SIAM J. Discrete Math.
Vol. 20, No. 1, pp. 105–118

LINEAR ORDERINGS OF SUBFAMILIES OF AT-FREE GRAPHS*

DEREK G. CORNEIL†, EKKEHARD KÖHLER‡, STEPHAN OLARIU§, AND LORNA STEWART¶

Abstract. Asteroidal triple free (AT-free) graphs have been introduced as a generalization of interval graphs, since interval graphs are exactly the chordal AT-free graphs. While for interval graphs it is obvious that there is always a linear ordering of the vertices, such that for each triple of independent vertices the middle one intercepts any path between the remaining vertices of the triple, it is not clear that such an ordering exists for AT-free graphs in general.

In this paper we study graphs that are defined by enforcing such an ordering. In particular, we introduce two subfamilies of AT-free graphs, namely, path orderable graphs and strong asteroid free graphs. Path orderable graphs are defined by a linear ordering of the vertices that is a natural generalization of the ordering that characterizes cocomparability graphs. On the other hand, motivation for the definition of strong asteroid free graphs comes from the fundamental work of Gallai on comparability graphs.

We show that cocomparability graphs ⊂ path orderable graphs ⊂ strong asteroid free graphs ⊂ AT-free graphs. In addition, we settle the recognition question for the two new classes by proving that recognizing path orderable graphs is NP-complete, whereas the recognition problem for strong asteroid free graphs can be solved in polynomial time.

Key words. graph algorithms, complexity, asteroidal triple free graphs, recognition algorithm, linear ordering

AMS subject classifications. 05C75, 05C85, 68R10

DOI. 10.1137/S0895480104445307

1. Introduction.

We say that a vertex in a graph $G = (V, E)$ *intercepts* a path in G if it is adjacent to at least one vertex of the path, and it *misses* the path otherwise. An *asteroidal triple (AT)* is an independent set of three vertices such that, between every pair, there is a path that is missed by the third. A graph is *AT-free* if it does not contain an AT.

One of the most compelling motivations for the study of AT-free graphs is the idea that these graphs exhibit a type of linear structure. Indeed, the linear structure exhibited by AT-free graphs is explained, in part, in [1], where it is shown that every connected AT-free graph contains a *dominating pair* (two vertices such that every path connecting them is a dominating set) and a type of linear "shelling sequence" called a *spine*.

The original motivation for the results of the present paper was the idea that AT-free graphs might be characterized by the existence of a vertex ordering satisfying

*Received by the editors July 26, 2004; accepted for publication (in revised form) August 12, 2005; published electronically March 3, 2006. An expanded abstract of this paper appeared as *On subfamilies of AT free graphs*, in Graph-Theoretic Concepts in Computer Science (Boltenhagen, 2001), A. Brandstädt and V. B Le, eds., Lecture Notes in Comput. Sci. 2204, Springer-Verlag, New York, 2001, pp. 241–253. With kind permission of Springer Science and Business Media.

http://www.siam.org/journals/sidma/20-1/44530.html

†Computer Science Department, University of Toronto, Toronto, ON M55 3G4, Canada (dgc@cs.toronto.edu).

‡Institut für Mathematik, Technische Universität Berlin, 10623 Berlin, Germany (ekoehler@math.TU-Berlin.DE).

§Department of Computer Science, Old Dominion University, Norfolk, VA 23529-0162 (olariu@cs.odu.edu).

¶Department of Computing Science, University of Alberta, Edmonton, AB T6G 2E8, Canada (stewart@cs.ualberta.ca).

certain conditions. Looking back at the introduction of AT-free graphs as generalizations of interval graphs, there is an immediate candidate for such an ordering by requiring that for any independent triple in this ordering the central vertex should intercept every path between the remaining vertices of the triple. It is easy to see that interval graphs and even cocomparability graphs have such an ordering of the vertices (see below). However, it is not clear whether every AT-free graph possesses such an ordering.

Vertex orderings have proven to be useful algorithmic tools for several families of graphs. For example, chordal graphs (respectively, cocomparability graphs) are characterized by the existence of vertex orderings that do not contain the forbidden ordered configuration shown in Figure 1 (a) [2] (respectively, (b) [8]). A graph is an *interval graph* if and only if it has a vertex ordering that contains neither of the configurations of Figure 1 (see, for example, [11]). Such vertex orderings are referred to as chordal orderings, cocomparability orderings, and interval orderings, respectively.

FIG. 1. *Forbidden ordered configurations.*

In other words, in an interval ordering, for every path on two vertices (that is, for every edge), the left endpoint of the path is adjacent to all vertices between the two endpoints of the path. In a cocomparability ordering, each vertex between the two endpoints of a P_2 is adjacent to one or both endpoints of the P_2. It is well known that interval graphs are exactly those graphs that are both chordal and cocomparability [5] or, equivalently, both chordal and AT-free [9]. Furthermore, cocomparability graphs are a proper subclass of AT-free graphs [6].

An alternate characterization of the cocomparability ordering is given in Observation 1.1.

OBSERVATION 1.1. *A vertex ordering $v_1, \dots, v_n$ of graph G is a cocomparability ordering if and only if for all v_i, v_j, v_k with $i < j < k$, vertex v_j intercepts each v_i, v_k-path of G.*

From this, one can easily see that a cocomparability graph must be AT-free since any independent triple occurs in some order, say, $x \prec y \prec z$, in a cocomparability ordering "$\prec$," and thus, there cannot exist an x, z-path missed by y. In an attempt to generalize the cocomparability ordering while retaining the AT-free property, we introduce the following definition.

DEFINITION 1.2. *A graph $G = (V, E)$ is* path orderable *if there is an ordering $v_1, \dots, v_n$ of the vertices such that for each triple of vertices v_i, v_j, v_k with $i < j < k$ and $v_i v_k \notin E$, vertex v_j intercepts each v_i, v_k-path of G; such an ordering is called a* path ordering.

