeling if necessary, assume $l(a) < l(b) < l(c)$, $l(b) < r(a)$, $l(c) < r(b)$, and that d is the only nonprobe. Since I_d must intersect both I_a and I_c, we have $r(a), l(c) \in I_d$, and hence $I_b \cap I_d \neq \emptyset$, a contradiction. Therefore exactly two vertices are nonprobes, and they are nonadjacent by definition. ■

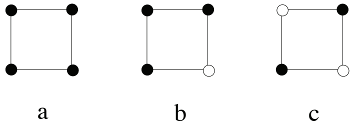

Figure 4: Darkened vertices represent probes. The first two are not possible induced subgraphs of a probe interval graph; the last one is.

Lemma 2.4 *G_1, G_2, G_3, G_4, and G_5 of Figure 2 are not induced subgraphs of any unit probe interval graph.*

Proof. Assume G_i is a unit probe interval graph, for $i = 1, \ldots, 5$. $G_i - \{a, c\} \cong K_{1,3}$ with b as the center and hence must be a probe by Lemma 2.2. But also $G_i - \{x, y\} \cong K_{1,3}$ with y as the center and hence a probe. But then b and y are adjacent probes in the induced 4-cycle $G_i - \{a, x\}$, a contradiction to Lemma 2.3. ■

We are now ready to characterize the forbidden induced subgraphs for bipartite UPIGs. To this end, we now look at the structure of graphs that contain none of the forbidden induced subgraphs of bipartite UPIGs. In particular, the next two lemmas help to describe the cycle structure in a bipartite graph with no induced subgraph isomorphic to any of those of Figure 2 and no chordless cycle on more than 4 vertices.

Lemma 2.5 *If G is a bipartite graph with no induced subgraph isomorphic to any of Figure 2 then in any 4-cycle in G, no adjacent vertices both have degree greater than two.*

Proof. Suppose G is a bipartite graph, and a, b, c, d is a 4-cycle in G, in that order, with $\deg(a) \geq 3$ and $\deg(b) \geq 3$. Let x be a vertex adjacent to a, not in the 4-cycle. Since G is bipartite, x cannot be adjacent to b, so there is a vertex y, not in the 4-cycle, adjacent to b, not adjacent to a. Among these six vertices there are only three remaining possible edges: $\{x, c\}$, $\{y, d\}$ and $\{x, y\}$, so eight possible subgraphs of G. Each of these is isomorphic to one of the graphs in Figure 2. ■

Next we will show that the graph contains no cycles other than 4-cycles.

Lemma 2.6 *If G is a bipartite graph with no induced subgraph isomorphic to any of the graphs in Figure 2 or a chordless cycle on more than four vertices, then G has no cycles on more than four vertices.*

Proof. Suppose G is a bipartite graph as stated with a cycle on more than 4 vertices, $v_1, v_2, \ldots, v_n$. This cycle must contain some chord: $\{v_i, v_j\}$. Each of these vertices has degree at least three, and together are in a cycle with other vertices of G. Consider a minimal cycle containing v_i and v_j. This cycle must have four vertices, however this contradicts Lemma 2.5. ■

Corollary 2.2 *If G is a bipartite unit probe interval graph, then the only cycles in G are 4-cycles.*

It follows immediately from Lemmas 2.5 and 2.6, that 4-cycles in G must be contained in an induced $K_{2,m}$ with $m \geq 2$, but G can contain no more complex a cycle structure. For reference, if we are considering a specific $K_{2,m}$ let V_1 be the partite set with two vertices and V_2 be the partite set with m vertices. If $m = 2$, then we have a 4-cycle and V_2 will contain two non-adjacent vertices of degree 2 in G. Note that even when $m > 2$ every vertex of V_2 will be degree 2 in G.

To represent a $K_{2,m}$ as a unit probe interval graph, we will put $V_1 = P$, and assign the intervals $[0, 1]$ and $[2, 3]$ to its two vertices. Now put $V_2 = N$ and assign the interval $[1, 2]$ to each vertex. See Figure 5 in which a $K_{2,3}$ is represented. The idea of the following proof is to take advantage of the duplicate nature of those nonprobe intervals. Basically, for each $K_{2,m}$ remove all but one vertex of V_2, making G a tree. Find a representation of this tree that guarantees each vertex of each V_1 is probe and the vertices of V_2 still remaining are nonprobes. We can then duplicate the intervals of the vertices in each V_2 to find a representation of G.

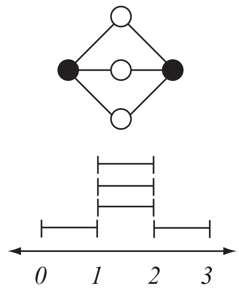

Figure 5: Darkened vertices represent probes and form the vertices of V_1. The remaining vertices are in V_2.

We now have enough background to prove Theorem 1.2.

Proof. (of Theorem 1.2) ($\Rightarrow$) It follows from Theorem 2.3, Lemma 2.4 and Theorem 1.1 that chordless cycles of length greater than 4, and the graphs in Figure 1 and Figure 2 are not induced subgraphs of a bipartite unit probe interval graph.

($\Leftarrow$) Suppose G is a bipartite graph with no induced subgraph isomorphic to any of Figure 1, Figure 2 or a chordless cycles of length greater than 4. We may assume G is connected since otherwise we may find a representation for each component of G separately. If G has no 4-cycles then G is cycle free and by Theorem 1.1 we know G is a UPIG. By Lemma 2.5 and Lemma 2.6 all 4-cycles in G occur within an induced $K_{2,m}$ where $m \geq 2$. We have already described an interval representation for $K_{2,m}$, consequently we may assume G is not isomorphic to a single copy of $K_{2,m}$.

Consider the vertex set of G. Let X be the set of vertices x in G in a $K_{2,m}$, $m \geq 2$, with neighborhood of x not entirely contained in a single $K_{2,m}$. Let X' be the set of all vertices x' such that x' is in a $K_{2,m}$ and is not adjacent to a vertex of X. Let Y be the set of vertices of degree 2 contained in a $K_{2,m}$, excluding any vertices of X'. That is, each vertex in the set V_1 is contained in $X \cup X'$ and each of the m vertices of V_2 is contained in Y. Any 4-cycle will alternate between vertices in X or X' and vertices in Y. Let G' be a tree constructed from G in the following way: In each $K_{2,m}$ remove all but one vertex of Y and the incident edges. Next construct a tree G'' as follows: For every vertex in X of degree 2 in G' attach a pendant vertex. Call the set of these new vertices Z. Observe the following: Every vertex of X will have degree at least 3 in both G and G'' and degree at least 2 in G'. Such vertices in X are adjacent to at most one vertex in Z. Every vertex of X' has degree 2 in G but degree 1 in G''. All remaining vertices of G that are not removed when constructing

G' will have the same degree in both G and G''. We claim G'' is a unit probe interval graph and will then show G is a unit probe interval graph.

Since G' is an induced subgraph of G, it contains no induced subgraph isomorphic to F_1, and hence G' is a caterpillar. Since G'' is constructed from G' by attaching vertices of degree 1 to vertices of degree 2, G'' is also a caterpillar and so contains no induced subgraph isomorphic to F_1 either.

Suppose that G'' contains an induced subgraph isomorphic to F_2 in Figure 1. The three vertices of degree 3 in this copy of F_2 must be degree three in G. Since no four cycle of G may contain adjacent vertices of degree 3 or greater, this set of vertices together with their neighbors contains a subgraph isomorphic to F_2 in G, which again contradicts our assumptions regarding G.

Finally, suppose that G'' contains one of the graphs in Figure 1 isomorphic to H_k, $k \geq 0$. Let H be such an induced subgraph of least size. Since H is not an induced subgraph of G, H must include some vertex of Z and consequently at least one vertex of X. Consider any vertex y in G'' with $y \in Y$ and $x_1, x_2 \in X \cup X'$, both adjacent to y. We have observed that y must be degree 2 in G'' and in G. If either x_1 or x_2 is in X' it is degree 1 and adjacent a vertex of degree 2, so cannot be contained in H. If both x_1 and x_2 are in X they are degree at least three in G''. Suppose H contains x_1, y, and x_2. Since x is degree 2, by examining the structures in Figure 1 we conclude that one of x_1 or x_2 must be degree 2 in H. Without loss of generality, we will assume x_1 is this vertex. So, there is some w which is a neighbor of x_1 contained in G'', but not H. By attaching this vertex we find either an induced F_2 or an induced H_i, $i \geq 0$, smaller than H, contradicting our choice of H. Therefore H cannot contain both of x_1 and x_2, and y must be degree 1 in H.