Observation 1.1 and Definition 1.2 imply that cocomparability graphs are path orderable. C_5, the chordless cycle on five vertices, is a path orderable graph which is not a cocomparability graph. It is clear that path orderable graphs must be AT-free. However, can Definition 1.2 be used for characterizing AT-free graphs? Figure 2 shows an AT-free graph together with an ordering that is not a path ordering. Hence, the question here is whether it can be turned into a path ordering. Unfortunately, we shall see later that path orderable graphs form a strict subset of AT-free graphs; in particular, the graph in Figure 2 will be shown to be not path orderable.

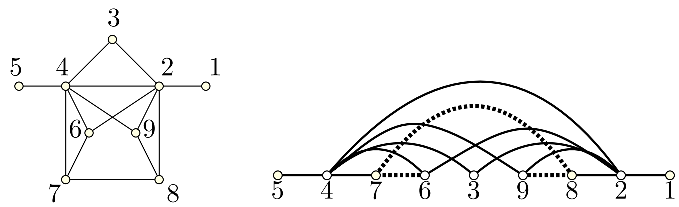

FIG. 2. *AT-free graph G with ordering that is not a path ordering; in particular, path* 6-7-8-9 *is not intercepted by* 3 *(the edges of the path are dashed).*

Nevertheless, since path orderable graphs are interesting in their own right, we attempted to provide a structural characterization of this graph class by identifying a type of forcing relation on nonadjacent pairs of vertices and the type of structure that makes the vertex ordering of Definition 1.2 impossible.

These investigations follow in Gallai's footsteps [3, 10] in that they involve ideas similar to his forcing relation on the edges of a comparability graph (equivalently, the nonedges of a cocomparability graph) and his definitions of wreaths and asteroids. Specifically, we define strong asteroids and show that path orderable graphs are strong asteroid free. However, it turns out that the strong asteroid concept does not provide a characterization of path orderable graphs; we shall see that path orderable graphs form a proper subclass of strong asteroid free graphs which, in turn, form a proper subclass of AT-free graphs.

Thus, we will identify two distinct subclasses of AT-free graphs, both of which contain cocomparability graphs:

cocomparability $\subset$ path orderable $\subset$ strong asteroid free $\subset$ AT-free.

The interest lies, in part, in the natural vertex ordering, in one case, and the relationship with Gallai's work, in the other case. Furthermore, the identification of these graph classes should allow us to narrow the gap between known polynomial and known NP-complete behavior of problems in the domain of AT-free graphs. For example, the complexity status for coloring, Hamiltonian path, and Hamiltonian cycle is still unresolved for AT-free graphs but is in P for cocomparability graphs.

We conclude the paper with a proof that the recognition of path orderable graphs is NP-complete, and with a polynomial time recognition algorithm for strong asteroid free graphs. We note that the NP-completeness result settles an open problem stated in [13].

Background. In his paper on comparability graphs [3, 10], Gallai studies the forcing between the edges imposed by a transitive orientation (to avoid misunderstandings, from now on we will refer to the transitive-forcing as *t-forcing*). Let G be an arbitrary graph. Two edges which share a common endpoint and whose other endpoints are nonadjacent t-force each other directly. That is, in any transitive orientation, either both edges are directed away from the common endpoint or both are directed toward it. The transitive closure of the direct t-forcing relation partitions the edges of G into *t-forcing classes*. Either there are exactly two different transitive orientations of the edges of a t-forcing class, or there is none. The latter case occurs when some edge is t-forced in both directions, in which case G is not a comparability graph. Edges xy and xz are said to be *knotted* if y and z are connected in $\overline{G}[\mathrm{N}(x)]$, the complement of the subgraph of G induced by $\mathrm{N}(x)$, where $\mathrm{N}(x)$, the neighborhood of x, is defined as $\mathrm{N}(x) = \{u \mid ux \in E\}$.

To capture the t-forcing in a given graph G, Gallai uses the concept of a *knotting graph*: For a graph $G = (V, E)$ the corresponding *knotting graph* is given by $\mathrm{K}[G] = (V_{\mathrm{K}}, E_{\mathrm{K}})$, where V_{K} and E_{K} are defined as follows. For each vertex v of G there are copies $v_1, v_2, \ldots, v_{i_v}$ in V_{K}, where i_v is the number of connected components of $\overline{G}[\mathrm{N}(v)]$. For each edge vw of E there is an edge $v_i w_j$ in E_{K}, where w is contained in the ith connected component of $\overline{G}[\mathrm{N}(v)]$ and v is contained in the jth connected component of $\overline{G}[\mathrm{N}(w)]$. Please refer to Figure 5 for an example of a graph together with its knotting graph.

In this graph two edges are incident if and only if they are knotted. The edges of the t-forcing classes of G are given by the connected components of $\mathrm{K}[G]$. Using this structure, Gallai shows that a graph G is a comparability graph if and only if $\mathrm{K}[G]$ is bipartite.

The following definitions from [3] describe structures which lead to t-forcing classes which cannot be transitively oriented and knotting graphs which are not bipartite.

Definition 1.3. *An* odd wreath *of size k in a graph is a cycle of knotted edges, specifically, a sequence of vertices $v_0, v_1, v_2, \ldots, v_k$, where k is odd, $v_1, \ldots, v_k$ are distinct, $v_0 = v_k$, and for all i, $0 \le i < k$, edges $v_i v_{i+1}$ and $v_{i+1} v_{i+2}$ exist in the graph and are knotted (addition modulo k).*

Definition 1.4. *An* odd asteroid *of size k in a graph is a sequence of vertices $v_0, v_1, v_2, \ldots, v_k$ where k is odd, $v_1, \ldots, v_k$ are distinct, $v_0 = v_k$, and for all i, $0 \le i < k$, there exists a $v_i v_{i+1}$-path in G which is missed by $v_{(i+\frac{k+1}{2})}$ (addition modulo k).*

Gallai points out that an odd asteroid is the complement of an odd wreath and proves that a graph is a comparability graph if and only if it contains no odd wreath or, equivalently, a graph is a cocomparability graph if and only if it contains no odd asteroid. Note also that an AT corresponds to an odd asteroid of size three.

As an example of an odd asteroid, consider the graph G in Figure 2. Here, the sequence of vertices $1, 3, 5, 7, 8, 1$ forms an odd asteroid of size 5 in G. The sequence $1, 5, 8, 3, 7, 1$ of vertices forms an odd wreath of size 5 in $\overline{G}$.

2. Path orderable graphs and strong asteroid free graphs. As we have seen, t-forcing is a fundamental concept for comparability graphs, and thus for cocomparability graphs as well. Given the similarities of the linear ordering characterizations of path orderable graphs and cocomparability graphs, one might expect a similar forcing concept for path orderable graphs. In fact such is the case.

For a graph G and a vertex v of G let $C_1, \ldots, C_k$ be the connected components of $G \setminus \mathrm{N}[v]$ and let $B_i^1, \ldots, B_i^\ell$ be the connected components of the graph induced by the vertices of C_i in $\overline{G}$ ($1 \le i \le k$); the B_i^j are called the *blobs* of v in G. (Here $\mathrm{N}[v] := \mathrm{N}(v) \cup \{v\}$ denotes the closed neighborhood of vertex v in G.) As an example, consider the graph in Figure 3.

Lemma 2.1. *Let G be a path orderable graph and let $v_1, \ldots, v_n$ be a path ordering of G. For every vertex v of G and for every blob B of v, the vertices of B occur either all before v in the path ordering or all after v in the path ordering.*

Proof. Suppose there are a vertex v and a blob B of v with $u, w \in B$ and $u \prec v \prec w$ in the path ordering "$\prec$" of G (see Figure 4 for a sketch of this setting). By the definition of blobs, u and w are in the same connected component C of $G \setminus \mathrm{N}[v]$. Since u and w are also in the same connected component B of C in $\overline{G}$, there has to be a path of nonedges in B between u and w. Thus, there is a pair of vertices u', w'

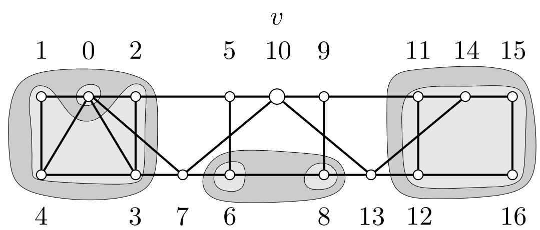

FIG. 3. *The blobs of vertex* $v = 10$ *are given by the sets* $\{0\}$, $\{1, 2, 3, 4\}$, $\{6\}$, $\{8\}$, $\{11, 12, 14, 15, 16\}$.

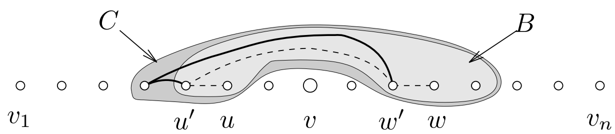

FIG. 4. *Proof idea of Lemma* 2.1.

in B with $u'w' \notin E$ and $u' \prec v \prec w'$. But $u', w' \in C$; therefore there is a u', w'-path in $G \setminus \mathrm{N}[v]$, contradicting the path ordering. ∎

By Lemma 2.1, any path ordering has to fulfill the property that if one of the vertices u of a blob B of v precedes v in the ordering, then all of the vertices of B occur before v.

Consider now the graph G in Figure 2. Following the above definition of blobs, vertex 3 has the three blobs $\{6, 7, 8, 9\}$, $\{5\}$, $\{1\}$; vertex 7 has the blobs $\{3, 1, 9\}$, $\{2\}$, $\{5\}$; vertex 8 has the blobs $\{3, 5, 6\}$, $\{4\}$, $\{1\}$; vertex 5 has only the blob $\{1, 2, 3, 6, 7, 8, 9\}$; and vertex 1 has only the blob $\{3, 4, 5, 6, 7, 8, 9\}$. Suppose there is a path ordering of G. By Lemma 2.1 we can, without loss of generality, assume that 1 precedes all vertices of its blob and thus 5 appears after all vertices of its blob in the path ordering; in particular, vertices 3, 6, 7, 8, 9 are between 1 and 5. Since 7 and 8 are in the same blob of 3, they appear either both before or both after 3 in the path ordering. However, if they both appear before 3, then, again by Lemma 2.1, we have a contradiction because 3 and 1 are in the same blob of 7, but on different sides in the path ordering. On the other hand, if both 7 and 8 appear after 3 in the path ordering we again have a contradiction, since 3 and 5 are in the same blob of 8 but on different sides in the path ordering. Hence there cannot be a path ordering for the graph in Figure 2.

COROLLARY 2.2. *The class of path orderable graphs is strictly contained in the class of AT-free graphs.*

LEMMA 2.3. *If a graph G is path orderable then every induced subgraph of G is path orderable.*

Proof. This follows by the definition of path orderable and since any path in an induced subgraph of graph G is also a path in G. ∎

When interpreting the constraints of Lemma 2.1 as orientations of the edges of $\overline{G}$, in the sense that edges from the same blob of a vertex v to v in $\overline{G}$ have to have the same orientation (i.e., representing before or after v in the path ordering), one can define the following forcing on the edge set of $\overline{G}$.

Let G be an arbitrary graph and let $e_1 = uv$, $e_2 = vw$ be edges of $\overline{G}$ with a common end-vertex v. Then one can define a relation $\approx$ by $e_1 \approx e_2$ (e_1 and e_2 *force each other* or *are knotted* at v) if and only if u and w are in the same blob of v (possibly $u = w$) in G. The transitive closure of this relation defines a class partition of the edges of $\overline{G}$, where two edges e_a, e_b are in the same class (*forcing class*) of $\overline{G}$ if there is a sequence $e_1, e_2, \ldots, e_k$ of edges such that $e_a = e_1 \approx e_2 \approx \cdots \approx e_k = e_b$. Observe that the forcing classes are refinements of the t-forcing classes.

An orientation of the edges of $\overline{G}$ is said to *agree with the forcing* if for any vertex v and any blob B of v all edges between B and v are oriented in the same direction (either toward v or away from v). For a graph G a linear ordering $v_1, \ldots, v_n$ of the vertices of G is said to *agree with the forcing* if the corresponding implied orientation of the edges of $\overline{G}$ (uv is oriented from u to v if $u \prec v$ in the linear ordering "$\prec$") agrees with the forcing.