Consider $x \in X$ adjacent to $z \in Z$ with both x and z in H. Since G'' is a caterpillar, x is adjacent to at most 2 vertices of degree greater than 1. One of these must be a vertex y in Y and hence at most degree 1 in H. In each H_i there are two vertices that are adjacent to vertices of degree 1 while being adjacent to at most one vertex of degree greater than 1. The vertex x must be one of those vertices. There is some vertex y' of Y which was removed from G when constructing G'. Remove z from H but place y' in H. Since the other neighbor of y, and hence y' is not in H, y' has degree 1. If necessary, repeat this process with any other vertex of H adjacent to a vertex in Z. This new graph will be isomorphic to H but entirely contained in G, a contradiction.

We have proven that G'' is a tree containing no induced subgraph isomorphic to a graph in Figure 1 and thus G'' is a unit probe interval graph. Since any vertex of X is degree 3 or more, it must be probe in any valid probe-nonprobe partition of G''. Consider any vertex x' in X'. In G'', x' is degree 1, adjacent to a vertex y in Y of degree 2, which is, in turn, adjacent to a vertex x in X of degree 3 or more. Since x is probe and x' is not adjacent to x, the interval for x' doesn't overlap the interval for x in any representation of G''. If y is nonprobe, x' must be probe. On the other hand if y is probe, no intervals may intersect y other than the intervals for x and x', and in order for G'' to be connected all intervals, other than for those three vertices, must be on the opposite side of y from x'. Consequently no interval intersects the interval for x' and we may make x' probe. So G'' has a partition where every vertex in X or X' is probe. If y in Y is probe, its interval only intersects the intervals for its two neighbors in X or X', so y may be made non-probe instead. To find a representation for G, start with G'' where every vertex in Y is a nonprobe, remove intervals for vertices in Z to form a representation of G'. Now make enough duplicates of intervals for each vertex in Y to replace the missing vertices. We have a unit probe interval representation for G. ■

Algorithm 1 G is a bipartite unit probe interval graph.

Input: Connected bipartite G
Output: TRUE or FALSE

procedure SpineSearch
$a \leftarrow 0$, $b \leftarrow 0$, $i \leftarrow 0$, order H' as $v_0, v_1, \ldots, v_k$, set Spine $= H'$
for $i \leftarrow 0$ to k and Spine $\neq \emptyset$ **do** remove v_i from Spine,
  case $(a, b, \deg_G(v_i))$ **of**
    $(<2, 0, 2)$: $a \leftarrow 0$, $i \leftarrow i+1$.
    $(<2, 0, >2)$: $a \leftarrow a+1$, $i \leftarrow i+1$.
    $(2, 0, 2)$: $b \leftarrow b+1$, $i \leftarrow i+1$.
    $(2, 0, >2)$: **return** FALSE.
    $(2, 1, 2)$: $b \leftarrow b+1$, $i \leftarrow i+1$.
    $(2, 1, >2)$: $a \leftarrow 1$, $b \leftarrow 0$, $i \leftarrow i+1$.
    $(2, 2, 2)$: $a \leftarrow 0$, $b \leftarrow 0$, $i \leftarrow i+1$.
    $(2, 2, >2)$: $a \leftarrow a+1$, $b \leftarrow 0$, $i \leftarrow i+1$.
    $(3, 0, 2)$: $a \leftarrow 2$, $b \leftarrow 1$, $i \leftarrow i+1$.
    $(3, 0, >2)$: **return** FALSE.
    endcase
  endfor
return TRUE
end of SpineSearch
procedure BUPIG
Find all blocks B_i (say $1 \leq i \leq k$) and cut-vertices C of G
and construct the block cut-point graph T of G
If $T - \{B_i : B_i$ contains a pendant of $G\}$ is not a path **return** FALSE
for each block B_i of G
  if B_i is not a K_1 a K_2 or a $K_{2,m}$ **return** FALSE
    else if $B_i \cong K_{2,m}$ and contains adjacent cut-vertices **return** FALSE
    else if $B_i \cong K_{2,m}$ with $m > 2$ and $V_2 \cap C \neq \emptyset$ **return** FALSE
    endif
  endfor
Let $Q = \left(\bigcup_{i=1}^{k} V(B_i)\right) \setminus C$ and construct $H = G - (P_G \cup Q)$
for each component H' of H
  if SpineSearch(H') returns FALSE **return** FALSE
    endif
  endfor
return TRUE
end of BUPIG

Figure 6: Algorithm A: Bipartite unit probe interval graph recognition algorithm.