Note that when the orientation of one of the edges of a forcing class is fixed, then the orientation of all the edges of its forcing class is determined; hence, either there are exactly two different orientations of the edges of a forcing class that agree with the forcing, or there is none. In the latter case, some edge is forced to be oriented in both directions, meaning that there is no ordering consistent with the forcing.

LEMMA 2.4. *A graph G is path orderable if and only if there is a linear ordering of the vertices of G agreeing with the forcing.*

Proof. If G is path orderable, then, by Lemma 2.1, the path ordering has to agree with the forcing relation.

Suppose there is a linear ordering "$\prec$" of G that agrees with the forcing relation and suppose there is a triple $u \prec v \prec w$ of vertices that violates the path ordering property, i.e., $uw \notin E$, and there is a u,w-path in $G \setminus \mathrm{N}[v]$. Hence, u and w are in the same connected component C of $G \setminus \mathrm{N}[v]$ and, since $uw \notin E$, u and w are also in the same blob B of v. But then this ordering does not agree with the forcing relation, which is a contradiction. ∎

COROLLARY 2.5. *A graph G is path orderable if and only if there is an acyclic orientation of $\overline{G}$, agreeing with the forcing relation.*

Proof. Determine a topological ordering, using the acyclic orientation of $\overline{G}$; then the corollary follows from Lemma 2.4. ∎

One can define a graph, similar to Gallai's knotting graph, representing the forcing classes of $\overline{G}$. For a graph $G = (V, E)$ the *altered knotting graph* is given by $\mathrm{K}^*[G] = (V_K, E_K)$, where V_K and E_K are defined as follows. For each vertex v of G there are copies $v_1, \ldots, v_{i_v}$ in V_K, where i_v is the number of blobs of v in $\overline{G}$. For each edge vw of E there is an edge $v_i w_j$ in E_K, where w is contained in the ith blob of v in $\overline{G}$ and v is contained in the jth blob of w in $\overline{G}$.

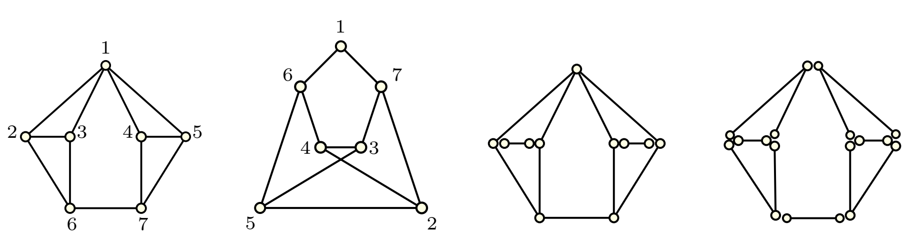

FIG. 5. *A graph G together with its complement $\overline{G}$, $\mathrm{K}[G]$, and $\mathrm{K}^*[G]$.*

As Gallai did for the knotting graph, we draw the altered knotting graph $\mathrm{K}^*[G]$ of a given graph G by putting different copies of the same vertex close together. See Figure 5 for an example of a graph G, together with its complementary graph $\overline{G}$, its knotting graph $\mathrm{K}[G]$, and its altered knotting graph $\mathrm{K}^*[G]$. The blobs of the vertices of $\overline{G}$ are as follows: vertex 1: $\{2,3\}$, $\{4,5\}$; vertex 2: $\{1\}$, $\{3\}$, $\{6\}$; vertex 3: $\{1\}$, $\{2\}$, $\{6\}$; vertex 4: $\{1\}$, $\{5\}$, $\{7\}$; vertex 5: $\{1\}$, $\{4\}$, $\{7\}$; vertex 6: $\{2,3\}$, $\{7\}$; vertex 7: $\{4,5\}$, $\{6\}$.

Our next task is to examine configurations which cannot occur in path orderable graphs. As a step toward this goal, we define restricted types of odd wreaths and asteroids.

DEFINITION 2.6. *An* odd strong wreath *of size k in a graph G is a sequence of vertices $v_0, v_1, \ldots, v_k$ where k is odd, $v_1, \ldots, v_k$ are distinct, $v_0 = v_k$, and for all i, $0 \le i < k$, edges $v_i v_{i+1}$ and $v_{i+1} v_{i+2}$ exist in the graph and are knotted in the altered sense; that is, v_i and v_{i+2} are in the same blob of v_{i+1} in $\overline{G}$ (addition modulo k).*

DEFINITION 2.7. *An* odd strong asteroid *of size k in a graph G is a sequence of vertices $v_0, v_1, \ldots, v_k$ where k is odd, $v_1, \ldots, v_k$ are distinct, $v_0 = v_k$, and for all i, $0 \le i < k$, v_i and v_{i+1} are in the same blob of $v_{(i+\frac{k+1}{2})}$ in G (addition modulo k).*

The two notions are complementary; that is, a graph G has an odd strong wreath if and only if $\overline{G}$ contains an odd strong asteroid. Furthermore, strong asteroids and strong wreaths are restricted types of asteroids and wreaths. We also note that the ATs correspond to the odd strong asteroids of size three. Figure 6 features a graph containing an odd strong asteroid as well as its complement that contains an odd strong wreath.

FIG. 6. *Graph of Figure 2, containing an odd strong asteroid and its complement, containing an odd strong wreath (vertices of the asteroid and the wreath are marked by $v_1, \ldots, v_5$ and $w_1, \ldots, w_5$, respectively; the edges of the wreath are dashed).*

DEFINITION 2.8. *A graph G is* strong asteroid free *if it does not contain an odd strong asteroid.*

Similar to the t-forcing results, the following holds.

LEMMA 2.9. *The forcing classes of a graph G are precisely the connected components of* $K^*[G]$.

The next two observations follow from the fact that an odd strong asteroid of size k in G corresponds to an odd cycle of size k in $K^*[\overline{G}]$.

OBSERVATION 2.10. *A graph G is strong asteroid free if and only if $K^*[\overline{G}]$ is bipartite.*

OBSERVATION 2.11. *A graph G is AT-free if and only if $K^*[\overline{G}]$ is triangle-free.*

LEMMA 2.12. *If a graph G is path orderable then $K^*[\overline{G}]$ is bipartite.*

Proof. Let $v_1, \ldots, v_n$ be a path ordering of G. Now orient the edges of $K^*[\overline{G}]$ as follows: $v_i v_j$ is oriented from v_i to v_j if $i < j$. Now, by Lemma 2.1, each vertex of $K^*[\overline{G}]$ has either only incoming or only outgoing edges. Hence, it is bipartite. ∎

Not only does the graph in Figure 2 show that path orderable graphs are strictly contained in AT-free graphs, but it also establishes that strong asteroid free graphs are strictly contained in AT-free graphs, as shown in the next lemma.

LEMMA 2.13. *The class of strong asteroid free graphs is strictly contained in the class of AT-free graphs.*

Proof. Consider the graphs of Figures 2 and 6. It is easy to check that the vertices named $v_1, \ldots, v_5$ in Figure 6 form an odd strong asteroid in G, and that G is AT-free. ∎

Similar to Lemma 2.3 one can prove the following lemma.

LEMMA 2.14. *If a graph G is strong asteroid free then every induced subgraph of G is strong asteroid free.*

In the case of comparability graphs, Gallai not only showed that the knotting graph $\mathrm{K}[G]$ of a comparability graph is bipartite but also proved that a bipartite knotting graph $\mathrm{K}[G]$ is a sufficient condition for G being a comparability graph. The major tool that he used for proving this result is a lemma which shows the following. Given a bipartite knotting graph $\mathrm{K}[G]$ consider a triangle of G with the property that at least two of the edges of the triangle are in the same t-forcing class; then in any orientation of G that agrees with the t-forcing, the triangle is not oriented cyclically.

It turns out that a similar lemma holds for strong asteroid free graphs, too. Specifically, for a graph G with a bipartite altered knotting graph $\mathrm{K}^*[\overline{G}]$, any orientation of $\overline{G}$ that agrees with the forcing relation does not contain a cyclically oriented triangle. However, contrary to the t-forcing relation, this lemma is not enough to imply that the orientation is acyclic and, indeed, we shall show that this is not necessarily the case.

OBSERVATION 2.15. *Given a vertex v in a graph H and vertices $u, w \in \mathrm{N}(v)$, which are the endpoints of an induced P_4 in $\mathrm{N}(v)$, then the edges uv and wv force each other (see Figure 7).*

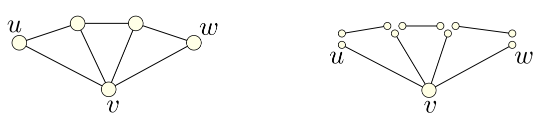

FIG. 7. *Vertex v with P_4 in $\mathrm{N}(v)$ together with the corresponding altered knotting graph.*

Remark 2.16. Using this observation one can create a *forcing path*, i.e., a path P, where each consecutive pair of edges of P is knotted at the common end-vertex by the help of an added P_4 as described in Observation 2.15; see Figure 8 (in the following, edges and vertices of the path P itself are called *original edges/vertices*, and the added edges and vertices are denoted as *auxiliary edges/vertices*). By the forcing, the orientation of any original edge of P forces the orientation of all other original edges of P. Note that the knotting graph of a forcing path does not contain a triangle or any odd cycle. Furthermore, if P has even length, then the end-edges of P are either both oriented toward the inner vertices of P or both oriented outward from the inner vertices of P. Similarly, if P has odd length, the end-edges of P have opposite orientations with respect to the inner vertices of P.

THEOREM 2.17. *The class of path orderable graphs is strictly contained in the class of strong asteroid free graphs.*

Proof. Consider the left graph in Figure 9. This graph is the complement of a strong asteroid free graph G. This is proved by constructing the altered knotting graph $\mathrm{K}^*[\overline{G}]$ (see the right graph in Figure 9). By Observation 2.15, the thick edges force each other, as shown in the altered knotting graph; and, without having a

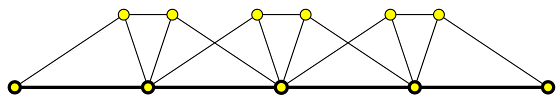

FIG. 8. *A forcing path of length* 4 *(original edges and vertices are bold).*

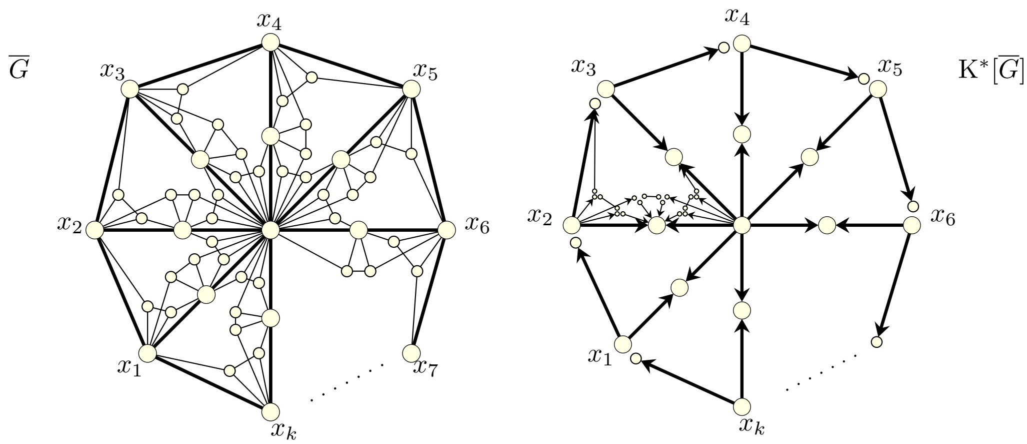

FIG. 9. *Complement of a strong asteroid free graph, which is not path orderable (left), together with its altered knotting graph (right). To ease understanding of its structure, in the knotting graph the corresponding auxiliary P_4 vertices are drawn in the figure for only one of the arms of the example. One of the two possible forced orientations of the main forcing class is given in the right picture.*

strong asteroid in G, there is a forced oriented cycle on the vertices $x_1, \ldots, x_k$ in $\overline{G}$. Consequently, by Corollary 2.5, G is not path orderable. This construction holds for any $k \geq 4$. ∎

3. Recognition of path orderable and strong asteroid free graphs. In this section, we show that the recognition of path orderable graphs is NP-complete. This result answers a question posed by Spinrad in [13]. In contrast, we describe how to recognize strong asteroid free graphs in polynomial time.