3 Recognition Algorithm

Theorem 1.2 is conducive to a linear-time recognition algorithm for bipartite UPIGs. Algorithm 1 of Figure 6 is based mostly on the block structure of bipartite UPIGs and consequently the algorithm's complexity is $\mathcal{O}(n+m)$, where we are given a connected bipartite graph G with $n = |V(G)|$ and $m = |E(G)|$.

We use the typical notation where by $G - S$, if $S \subseteq V(G)$, we mean the graph induced on $V(G) \setminus S$. We use P_H to denote the set of vertices of degree one in graph H. Recall that a *block* of a graph is a maximal subgraph that has no cut-vertex, and note that if H is a block, then H is 2-connected or $|V(H)| \leq 2$. For an induced subgraph isomorphic to $K_{2,m}$ with $m > 2$, we use V_2 to denote the partite set of size m, as in the remarks before the proof of Theorem 1.2. Recall that the *block cut-point* graph T of a graph G is the bipartite graph with bipartition $V(T) = C \cup B$, where C is the set of cut vertices of G, B the set of blocks of G, and, for $c \in C$ and $b \in B$, $cb \in E(T)$ if and only if b contains c in G.

The following corollary, which follows from the results in this paper, is the key to the correctness of Algorithm 1.

Corollary 3.1 *Let G be a bipartite unit probe interval graph, B a block of G, and C the set of cut-vertices of G. Then*

1. *Any block of G is either a K_1 a K_2 or a $K_{2,m}$, with $m \geq 2$;*
2. *No nontrivial block B contains cut-vertices which are adjacent;*
3. *If $V(B) = V_1 \cup V_2$ and $B \cong K_{2,m}$ with $|V_2| = m > 2$, then V_2 cannot contain a cut-vertex of G;*
4. *No non-trivial block can contain more than two cut-vertices;*
5. *The block cut-point graph T of G is a caterpillar, with $\deg_T(v) = k > 2$ only if at least $k-2$ of the blocks to which v is incident are K_2's;*
6. *With $Q = \left(\bigcup_{\text{blocks of } G} V(B)\right) \setminus C$, then $H = G - (P_G \cup Q)$ is a forest of paths.*

Proof. The first item follows because otherwise G contains G_3, G_4, or G_5 of Figure 2 as an induced subgraph, or has a cycle structure contrary to Corollary 2.2. If the second item were not true, then G contains an induced G_1 from Figure 2. Item 3 follows because otherwise G contains an induced G_2 of Figure 2.

If a non-trivial block B contains more than two cut vertices, say c_1, c_2, c_3 then there are induced paths of length at least 1 from each c_i into blocks of G other than B. Say P_i consists of $c_i, v_{i,1}, \ldots, v_{i,k}$, $1 \leq i \leq 3$, $k \geq 1$. Then, since G is bipartite and the paths P_i are contained in separate blocks, $\{v_{1,1}, v_{1,2}, v_{1,3}\}$ is an asteroidal triple of G. For the fifth item we know, from item 4, that a cut-vertex of G cannot be incident with more than two non-trivial blocks of G. The sixth item is simply an observation that follows from the preceding items. ∎

Theorem 3.1 *Algorithm 1 is correct and requires $O(n+m)$ steps.*

Proof. Let G be a connected bipartite graph with $|V(G)| = n$ and $|E(G)| = m$ to which Algorithm 1 is applied.

The sub-routine SpineSearch returns FALSE at line 4.4 if G contains an induced F_2 from Figure 1, and returns FALSE at line 4.10 if G contains an induced H_i, for $i \geq 0$, from Figure 1. That SpineSearch is examining the spine of an induced caterpillar, as in the proof of Theorem 1.2, follows from that theorem and from Corollary 3.1 (6).