First, observe that the recognition problem of path orderable graphs is obviously in NP, since by Lemma 2.1 for a given ordering one can easily check in polynomial time whether it is a path ordering. If there is only one forcing class for the edge set of $\overline{G}$ one can also check in polynomial time whether G is path orderable: Compute $K^*[\overline{G}]$, check whether it is bipartite, assign an orientation to $K^*[\overline{G}]$ by orienting all edges from one of the bipartition classes to the other, and check whether this orientation is acyclic on $\overline{G}$.

Similarly one can check whether G is path orderable if the number of forcing classes of $\overline{G}$ is bounded by a constant.

For comparability graphs, Gallai's results for the general case, i.e., where no assumption on the number of edge classes is made, lead to a polynomial time recognition algorithm. For this he introduced the (by now well-known) concept of modular decomposition and proved that, using this decomposition scheme, the problem of recognizing comparability graphs reduces to the problem of recognizing prime comparability graphs. But what about the recognition of path orderable graphs? Can one extend the decomposition scheme to this problem?

NOT-ALL-EQUAL 3SAT. [4]
INSTANCE: Set U of variables, collection $\mathcal{C}$ of clauses over U such that each clause $c \in \mathcal{C}$ has $|c| = 3$.
QUESTION: Is there a truth assignment A for U such that each clause in C has at least one *true* literal and at least one *false* literal?

Remark 3.1. Without loss of generality one can assume that none of the clauses contains more than one literal of a variable.

To prove the NP-hardness of the recognition problem of path orderable graphs, we use a transformation from NOT-ALL-EQUAL 3SAT (NAE 3SAT). Given an instance I of NAE 3SAT, a graph G is constructed, which is the complement of a path orderable graph if and only if I is NAE 3SAT-satisfiable. In particular, it will be shown that I is NAE 3SAT-satisfiable if and only if there is an acyclic orientation of G that agrees with the forcing. By Corollary 2.5 this is equivalent with $\overline{G}$ being path orderable.

The basic construction of G is as follows. For every variable x of U an edge e_x is created (called a *variable edge* in the following) and the two possible orientations of e_x are associated with the two possible values *true* and *false* of x.

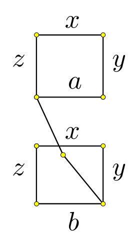

FIG. 10. *Gadget for clause* $x \vee y \vee z$.

For each clause $C = x \vee y \vee z$ with literals x, y, z a gadget is constructed, mainly consisting of two C_4's as shown in Figure 10. In each of the C_4's three of the edges (the *base-edges*) correspond to the three literals x, y, z of C. As will be explained below, a *true* literal of C will correspond to a clockwise orientation of the corresponding base-edges in both of the C_4's, whereas a *false* literal will correspond to a counterclockwise orientation of the corresponding base-edges in both C_4's. Furthermore, in each orientation that agrees with the forcing, the fourth edges of the two C_4's, which will be called the *bridge edges* (edges a and b in Figure 10), will be guaranteed to have opposite orientations in the two C_4's. This is realized by making these bridge edges the end-edges of a forcing path of length 4. Consequently, with this construction, a truth assignment of the variables of U that sets all three literals of C to *true* (*false*) results in a clockwise (counterclockwise) orientation of all three base-edges in both C_4's and, since the bridge edges have opposite orientations in the two C_4's, at least one of the C_4's has a cyclic orientation. On the other hand, by the above correspondence between the orientations of the base-edges and the truth-values of the corresponding literals, each acyclic orientation of G that agrees with the forcing leaves at least one literal of C *true* and one *false*.

Next, it has to be ensured that the value of a variable and the value of the literals of this variable coincide; i.e., the orientation of the variable edge of x for value *true* has to result in a counterclockwise orientation of the base-edges for $\overline{x}$ in all the gadgets

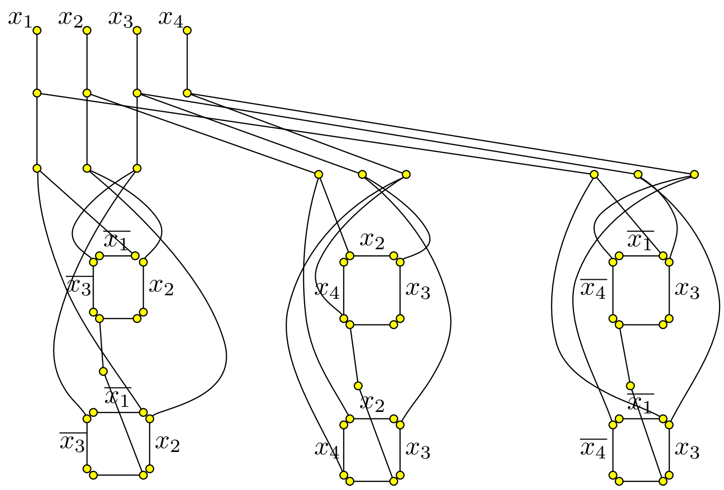

FIG. 11. *General structure of* $\mathrm{K}^*[G]$ *for the instance* $I = (\overline{x_1} \vee x_2 \vee \overline{x_3}) \wedge (x_2 \vee x_3 \vee x_4) \wedge (\overline{x_1} \vee x_3 \vee \overline{x_4})$ *(auxiliary vertices and edges are omitted).*

for clauses containing literal $\overline{x}$ and in a clockwise orientation of the base-edges for x in all the gadgets for clauses containing literal x. This is realized by connecting each variable edge to all corresponding base-edges by the help of forcing paths that are joined appropriately. In other words, for each variable a separate edge class is created, containing the variable edge and all base-edges corresponding to literals of this variable. The general structure of the connection between variable edges and base-edges by forcing paths is shown in Figure 11; for easier understanding the auxiliary edges and vertices of the forcing paths are omitted in this picture. For a variable edge e_x (see top of Figure 11) a downward orientation corresponds to assigning *false* to variable x, whereas an upward orientation corresponds to assigning *true* to x. For each literal x or $\overline{x}$, there is a forcing path of length 4, having e_x and the corresponding base-edge as its end-edges; depending on whether the literal is $\overline{x}$ or x, either the start- or the end-vertex of the base-edge (with respect to a clockwise ordering in the C_4) is made the end-vertex of the forcing path.

Now, by Remark 2.16, assigning an upward orientation to the variable edge e_x results in the desired clockwise orientation of the base-edges of the literals x and a counterclockwise orientation of the base-edges of the literals $\overline{x}$ for any orientation agreeing with the forcing.

In Figure 12 (left) the complete construction of G for a single clause C together with the variable edges and the forcing paths is given, including all auxiliary edges and vertices. In the right part of the figure the corresponding altered knotting graph $\mathrm{K}^*[G]$ is shown.

We now study properties of orientations of G that agree with the forcing. For this it is sufficient to consider $\mathrm{K}^*[G]$. Observe first, that, by the construction, $\mathrm{K}^*[G]$ is bipartite; indeed, $\mathrm{K}^*[G]$ is even a forest and for each of the variables there is exactly one connected component in $\mathrm{K}^*[G]$ that contains both the variable edge and all base-edges corresponding to this variable. Note furthermore that an oriented cycle in an orientation of G can contain neither a source nor a sink vertex of that orientation. Consequently, all the vertices of G, having only one copy in $\mathrm{K}^*[G]$, cannot be contained

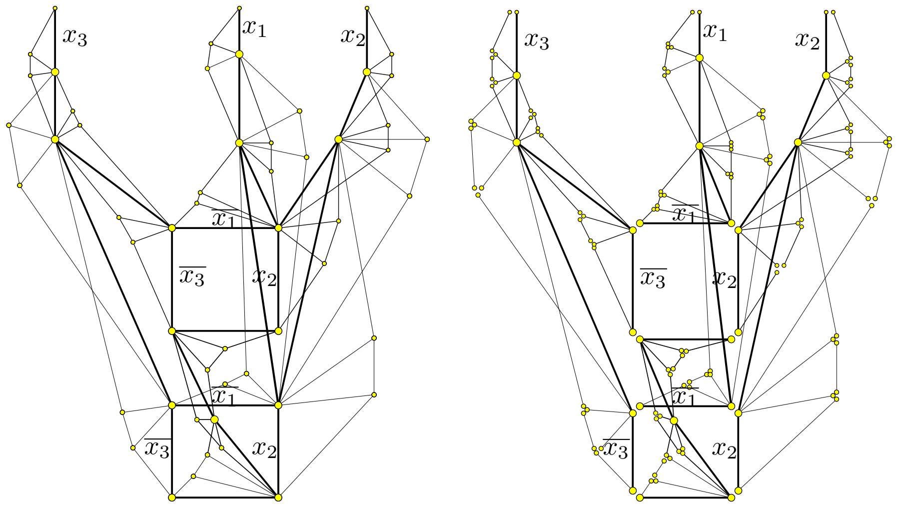

FIG. 12. *Left: Complete construction for a gadget of the clause* $(\overline{x_1} \vee x_2 \vee \overline{x_3})$ *together with the variable edges and the forcing paths. Right: The corresponding altered knotting graph.*

in any such cycle, since they have to be sources or sinks in any orientation of G, which agrees with the forcing. After deleting all those vertices from G, the only cycles of the remaining graph are the two four-cycles per gadget and some triangles, each consisting of auxiliary edges and at most one of the C_4-edges (see Figure 13). Consider any of

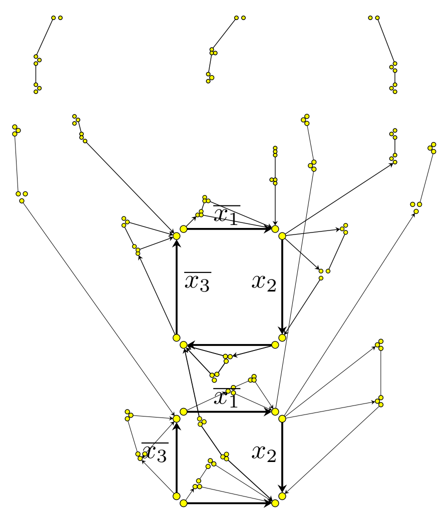

FIG. 13. *A clause-gadget after removing all source and sink vertices.*

those remaining triangles. By the construction, at least two of the three triangle-edges are incident to the same vertex of $\mathrm{K}^*[G]$. Consequently, in any orientation that agrees with the forcing relation, these two edges prevent the triangle from being cyclically oriented. Hence, when checking an orientation (that agrees with the forcing) of the constructed graph G to be acyclic, it is sufficient to show that each of the two C_4's per gadget is acyclically oriented.

OBSERVATION 3.2. *Given an orientation of G that agrees with the forcing, this orientation is acyclic if and only if it is acyclic on both C_4's of each of the clause gadgets.*

Now we are ready to show the following lemma.

LEMMA 3.3. *There is an acyclic orientation of G agreeing with the forcing relation if and only if $\mathcal{C}$ has an NAE 3SAT satisfying assignment.*

Proof. Suppose that there is an NAE 3SAT satisfying assignment A. An acyclic orientation of G that agrees with the forcing can be constructed as follows. We assign orientations to the variable edges (the edges on top of Figure 11) by orienting an edge downward if the corresponding variable is set *false* in A and upward otherwise. Consequently, all edges of the connected components of those edges in $\mathrm{K}^*[G]$ have a forced orientation as well.