The step at line 7 checks that G has no asteroidal triples and not too much branching, by Corollary 3.1 (4) and (5). Line 9, 10, and 11, follow via Corollary 3.1 (1), (2), and (3), and essentially line 9 checks that G has no cycles on more than 4 vertices, and hence that G has no G_3, G_4 or G_5 of Figure 2, while lines 10 and 11 check that G has no G_1 or G_2 of Figure 2. Therefore, if Algorithm 1 returns TRUE, G has no forbidden structures, and conforms to the conditions, of Theorem 1.2.

It remains to show that Algorithm 1 has $O(n+m)$ complexity. Clearly the sub-routine SpineSearch requires $O(n+m)$ time as do the steps at 9, 10, 11, and 12 given that the blocks, cut-vertices and block cut-point graph has been constructed, which can be done in $O(n+m)$ time, [8, 18]. ■

References

[1] D. E. Brown, A. H. Busch and G. Isaak, Linear time recognition algorithms and structure theorems for bipartite tolerance graphs and bipartite probe interval graphs, *Discrete Math. Theor. Comput. Sci.* **12** (2010), no. 5, 63–82.

[2] D. E. Brown, S. C. Flink and J. R. Lundgren, Interval k-graphs, *Congressus Numerantium* **156** (2002), 5–16.

[3] D. E. Brown, J. R. Lundgren and L. Sheng, A characterization of cycle-free unit probe interval graphs, *Discrete Appl. Math.* (to appear).

[4] D. E. Brown and J. R. Lundgren, Bipartite probe interval graphs, interval point bigraphs, and circular arc graphs, *Australas. J. Combin.* **35** (2006), 221–236.

[5] G. J. Chang, A. J. J. Kloks, J. Liu and S.-L. Peng, The pigs full monty — a floor show of minimal separators, *Lec. Notes in Comp. Sci.* **3404** (2005), 521–532.

[6] P. C. Fishburn, *Interval orders and interval graphs*, Wiley & Sons, 1985.

[7] D. R. Fulkerson and O. A. Gross, Incidence matrices and interval graphs, *Pacific J. Math.* **15** (1965), 835–855.

[8] Alan Gibbons, *Algorithmic graph theory*, Cambridge University Press, 1985.

[9] M. C. Golumbic, *Algorithmic graph theory and perfect graphs*, Academic Press, New York, 1980.

[10] M. C. Golumbic and M. Lipshteyn, On the hierarchy of interval, probe, and tolerance graphs, *Congressus Numerantium* **153** (2001), 97–106.

[11] J. L. Johnson and J. P. Spinrad, A polynomial time recognition algorithm for probe interval graphs, in *Proc. 12th ACM-SIAM Symp. on Discrete Algs. (SODA01)*, pp. 477–486, Association for Computing Machinery, New York, NY, 2001.

[12] M. Lipshteyn, *Probe graphs*, Master's thesis, Bar-Ilan University, 2001.

[13] R. M. McConnell and J. P. Spinrad, Construction of probe interval models, in *Proc. 13th ACM-SIAM Symp. on Discrete Algs. (SODA02)*, pp. 866–875, Association for Computing Machinery, San Francisco, CA, 2002.

[14] F. R. McMorris, C. Wang and P. Zhang, On probe interval graphs, *Discrete Appl. Math.* **88** (1998), 315–324.

[15] N. Pržulj and D. G. Corneil, 2-tree probe interval graphs have a large obstruction set, *Discrete Appl. Math.* **150** (2005), 216–231.

[16] F. S. Roberts, *Discrete mathematical models*, Prentice-Hall, Upper Saddle River, NJ, 1976.

[17] L. Sheng, Cycle-free probe interval graphs, *Congressus Numerantium* **140** (1999), 33–42.

[18] R. E. Tarjan, Depth-first search and linear graph algorithms, *SIAM J. Computing* **1** (1972), 142–160.

[19] P. Zhang, Probe interval graphs and their applications to physical mapping of DNA (1994), manuscript.

[20] P. Zhang, *Methods of mapping DNA fragments*, United States Patent, 1997. `http://www.cc.columbia.edu/cu/cie/techlists/patents/5667970.htm`.

[21] P. Zhang, E. A. Schon, S. F. 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** (1994), 309–317.

(Received 14 Feb 2010; revised 14 Sep 2011)