The only edges that have not been assigned an orientation in this way are the forcing classes of the bridge edges of every C_4 and the single edges of the auxiliary P_4's (see connected components of the knotting graph in Figure 12, right). The single edges can be assigned an arbitrary orientation and for each of the bridge edge classes just one edge is oriented arbitrarily, forcing the orientation of all other edges of this class. Obviously, this orientation agrees with the forcing.

By the forcing of the edges and the appropriate knotting of the forcing path from the variable representing edges to the edges representing the literals, each *true* literal in a clause C leads to a clockwise oriented edge, and analogously, each *false* literal implies a counterclockwise oriented edge in the corresponding C_4's. Since every clause has at least one *true* and one *false* literal, each of the C_4's has both an edge that is oriented clockwise and one that is oriented counterclockwise. Hence, none of the C_4's is cyclically oriented and, by Observation 3.2, the orientation is acyclic.

Suppose now that there is an acyclic orientation of G that agrees with the forcing relation. We assign to a variable x of U the value *true* if the edge representing variable x (edges on top of Figure 11) is oriented upward and *false* otherwise. Since the orientation agrees with the forcing relation, all we have to show is that all of the clauses have at least one *true* and one *false* literal. Suppose there is a clause C, which has only *true* (*false*) literals. By the definition of G and the forcing relation, three edges in each of the C_4's in C's gadget are oriented counterclockwise (clockwise). Since the bridge edges have opposite orientations in the two C_4's of C, exactly one of the C_4's is oriented cyclically, contradicting that the orientation of G is acyclic. $\square$

Since it is easy to see that the construction of graph G is polynomial in the size of the input U and $\mathcal{C}$, Lemma 3.3 directly implies the following theorem.

THEOREM 3.4. *The problem of deciding whether a graph is path orderable is NP-complete.*

In contrast to Theorem 3.4, a polynomial time recognition algorithm for strong asteroid free graphs follows from Observation 2.10. Given graph G, the altered knotting graph of $\overline{G}$, $\mathrm{K}^*[\overline{G}]$, can be computed in polynomial time: for each vertex v of G, the blobs of v in G can be computed in $O(n^2)$ time; each vertex has fewer than n blobs. Thus, $\mathrm{K}^*[\overline{G}]$ has $O(n^2)$ vertices and $O(n^2)$ edges (since each edge of $\overline{G}$ corresponds to exactly one edge of $\mathrm{K}^*[\overline{G}]$) and can be constructed in $O(n^3)$ time. To test whether $\mathrm{K}^*[\overline{G}]$ is bipartite can be done in $O(n^2)$ time. Overall, the recognition algorithm requires $O(n^3)$ time.

THEOREM 3.5. *Strong asteroid free graphs can be recognized in time $O(n^3)$.*

4. Concluding remarks. We have defined two graph classes and shown that cocomparability graphs ⊂ path orderable graphs ⊂ strong asteroid free graphs ⊂ AT-free graphs. Furthermore, we have shown that the recognition problem for path orderable graphs is NP-complete, and the recognition of strong asteroid free graphs can be solved in polynomial time. We note that AT-free graph recognition is also in P [1, 7].

Although it is somewhat disappointing that no two of these families are equivalent, these classes may give insight into some open problem complexities on AT-free graphs. By adding graph classes in the hierarchy between cocomparability graphs and AT-free graphs, we may be able to identify more precisely the boundary between polynomial and NP-complete behavior of some of the problems which are known to be polynomially solvable on cocomparability graphs but either NP-complete or unresolved on AT-free graphs. Examples of such problems include graph coloring, clique cover, clique, and the Hamiltonian path and cycle problems. One step in this direction is the observation that the clique problem is NP-complete for path orderable graphs. This follows from the facts that the complements of triangle-free graphs are contained in path orderable graphs, and the independent set problem is known to be NP-complete on triangle-free graphs [12].

Acknowledgment. The authors wish to thank the Natural Science and Engineering Research Council of Canada for financial support.

REFERENCES

[1] D. G. Corneil, S. Olariu, and L. Stewart, *Asteroidal triple-free graphs*, SIAM J. Discrete Math., 10 (1997), pp. 399–430.
[2] R. R. Fulkerson and O. A. Gross, *Incidence matrices and interval graphs*, Pacific J. Math., 15 (1965), pp. 835–855.
[3] T. Gallai, *Transitiv orientierbare Graphen*, Acta Math. Acad. Sci. Hungar., 18 (1967), pp. 25–66.
[4] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman, New York, 1979, ch. A9, p. 259.
[5] P. C. Gilmore and A. J. Hoffman, *A characterization of comparability graphs and of interval graphs*, Canad. J. Math., 16 (1964), pp. 539–548.
[6] M. C. Golumbic, C. L. Monma, and W. T. Trotter, *Tolerance graphs*, Discrete Appl. Math., 9 (1984), pp. 157–170.
[7] E. Köhler, *Graphs without asteroidal triples*, Ph.D. thesis, Technische Universität Berlin, Berlin, Germany, 1999.
[8] D. Kratsch and L. Stewart, *Domination on cocomparability graphs*, SIAM J. Discrete Math., 6 (1993), pp. 400–417.
[9] C. G. Lekkerkerker and J. C. Boland, *Representation of a finite graph by a set of intervals on the real line*, Fund. Math., 51 (1962), pp. 45–64.
[10] F. Maffray and M. Preissmann, *A translation of Tibor Gallai's paper: Transitiv orientierbare Graphen*, in Perfect Graphs, J. Ramirez-Alfonsin and B. Reed, eds., John Wiley, New York, 2001, pp. 25–66.
[11] S. Olariu, *An optimal greedy heuristic to color interval graphs*, Inform. Process. Lett., 37 (1991), pp. 65–80.
[12] S. Poljak, *A note on stable sets and colorings of graphs*, Comment. Math. Univ. Carolin., 15 (1974), pp. 307–309.
[13] J. P. Spinrad, *Efficient Graph Representations*, Fields Inst. Monogr. 19, AMS, Providence, RI, 2003.